OSINT TECHNIQUES: RESOURCES FOR UNCOVERING ONLINE INFORMATION

MICHAEL BAZZELL

TENTH EDITION

Order 0138471223

OSINT TECHNIQUES

RESOURCES FOR UNCOVERING ONLINE INFORMATION

TENTH (DIGITAL) EDITION

MICHAEL BAZZELL

Order 0138471223

OSINT TECHNIQUES:
RESOURCES FOR UNCOVERING ONLINE INFORMATION
TENTH (DIGITAL) EDITION

Copyright © 2023 by Michael Bazzell

Project Editors: Y. Varallo, Janice Bartlett

Technical Editors: Jason Edison, David Westcott, Peter Richardson, Eric Swenson, Rhianna Huizenga

Cover Concept: "Anonymous Reader", Cover Modification: Astropost.io

All rights reserved. No part of this book may be reproduced in any form or by any electronic or mechanical means, including information storage and retrieval systems, without permission in writing from the author.

Tenth Edition First Published: January 2023

The information in this book is distributed on an "As Is" basis, without warranty. The author has taken great care in preparation of this book, but assumes no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

Rather than use a trademark symbol with every occurrence of a trademarked name, this book uses the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Due to the use of quotation marks to identify specific text to be used as search queries and data entry, the author has chosen to display the British rule of punctuation outside of quotes. This ensures that the quoted content is accurate for replication. The author has also chosen to omit smart or curly single and double quotes in order to maintain proper emphasis within search queries and scripts. Only "straight" quotation marks and apostrophes are presented. To maintain consistency, these formats are continued throughout the entire book.

The technology referenced in this book was edited and verified by a professional team for accuracy. Exact tutorials in reference to websites, software, and hardware configurations change rapidly. All tutorials in this book were confirmed accurate as of January 1, 2023. Readers may find slight discrepancies within the methods as technology changes.

Digital Revision 2024.02.11

Order 0138471223

SECTION I: OSINT Preparation **11**
CHAPTER 1: Computer Optimization **12**
CHAPTER 2: Linux Virtual Machines **24**
CHAPTER 3: Web Browsers **39**
CHAPTER 4: Linux Applications **62**
CHAPTER 5: VM Maintenance & Preservation **107**
CHAPTER 6: Mac & Windows Hosts **125**
CHAPTER 7: Android Virtualization **138**
CHAPTER 8: Custom Search Tools **150**
SECTION II: OSINT Resources & Techniques **158**
CHAPTER 9: Search Engines **161**
CHAPTER 10: Social Networks: Facebook **188**
CHAPTER 11: Social Networks: Twitter **204**
CHAPTER 12: Social Networks: Instagram **226**
CHAPTER 13: Social Networks: General **237**
CHAPTER 14: Online Communities **252**
CHAPTER 15: Email Addresses **272**
CHAPTER 16: Usernames **284**
CHAPTER 17: People Search Engines **291**
CHAPTER 18: Telephone Numbers **303**
CHAPTER 19: Online Maps **312**
CHAPTER 20: Documents **328**
CHAPTER 21: Images **338**
CHAPTER 22: Videos **354**
CHAPTER 23: Broadcast Streams **366**
CHAPTER 24: Domain Names **377**
CHAPTER 25: IP Addresses **399**
CHAPTER 26: Government & Business Records **408**
CHAPTER 27: Virtual Currencies **417**
CHAPTER 28: Application Programming Interfaces (APIs) **421**
CHAPTER 29: Advanced Linux Tools **434**
SECTION III: Leaks, Breaches, Logs, & Ransomware **445**
CHAPTER 30: Data Leaks **449**
CHAPTER 31: Data Breaches **471**
CHAPTER 32: Stealer Logs **488**
CHAPTER 33: Ransomware **499**
SECTION IV: OSINT Methodology **508**
CHAPTER 34: Methodology & Workflow **509**
CHAPTER 35: Documentation & Reporting **524**
CHAPTER 36: Policy, Ethics, & Development **538**
CONCLUSION: **549**

These contents are provided as a summary. Please use the search feature of your PDF reader to find any exact terms or phrases, as that is much more beneficial than any index or table of contents.

Order 0138471223

ABOUT THE AUTHOR

MICHAEL BAZZELL

Michael Bazzell investigated computer crimes on behalf of the government for over 20 years. During the majority of that time, he was assigned to the FBI's Cyber Crimes Task Force where he focused on various online investigations and open source intelligence (OSINT) collection. As an investigator and sworn federal officer through the U.S. Marshals Service, he was involved in numerous major criminal investigations including online child solicitation, child abduction, kidnapping, cold-case homicide, terrorist threats, and advanced computer intrusions. He has trained thousands of individuals in the use of his investigative techniques and privacy control strategies.

After leaving government work, he served as the technical advisor for the first season of the television hacker drama *Mr. Robot*. His books *OSINT Techniques* and *Extreme Privacy* are used by several government agencies as training manuals for intelligence gathering and privacy hardening. He now hosts the *Privacy, Security, and OSINT Show*, and assists individual clients in achieving ultimate privacy, both proactively and as a response to an undesired situation. More details can be found on his website at *IntelTechniques.com*.

Order 0138471223

DIGITAL EDITION PREFACE

The print versions of this tenth edition were released on January 1, 2023. At that time, we only released paperback and hardcover versions. There were several reasons for this. Most people believe it was to curb piracy, but that is not true. Yes, pirated copies of our digital releases were immediately shared to thousands of people the day after a release, but the print copies were also pirated within a month once people cut up their books and scanned all the pages. Counterfeit copies were a bigger issue than piracy. Pristine PDF files were turned into cheaply-printed paperbacks which then flooded various markets. The hassle of dealing with that was more time-consuming than the benefit of digital books. We also refused to issue a Kindle edition because of Amazon's history of deleting books from people's devices (and tracking their reading behavior). Therefore, we released print copies, which we believe served as a better reference for daily consumption at the time, especially for those taking classes based on this large text.

Later in 2023, we began a shift. We decided to stop releasing print versions of our books. We did not like the exclusivity agreements required by Amazon and their overall monopoly of the self-publishing market. We have always been bothered that an Amazon account was required to purchase a book about OSINT (the irony). We also saw their print quality decline drastically. In February of 2023, we released our first digital PDF guide about mobile devices as part of our ***Extreme Privacy*** series. We later released guides about macOS devices, Linux devices, and Firewalls. These were all successful, and each include free updates whenever needed. More details can be found at **https://inteltechniques.com/books.html**.

With this release, we are finally providing an official digital PDF version which is an exact replica of the printed tenth edition of ***OSINT Techniques***. It is designed for those who prefer a digital experience and want the ability to easily copy text or print a specific page. We respect that a digital version is much easier to access at all times than a 550-page book, and the full-color screen captures appear nicer than those in the monochrome prints.

There will be no digital updates to this version, as we want to preserve the content as originally published. This book offers a very specific glimpse of OSINT at the time of publication (2023). Even though a few techniques mentioned in the following pages are no longer 100% accurate, we believe that a lot can be learned from the tutorials presented here. However, purchase of this PDF includes access to the IntelTechniques online portal which provides free updates to the search tools, virtual machine commands, and other content. New material has likely been added since the original release of this new digital edition. If you purchased this PDF, I sincerely thank you for supporting our efforts to keep updated content flowing through our site. If you otherwise "acquired" this file, please consider purchasing a legitimate copy for yourself or someone else. Either way, I hope you gain something from this project.

MB

Order 0138471223

TENTH EDITION PREFACE

The previous (ninth) edition of this book was originally written in late 2021. In late 2022, I was asked to update this book, as it is required reading for numerous college courses, university degrees, and government training academies. I never want stale or inaccurate information being presented within training programs, so I created this special revision. In many previous editions, I only published a new version once I had at least 30% new material and 30% updated content. The recycled material was kept to a maximum of 40%. With this edition (and the 9th), I have deviated away from that rule. **I estimate that 20% of the content is brand new, 20% has been updated to reflect changes throughout 2022, and the remaining 60% is recycled from the previous edition.** Much of the ninth edition content was still applicable and only needed minor updates to reflect changes since 2021. If you have read the previous edition, you will find most of those overall strategies within this book. However, I have added many new OSINT methods which complement the original text in order to cater to those who always need accurate information. I also removed a lot of outdated content which was no longer applicable. I believe there is much new value within this updated text. **The majority of the updates are available in chapters 2, 3, 4, 5, 6, 7, 8, 23, 28, 30, 31, 32, and 33, along with the digital files which accompany them.** The other chapters each have minor updates.

This edition presents six new chapters unavailable within previous versions. These include Broadcast Streams (23), Application Programming Interfaces (28), and an entire new section containing four chapters about Data Leaks (30), Data Breaches (31), Stealer Logs (32), and Ransomware (33). I am very excited to release these new chapters, as I believe they introduce the future of OSINT analysis.

My primary goals with this new edition are three-fold. First, we need to continue our path of self-reliance which was introduced in the previous editions. This tenth edition introduces a completely rebuilt Linux OSINT virtual machine which is simpler to create, full of new features, and easier to operate. It is also much more stable than previous builds. I present all required commands to replicate my own system and offer an automated script which allows you to generate your own working environment within minutes. The Windows and Mac sections were also updated to reflect the changes offered within Linux. The online and offline search tools were updated to simplify many of the search techniques explained throughout the book. Second, I have reworked all of the various OSINT techniques presented throughout the chapters. Many of the search methods targeted toward Facebook, YouTube, Twitter, Instagram, and other online services which were presented in the previous editions have begun to fail as these companies change their products. The updated tutorials here should offer solutions. Finally, I have removed all outdated content in order to prevent frustration while attempting broken techniques.

Please consider the following technical note in regard to this book. I typically push my self-published titles through five rounds of editing. The fees associated with editing a book of this size (over 260,000 words) are substantial. This edition was put through only two rounds of editing, so I expect a few typos still exist. If you find any, consider reporting them to books@inteltechniques.com. We also want to hear when things break. My staff cannot respond directly to any emails, but we can correct future copies. The decision to restrict editing was mostly due to hard deadlines for courses, but book piracy also played a strong role. We have seen a drastic shift from most readers purchasing the book to the vast majority downloading illegal free PDF copies available a few weeks after the initial release. If you purchased this digital edition, I sincerely thank you. You represent a shrinking society. If you downloaded this book from a shady site, please consider purchasing a legitimate copy. We also offer a digital supplement to the Leaks, Breaches, & Logs section (which includes free updates). Details are at https://inteltechniques.com/books.html.

Keeping a book up to date about ways to access information on the internet is a difficult task. Websites are constantly changing or disappearing, and the techniques for collecting all possible public information from them are affected. However, new resources appear constantly, and much of this book contains new techniques which were previously not available. This book was accurate as of January 1, 2023. **If, or more likely when, you find techniques which no longer work, use the overall lessons from the entire book to push through the changes and locate your content.** Once you develop an understanding of the strategies, you will be ready to adapt. I am truly excited to introduce a new level of OSINT. ~MB

Order 0138471223

INTRODUCTION

OSINT TECHNIQUES

I taught my first Open Source Intelligence (OSINT) course in 1999 to a small group of local police chiefs in Illinois. I had not heard of the term OSINT at the time and I did not realize there was an official name for the methods I was teaching. I simply thought I was demonstrating a powerful way to use the internet as a part of everyday investigations. Most in the room had little interest in my session, as they had not experienced the internet much. Well, times sure have changed. OSINT is now quite the buzz word within several communities. Hacking conferences proudly host OSINT training sessions; cyber security groups include it within their strategies for hardening customers' systems; law enforcement agencies dedicate entire investigation units to online research into criminals; journalists rely on the methods every day; and even the social engineering crowd has adopted it as part of their playbook. I never knew OSINT would become a household name within most technology circles.

As my company continues to provide OSINT training sessions, the audiences seem to grow every year. It is no longer a course reserved for tech-savvy employees. We now see people with minimal online experience being thrown into investigation and analyst positions. We see crowds desperate for the latest investigation techniques, only to see those methods disappear without notice as social networks come and go, or change their search options. Search techniques seem to be more fickle than ever, and I am always concerned about losing an online resource.

What is OSINT?

Open Source Intelligence, often referred to as OSINT, can mean many things to many people. Officially, it is defined as any intelligence produced from publicly available information that is collected, exploited, and disseminated in a timely manner to an appropriate audience for the purpose of addressing a specific intelligence requirement. For the CIA, it may mean information obtained from foreign news broadcasts. For an attorney, it may mean data obtained from official government documents which are available to the public. For most people, it is publicly available content obtained from the internet.

Overall, this book includes several hundred sources of free information and software which could identify personal information about anyone you might be investigating. All of the resources are 100% free and open to the public, with a few minor exceptions. Each method is explained, and any creative search techniques involving the resources are detailed. When applicable, actual case examples are provided to demonstrate the possibilities within the methods. The book can be read in any order and referenced when a specific need arises. It is a guidebook of techniques that I have found successful in my investigations.

Locating this free online information is not the final step of OSINT analysis. Appropriate collection and reporting methods will be detailed and referenced. Whether the data you obtain is for an investigation, a background check, or identifying problem employees, you must document all of your findings. You cannot rely on the information being available online forever. A website may shut down or the data may be removed. You must preserve anything of interest when you find it. The free software solutions presented here will help you with that. OSINT search techniques do not apply only to websites. There are many free programs which automate the search and collection of data. These programs, as well as application programming interfaces, will be explained to assist the advanced investigator of open source intelligence. In summary, this book is to serve as a reference guide to assist you with conducting more accurate and efficient searches of open source intelligence. This is not a debate of the various opinions about online reconnaissance for personal information. It is not a historical look at OSINT or a discussion of your administrative policy. Furthermore, it is not a how-to guide for criminals to steal your identity. Nothing in this book discusses illegal methods.

Order 0138471223

Book Audience

When I first considered documenting my OSINT techniques, the plan was to post them on my website in a private area for my co-workers. This documentation quickly turned into over 250 pages of content. It had grown too big to place on my site in a manner that was easy to digest. I changed course and began putting together this book as a manual to accompany my multiple-day training sessions. It has grown to a huge textbook which could never include every beneficial resource on the internet.

Many readers work in some form of law enforcement or government agency. Police officers can use these techniques to help locate missing children or investigate human trafficking. Intelligence analysts can apply these methods to a large part of their daily work as they tackle social media posts. Detectives can use the search techniques to re-investigate cases that have gone unsolved. I embrace these techniques being used to locate facts which can help solve crimes. This book also caters to the private sector, especially security divisions of large corporations. It can help these teams locate more concise and appropriate information relative to their companies. These methods have been proven successful for employees who monitor any type of threat to their company, from physical violence to counterfeit products. I encourage the use of these techniques to institutions which are responsible for finding and eliminating "bad apples". This may be the human resources department, application processing employees, or "head hunters" looking for the best people. The information about a subject found online can provide more intelligence than any interview or reference check.

Parents and teachers are encouraged to use this book as a guide to locating social media content posted by children. In many households, the children know more about the internet than the adults. The children use this to their advantage and often hide content online. They know that it will not be located by their parents and teachers, and often post inappropriate content which can become harmful in the wrong hands. This book can empower adults and assist with identifying important personal information which could pose a threat toward children. A large portion of my intended audience is private investigators. They can use this book to find information without possessing a deep understanding of computers or the internet. Explicit descriptions and occasional screen captures will ensure that the techniques can be recreated on any computer. Several universities have adopted this book as required reading, and I am honored to play a small role in some amazing courses related to network security.

I realize that people who use these techniques for devious purposes will read this book as well. Colleagues have expressed their concern about this possibility. My decision to document these techniques came down to two thoughts. First, anyone that really wants to use this information in malicious ways will do so without this book. There is nothing in here that could not be replicated with some serious searching and time. The second thought is that getting this information out to those who will use it appropriately is worth the risk of a few people using it for the wrong reasons. Please act responsibly with this information.

Finally, a parting thought before you begin your journey through OSINT analysis and collection. This book was written as a reference guide. It does not necessarily need to be read straight-through sequentially, but it was written as a chronological guide for most investigators. I encourage you to skip around when needed or if you feel overwhelmed. The second chapter about Linux may make you want to abandon the teachings before ever utilizing an online resource or website. When you encounter material that seems too technical or not applicable, please move on to the next chapter and consider returning later. The book is suitable for all skill levels, and there is something here for everyone. You can always return to the advanced topics when more appropriate.

Digital Files

Throughout this book, I refer to several files which can be downloaded in order to simplify your usage of the various tools, techniques and scripts. These are all hosted on my website, and available free to you. As each file or script is referenced, I provide a download link to simplify the learning process. Please embrace these files as a vital part of this book. They should minimize frustration and save time as you replicate the tutorials.

Order 0138471223

SECTION I
OSINT PREPARATION

This entire section explains the essential steps which I believe any online investigator should complete before ever conducting a search online. We should never jump into an investigation without being digitally secure with a clean computer and software which has not been compromised from previous activity. We should begin each investigation with confidence, knowing we are working within an environment without any contamination from previous investigations. It will take a lot of work to create our perfect playground, but replicating a pristine environment for each investigation will be easy. Much like a DNA forensics lab must be configured and ready before an investigation, your OSINT lab should face the same scrutiny.

The first four editions of this book began with search engine techniques. Right away, I offered my methods for collecting online information from various popular and lesser-known search websites. This may have been due to my own impatience and desire to "dive in" and start finding information. This edition will begin much differently. Before you attempt any of the search methods within this book, I believe you should prepare your computing environment. I was motivated to begin with this topic after teaching a multiple-day OSINT class. On day two, several attendees brought laptop computers in order to attempt the techniques I was teaching during the course. During a break, I observed police officers searching Facebook on patrol vehicle laptops; private investigators using Windows XP while browsing suspects' blogs; and cyber security professionals looking at hacker websites without possessing any antivirus software, script blockers, or a virtual private network (VPN).

I have also been guilty of all of this. Early in my career of researching OSINT, I did not pay much attention to computer security or proper browsing habits. While I was aware of malicious software, I knew I could reinstall Windows if something really bad happened. This was reactive thinking. I believe that we must all proactively attack vulnerabilities in our own privacy and security while conducting online research. This section is not meant to be a complete guide to computer security or a manual for total privacy. Instead, I hope to quickly and efficiently propose the most beneficial strategies which will protect you from the majority of problems. Applying the changes mentioned in this section will provide a valuable layer of security to your online investigations and overall computing habits.

In the following chapters, I will explain how to ensure your computer host is secure; configure virtual machines for each investigation; embed Linux applications into Windows and Mac hosts; customize OSINT software which will be available at all times; create your own set of search tools to automate queries; prepare a virtual Android environment for mobile investigations; and easily clone all of your work for immediate replication if anything should become corrupt, damaged, or compromised. Your efforts now will pay off ten-fold in the future.

There is a lot to digest here. Please allow yourself to skip over any technical sections and return to them once you understand the overall intent of the material. Many readers skip to Section Two (Chapter Nine - Search Engines) and start practicing various online OSINT techniques. Some navigate directly to Chapter Eight to start using the free custom offline search tools. Others read through the entire book before ever touching a computer. Approach the content in the way which suits you best. Most importantly, do not let the technical aspects deter you from finding areas which will benefit your own investigations the most.

Let's get started.

CHAPTER ONE
COMPUTER OPTIMIZATION

What data is on your computer? Is there a virus, malicious software, or spyware hanging around from casual browsing in questionable places? Does your internet cache include tracking cookies from Amazon, Facebook, Google, and others? Is there evidence of your last investigation stored within your bookmarks, documents, or download queue? If the answer is "maybe" or "I don't know" to any of these, you have a contaminated computer. If your investigation enters court testimony, you may have a lot of explaining to do once an expert witness who specializes in computer forensics takes the stand. If your screen captures display evidence unrelated to your investigation, you could compromise the entire case. You may think I am being overly cautious, but we can no longer take any chances when it comes to the purity of our online evidence.

First, we need to focus on the idea of a clean host. Your host machine is your traditional physical computer. It may be the laptop or desktop owned by your agency or purchased with personal funds. It is the device with which you obtain your access to the internet, but not necessarily the direct avenue which will be used for your investigations. In the next chapter, I present my options for protection during online investigations by using a virtual machine (VM) on top of your host. Before we can consider building a VM, we must know we have a host without any contamination. In a perfect world, you have just purchased a brand-new computer and are ready to tackle your first investigation with it. There is no contamination because you have yet to turn it on. In the real world, you are stuck with used equipment, repurposed gear, and outdated operating systems. Regardless of whether you have a new machine or old, this chapter contains steps we must take before going online.

Assume you possess a laptop computer which you have had for a year or two. You installed some traditional software such as Microsoft Office and maybe added a better browser such as Firefox. You have checked your email, logged in to your Amazon account, and browsed the web as anyone else would with their own machine. Each time you visit any website, you collect detailed temporary files which are personalized only for you. They contain session details and account data. We all constantly leave a digital trail within every device we touch. Now imagine that you begin an investigation into a target on this same machine. A Google search leads you to an Amazon Wishlist. Loading to that page connects your Amazon account to the query, and your name is visible within the screen capture. Even when you log out of services such as Amazon, persistent "cookies" linger and let companies know you are still online. They follow you to websites you visit, attempting to collect your interests in order to present targeted advertisements. Your fingerprint is now all over this investigation. I pick on Amazon, but Google, Facebook, and others are much worse.

If I have not convinced you that your machine is contaminated, consider the following scenario, which happened to me many years prior to this writing. While investigating your suspect, you check your email in another browser tab. You also take advantage of various instant messenger applications in order to communicate with colleagues. You finish your investigation and submit your evidence for discovery. A suppression hearing is scheduled because the opposing party in your case wants evidence thrown out. During arguments, the other side demands an exact clone of the computer used during the investigation be provided to their own digital examiner. The judge agrees, and orders you to allow the opposing side to make an identical clone of your machine. You begin thinking about the personal activity and online purchases which are now going to surface during this trial.

We can avoid all of this. I present a firm rule which will not sit well with all readers. **You should possess a dedicated machine for the sole use of online investigations.** It should have no personal usage and no unnecessary activity. It should be a machine only used as part of your profession. Even if you only have a used computer, you can bring life back to the machine and start over. This requires much more than simply deleting files and running a computer cleaning application. These only remove the obvious data, and will never truly eliminate all contamination from the machine. To do things right, we should completely reformat and reinstall all software. **This will erase all data on your machine, so proceed with caution!** The following pages explain my recommendations for Windows, macOS, and Linux hosts, and the benefits and risks of each.

Microsoft Windows Host

Most readers will possess a Windows-based computer. This is my least favorite option for this type of work, but we must all deal with any limitations presented by our employers or our own comfort within an operating system. I was a dedicated Windows user from 3.11 to 7, but transitioned away when 10 hit the scene. If your host computer has a Windows operating system, there are several things to consider. First, which version do you have and which do you want? As I write this, I believe Windows 10 is the best version to possess. It will be supported with updates throughout October of 2025. At that time, one will be forced to move to Windows 11 or whatever Windows 12 will look like in the future. While Windows 11 will work as an OSINT machine host, you may encounter additional hurdles in order to make everything work. Let's start with an optional complete wipe of the system.

Complete System Wipe

First and foremost, backup any important data. Connect an external drive via USB and copy any documents, configuration files, and media which will be removed when you reformat the machine. Common locations include the Desktop, Downloads, and Documents folders within the home folder of the current user. Double check that you have everything you need, because the next step is to remove all data from the drive. Most modern Windows computers possess a hidden "restore" partition. To factory reset Windows 10, go to "Start" > "Settings" > "Update & Security" > "Recovery" and click the "Get started" button under "Reset this PC". Select "Remove everything", which results in the following two options:

- Just remove my files
- Remove files and clean the drive

Choose the "clean the drive" option and wait for completion. The result will be a new operating system free of any previous contamination. If you do not have this option, or possess an older Windows operating system, you will need the original installation media or a restore CD from the manufacturer. Upon boot, refuse any requests to create a Microsoft account, and only provide the necessary information to log in to Windows, such as a vague username and password. I prefer to eliminate any internet connection to this machine before I conduct this activity. This usually prevents Microsoft from demanding an online account.

If you have a new computer or are reinstalling the operating system, you will be prompted to choose "Express Settings" or "Customize Settings". Choose the custom option which will present many choices for your new system. Disable each option presented on the screen. This will disable some of the most intrusive privacy violations such as the ability to collect keystrokes as you type and sending usage data to Microsoft.

Windows 11 requires some slight modification. To factory reset Windows 11, go to "Start" > "System Settings" > "Recovery" > "Reset PC". Select the option to remove your files and choose the "Local" reset option if available. Continue through all confirmation screens and allow any restart requests. Windows 11 will be more forceful about creating an online user account, but lack of any internet connection should allow you to bypass this with enough resistance. Eventually, I suspect all versions of Windows will enforce online accounts.

Antivirus & Antimalware

There are a dozen popular antivirus companies that will provide a free Windows antivirus solution. For most Windows users, I simply recommend to use Microsoft's products. Users of Windows 7 should use Microsoft Security Essentials, while Windows 10 and 11 users should use the default Windows Defender included with their installation. Privacy enthusiasts will disagree with this advice, and I understand their stance. Microsoft products tend to collect your computer usage history and analyze the data. Unfortunately, their core operating systems also do this, and it is difficult to disable long term. Therefore, I believe that Windows users are already disclosing sensitive information to Microsoft. Using their antivirus solutions will not likely enhance the data being collected.

I believe Windows users should also add protection from malicious software, otherwise known as malware. Again, there are numerous free options from which to choose. I recommend Malwarebytes for Windows users who desire additional protection. I suggest executing, updating, and scanning at least once a week on every Windows device you use.

- Navigate to http://www.malwarebytes.com/ and select the "Free Download" option.
- Conduct a default installation.
- On a weekly basis, launch the program, update the database, and conduct a full scan.
- Malwarebytes will remove any issues it finds.

If prompted, decline any premium features or trials. The free version is sufficient and preferred. Proper antivirus and antimalware protection will greatly enhance your overall Windows computing experience. It will help your computer to run smoothly and may prevent malicious files from infecting your operating system. It will help protect the integrity of any online investigations. I refer to these steps as the "staples". They are the minimum requirements before proceeding and apply to any computer user. Ideally, you will never use your host operating system for any web browsing or investigations, and all of this will be overkill. However, it is better to be safe than sorry. Always consider the integrity of your investigations.

Telemetry

Microsoft's Telemetry service continuously collects the following data, plus numerous additional details, sending it to their corporate servers in Seattle.

- Typed text on keyboard
- Microphone transmissions
- Index of all media files on your computer
- Webcam data
- Browsing history
- Search history
- Location activity
- Health activity collected by HealthVault, Microsoft Band, and other trackers
- Privacy settings across Microsoft application ecosystem

This data would make it very easy to identify you, your location, and all online activity. Microsoft claims this collection of data is only to enhance your experience. I find this invasive, and I will present options to disable much of the data collection. I strongly advise users to attempt to minimize the amount of data Microsoft collects about your computer usage. I already explained a few options during the installation process, but there is much more content which needs blocked. There are many free utilities which assist with this, but I have found **O&O Shut Up 10** to be the most effective and current.

Download the latest version at https://www.oo-software.com/en/shutup10 then install and launch the software. You will see many individual options which can be enabled or disabled. A red icon indicates that feature is disabled while green indicates enabled. The wording can be murky. In general, anything red indicates that data about that topic is being sent to Microsoft while green indicates the service is blocked.

As an example, the first option states "Sharing of handwriting data disabled". The default option is disabled (red). Switching to green tells us that this threat is disabled, and we are protected. Some may want to play with each individual setting. Most choose a pre-determined level of privacy. In the "Actions" option at the top, you will see three categories of "Recommended", "Recommended and somewhat recommended", and "Apply all settings". The first option is very safe and applies normal blocking such as disabling advertisement IDs. The second option is a bit stricter and blocks everything except automatic Windows updates, Windows Defender, and OneDrive. The last option blocks everything possible.

My preference is to select the "Recommended and somewhat recommended" option, and then enable the "Microsoft OneDrive Disabled" option. This leaves updates and Defender running, which are vital to the overall security of your machine. After you have made your selections, close the program and allow Windows to reboot. Open the application again to make sure your desired settings were maintained. Every time you update the Windows operating system, take a look to see if you need to re-enable your choices here. If you ever have troubles because of your level of protection, you can reverse these changes any time from within the application.

System Cleaner

In some previous books, I recommended a cleaning application called CCleaner. I no longer use this product because of some unethical practices of its owner Piriform. Some versions of CCleaner contain Ad-ware which has been accused of collecting user metrics. My preference today is to use **BleachBit** (bleachbit.org). BleachBit is very similar to CCleaner, but can be a bit more aggressive. I select all available options with the exception of "Wipe Free Space". Choosing this would overwrite all free space on the hard drive which is time consuming. BleachBit removes leftover internet history content, temporary files, and many other types of unwanted data. I execute this program weekly on any Windows or Linux host I use.

Virtual Private Network (VPN)

I believe that every OSINT researcher should possess and use a virtual private network (VPN) at all times. A VPN extends a private network across a public network, such as the internet. It enables users to send and receive data across shared or public networks as if their computing devices were directly connected to the private network, thus benefiting from the functionality and security of the private network. A VPN masks your identity online. Two specific examples should help demonstrate the need for this resource.

If you are on your home computer, and connected to the internet, you are using a connection from your internet service provider (ISP). If you navigate to a website that is monitoring visitors, it knows your IP address, approximate location, and internet provider and type (cable, DSL, etc.). However, if you are on that same computer, with the same internet connection, you can use a VPN to protect you. The VPN software connects your computer to one of their servers over the internet connection. This encrypted traffic cannot be deciphered by the ISP. When your traffic gets to the VPN server, it sends and receives your data, returning incoming packets to you. The websites that you visit believe that you have the IP address of the VPN server. They do not know what type of internet connection you possess nor your location.

Some readers may wonder why they cannot simply use the free Tor service for this scenario. While you could, it is not usually advised. Tor connections can be too slow for constant use. Also, some websites will not let you access their services through a Tor proxy. Connecting to your bank through Tor will likely set off alarms, and may prevent you from access. Visiting social networks through Tor can also be an issue. I believe that Tor is great when you truly need to hide your entire connection, and I will discuss more on that later. I believe that every day browsing is better suited for a VPN.

If you work for a large corporation, you may already have access to a corporate VPN. Ask around and identify your options. These are great for security, but not so much for privacy. I never recommend a corporate VPN for online investigations. Instead, you need to purchase a VPN service. While there are a few providers that give away free VPNs, I never recommend them. They are extremely slow and often use your internet connection for other people's traffic. Instead, consider purchasing access from a reputable provider such as Proton VPN or Private Internet Access (PIA). I explain the benefits of each at **https://inteltechniques.com/vpn.html**.

VPNs can be launched in many ways. Some run through a firewall or router, which may be overkill for your needs. This is especially true if you conduct investigations from a laptop at multiple locations. Some use various web browser extensions which allow the VPN to intercept data. I do not recommend this as it would only protect your browser traffic instead of your entire host. My advice for you is to protect your entire host computer with a dedicated VPN application. This will also protect your virtual machines, which will be explained soon.

Both Proton VPN and PIA provide a software application to all premium user accounts. I find this sufficient for our needs and installation is easy for both Windows and Mac. Each of these providers allow you to connect to your choice of dozens of servers worldwide. I can choose California when I want to appear on the west coast or New York when I want to appear in the east. I can choose London in order to bypass restrictions while watching the BBC online or Toronto when I need to appear as a Canadian user. Your yearly access can be used on multiple devices simultaneously. My personal policy on VPNs is quite simple. I always use a VPN on any device that I connect to the internet. This includes desktops, laptops, and cell phones.

Lately, I have had better success with Proton VPN than PIA in regard to online investigations. PIA is one of the largest VPN providers in the world, and many sites block them due to abuse and fraud. Proton VPN is equally as secure, but not as popular. When a website blocks me because I possess an IP address from PIA, I can almost always connect to the site by switching over to Proton VPN. Proton VPN is a bit more expensive than PIA, but you may have less headaches.

What do I do? At my home, I rely on Proton VPN to protect all traffic through my pfSense firewall (explained in my book ***Extreme Privacy***). When not at home, I have the Proton VPN app ready on my laptop. Whenever I need to bypass VPN-restricted websites, I launch the PIA app on my laptop and connect to a dedicated IP address VPN. When finished with that need, I disconnect PIA and return to my Proton VPN protection. I maintain active accounts through both providers, each paid with Bitcoin in an alias name. **Overall, any reputable VPN is better than no protection.** I always display current recommended VPN affiliate purchase options on my site at **https://inteltechniques.com/vpn.html**.

Password Manager

While you conduct your online investigations, you will likely create and maintain numerous accounts and profiles across various services. Documenting the profile details, including passwords, for these services can be a daunting task. A password manager provides a secure database to store all of the various settings in regard to these profiles. My choice is KeePassXC.

KeePassXC is an open-source password manager that does not synchronize content to the internet. There are many convenient online password managers which are secure and keep all of your devices ready for automated logins. Those are great for personal security, and millions of people are safely using them. However, it is not enough for our needs. Since you will be storing data connected to online investigations, you should protect it in an offline solution. KeePassXC is cross-platform and free. It will work identically on Mac, Windows, or Linux. Download the software from https://keepassxc.org, and conduct the following as an exercise.

- Launch KeePassXC and select "Database" > "New Database".
- Provide a name to your new password database, such as Passwords.
- Move the encryptions settings slider completely to the right and click "Continue".
- Assign a secure password which you can remember but is not in use anywhere else.
- Click "Done" and select a safe location to store the database.
- Close the program and verify you can open the database with your password.

You now have a secure password manager and database ready for use on your host machine. Whenever you start using virtual machines, as explained in the next chapter, all of your passwords will be available within the host, and the VM will store no credentials. Next, assume you are ready to change the password to your covert Facebook profile. Navigate to the menu which allows change of password. Next, conduct the following within KeePassXC.

- Right-click within the right column and select "New Group".
- Name the group Facebook and click "OK".
- Select the Facebook group on the left menu.
- In the right panel, right-click and select "New Entry".

- Provide the name of your covert account, username, and URL of the site.
- Click the black dice icon to the right of the "Repeat" field.
- Click the eyeball logo underneath the black dice logo.
- Slide the password length slider to at least 40 characters.
- Copy the generated password and paste into the "Password" and "Repeat" fields.
- Change your Facebook password to this selection within your account.
- Click "OK" and save the database.

You successfully created a new, secure, randomly generated password for your covert profile. You will not remember it, but your password manager will. From this moment forward, you will change every password to any site that you access upon logging in. The next time you log in to your secure sites, change any passwords. Allow your password manager to generate a new random password containing letters, numbers, and special characters. If the website you are using allows it, choose a password length of at least 50 characters. When you need to log in, you will copy and paste from the password manager. For each site which you change a password, your password manager will generate a new, unique string. This way, WHEN the site you are using gets breached, the passwords collected will not work anywhere else. More importantly, recycled passwords will not expose your true accounts after the breached data becomes public. There should be only a handful of passwords you memorize, which brings us to the next point.

The password to open your password manager should be unique. It should be something you have never used before. It should also contain letters, numbers, and special characters. It is vital that you never forget this password, as it gives you access to all of the credentials that you do not know. I encourage users to write it down in a safe place until memorized. It is vital to make a backup of your password database. When you created a new database, you chose a name and location for the file. As you update and save this database, make a copy of the file on an encrypted USB drive. Be sure to always have a copy somewhere safe, and not on the internet. If your computer would completely crash, and you lose all of your data, you would also lose all of the new passwords you have created. This would be a huge headache. Prepare for data loss now.

I no longer recommend the KeePassXC browser extension for OSINT investigations. There is nothing insecure about it, but it requires you to copy the master database within any virtual machines unnecessarily, and technically provides some connection from one investigation to another within the VM. This is not a huge deal, but we should always be willing to scrutinize our strategies and become more secure investigators.

The attraction to online password managers such as Bitwarden is the ability to sync the password database to all devices over the internet. I understand the benefits of these features, but it also comes with risk. All reputable online password managers encrypt the passwords locally on the user's device before syncing with their own servers. Theoretically, no one at the password manager company would have the ability to see your individual passwords. However, nothing is hack-proof. It is only a matter of time before something goes wrong. For personal machines, I believe Bitwarden is a great password solution. For online investigations, I believe we should keep our data away from the internet.

By keeping your passwords in an offline database, you eliminate this entire attack surface. By keeping your password manager ready in your host machine, you will have immediate access to it regardless of which virtual machine you are using during an investigation.

Hopefully, you now have a newly-refreshed Windows machine with many protections against the threats you will face while conducting online investigations. Your system is ready for virtual machines to be built and executed, and you will be ready to defend the security of your strategies if ever questioned in court or by a client. Before we get into the real power of VMs, lets discuss similar strategies for macOS and Linux.

Apple macOS Host

At risk of sounding hypocritical, **I prefer modern macOS machines for OSINT use over Windows and Linux.** Before readers of my book *Extreme Privacy* and listeners of my related podcast scold me, please consider my reasons. I always have three computers which are part of my daily office usage. My personal machine is a Linux laptop. It is only for personal activity and is never used as part of any investigations. My OSINT machine is a MacBook Pro with an M1 Apple Silicon chip. It is blazing fast and executes virtual machines better than my Linux host. It is also quite secure by default. I never touch the host operating system, and conduct all investigations within various virtual machines. I rely on a free and open-source application called UTM for VMs, which I find to be superior to the Linux and Windows options explained within the next chapter. My third computer is devoted to breach data work, as explained within Section Three. Let's tackle macOS.

Complete System Wipe

Similar to Windows, make a backup of any valuable data. Common locations include the Desktop, Downloads, and Documents folders within the home folder of the current user. Within your Apple device, apply all updates and fully upgrade to the latest OS (currently Ventura). Once complete, conduct the following (based on M1).

- Open "System Settings" and type "reset" in the search box.
- Select "Transfer or Reset" and click "Erase All Content and Settings".
- Enter your password, click continue, confirm erase option, and allow device to reboot.
- Connect to Wi-Fi in the upper-right (if required for "Activation Lock").
- Allow activation to complete (enter account password if required).
- Click "Restart" when activation completes, then click "Get Started".
- Select desired language and region, then click "Not Now" in the Accessibility screen.
- Click "Continue" without selecting a Wi-Fi network and confirm your choice.
- Click "Continue" on the Data and Privacy screen and "Not Now" at the Migration Assistant.
- Click "Agree" to the Terms and Conditions and confirm "Agree".
- Provide a generic computer name and secure password.
- Click "Continue" then "Continue without enabling Location Services".
- Click "Don't Use" to confirm choice, then choose your desired time zone.
- Disable all analytics options and click "Continue", then click "Set Up Later" for Screentime.
- Uncheck "Enable Ask Siri" and click "Continue", then click "Set Up Touch ID Later".
- Click "Continue", then choose your desired look.

You should now have a clean computer. Next, we need to apply protection to the host. There is an untrue belief that Mac and Linux computers cannot become infected with viruses. While Mac and Linux virus infections are much rarer than on Windows, it still happens. Let's install Rosetta, then several security applications. Please note that all commands presented within this section are available at **https://inteltechniques.com/osintbook10/**. When prompted, enter the username of "**osint**" and your password of "**book1443**" (without quotes).

Rosetta

If you plan to use your M1, M2, or newer Apple computer to replicate the tutorials presented in later chapters, I highly recommend you ensure that Rosetta is installed on your system. This software, which is provided by Apple, helps applications which do not support ARM-based devices run smoothly on new macOS systems. You likely already have it installed, but you can confirm with the following command.

```
softwareupdate --install-rosetta --agree-to-license
```

Antivirus & Antimalware

Mac users do not have any built-in antivirus protection, and most do not need any. The software architecture of Mac computers is much more secure, and viruses are rare (but they do still occur). I do not recommend free commercial products such as Avast, Kaspersky, and others. They tend to be more of an annoyance than helpful, and their business practices can be questionable. However, I do believe that it is irresponsible to have absolutely no protection whatsoever. I was once asked during testimony of a federal trial to disclose any security software present within my investigation computers. I was glad my response was not "none". This would have likely introduced a defense blaming an infected workspace. I was proud to disclose my open-source solutions. When I conduct investigations from a Mac computer, I always possess three software applications which can be executed at any time without any of them running full-time in the background of my operating system. However, we must first install Homebrew. Homebrew (often referred to as Brew) is a very beneficial program when there is a need to install macOS software which would usually already be present on a Linux computer. It also happens to have pre-configured versions of the applications I am about to discuss. The easiest way to install Brew is to visit the website brew.sh and copy and paste the following command into the Terminal application ("Applications" > "Utilities" > "Terminal").

```
/bin/bash -c "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Be sure to follow the instructions presented at the end of installation. We can now use Brew to install the following host applications.

Task Explorer (objective-see.com/products/taskexplorer.html): This free Mac-only application is simple yet effective. It identifies all running processes and queries them through a service called Virus Total. If it finds a suspicious file, it alerts you with a red flag in the lower-right corner. Clicking the flag allows you to see more details about the potential threat. I execute this program weekly from any Mac machine I am using. If you have picked up a virus on your host, this program should identify it quickly. However, it does not remove any infections. For that, you will need to research any suspicious files. Install this application with the following Terminal command.

```
brew install --cask taskexplorer
```

KnockKnock (objective-see.com/products/knockknock.html): Similar to the previous option, which is maintained by the same company, this program conducts a scan of your Mac device. However, it is looking for persistent programs which are set to launch upon boot. Since most viruses inject themselves to launch the moment your computer starts, this program may identify threats which were missed by the previous program if they were not running at the time. After opening this application, click the scan button and allow the process to complete. You will receive a notification about any suspicious files. I execute this weekly along with Task Explorer. Please note that it also only notifies you of issues, and does not remove them. Install this application with the following Terminal command.

```
brew install --cask knockknock
```

ClamAV (clamav.net): ClamAV (not to be confused with the unnecessary paid option of ClamXAV) is a community-driven and open-source antivirus database, which is freely available to anyone. It usually does not score very high on "Top 10 Antivirus" websites, which are commonly paid advertisements. However, it is completely free; does not run on your system non-stop; only executes when you desire; and can be completely removed easily. Unfortunately, there is no easy software installation process, and no point-and-click application. You will need to manually update the database through a Terminal command, then scan your system. ClamAV does not remove any viruses by default, it only discloses the presence and location of suspicious files. In my use, ClamAV has never found a virus which impacted a macOS computer. Instead, it has identified numerous malicious files which target Windows machines, but were present on my system (mostly as email attachments). This notification allowed me to manually remove those files, which could prevent future infection of my

Windows virtual machines. If you have concerns about having a "naked" Apple computer with no antivirus, the following instructions will configure macOS to be better protected. Enter the following commands, hitting "Return" after each line, into the same Terminal application used previously.

```
brew analytics off
brew install clamav
sudo mkdir /usr/local/sbin
sudo chown -R `whoami`:admin /usr/local/sbin
brew link clamav
cd /opt/homebrew/etc/clamav/
cp freshclam.conf.sample freshclam.conf
sed -ie 's/^Example/#Example/g' freshclam.conf
```

These steps will install ClamAV; switch to the installation directory; make a copy of the configuration file; and then modify the configuration file to allow ClamAV to function. You are now ready to update your antivirus database and conduct a scan. Type the following commands into Terminal, striking return on your keyboard after each line.

```
freshclam -v
clamscan -r -i /
```

The first option will download all virus definition updates, and should be executed before each scan. The second option conducts a scan of the entire computer, and will only prompt you with details of found viruses. While it may appear to be dormant, it is working, and will notify you upon completion. All of these commands must be exact. Always copy and paste them from my site instead of manually typing from the book.

ClamAV may occasionally present a false-positive report of a virus. Do not panic. Research the file on the internet and identify the issues. If you receive reports of malicious files within email, simply delete those messages. Note that the above scans only SEARCH for viruses, they do not REMOVE threats. If you would like to conduct a scan and automatically remove suspicious files, you must conduct a different command. Please note this could be dangerous, and could permanently remove necessary files. I always run a scan, research the threats found, and execute the following scan ONLY if I am confident the files should be removed.

```
clamscan -i -r --remove=yes /
```

I confess I do not execute ClamAV often. A full scan can take hours and is unlikely to locate threats not found by the previous two applications. However, Task Explorer and KnockKnock do not protect against malicious applications which target Windows environments. ClamAV may find files which are malicious even if they are not a direct threat to your Mac computer. If you conduct government investigations, especially those which may result in prosecution, I believe you have an obligation to possess and execute some type of traditional antivirus software. ClamAV is a safe and reliable option. If I were still investigating federal crimes, I would conduct a complete scan of my Mac computer weekly.

Telemetry

Apple computers collect and send as much, if not more, telemetry as Windows machines. Apple insists this is for your benefit and to help create a fun and efficient computer experience. There is no application similar to O&O Shut Up for macOS. If you would like to stop Apple from collecting any details about your usage, you will need to install a software firewall application. I use a paid program called Little Snitch, but a free option called LuLu may work just as well for your needs. First, you should question whether you need this type of protection on an OSINT machine. As I stated previously, I never conduct any investigations within the macOS operating system. I only rely on virtual machines, and my activities within those machines are not seen by Apple. A firewall might be overkill for this situation, but I use one anyway.

I have two long blog posts explaining my usage of Little Snitch to protect my Apple machines. Instead of recreating those posts here, I will leave you with links to each for your own consideration. You can also simply search "telemetry" on my blog and easily get to these posts.

https://inteltechniques.com/blog/2021/08/03/minimizing-macos-telemetry/
https://inteltechniques.com/blog/2021/08/18/macos-telemetry-update/

Overall, if you are conducting OSINT investigations within your macOS operating system, I absolutely believe you should have strong telemetry protection enabled via a firewall application. If you do not conduct investigations within the operating system, and rely only on virtual machines as explained in the next chapter, then I do not believe it is as important.

System Cleaner

BleachBit technically supports macOS, but the application is not maintained and it does not function appropriately. This leaves us without a reliable system cleaner for Mac computers. Fortunately, **Privacy.sexy** (privacy.sexy) has a solution. This site aids in the creation of a custom script which will clean your system based on your desired actions. Click the "macOS" option and then navigate through the privacy and security categories, selecting the options you want to apply toward your system. The area to the right will populate the commands which clean the chosen areas. When finished, click the "Download" button at the bottom and the final script will be generated. Follow the directions to execute your script whenever desired. This script replicates the options present within system cleaners such as BleachBit, but without the application overhead and potential for undesired activity. I execute my generated script weekly. Be careful here. Some options can be quite aggressive. Research anything you do not understand, and start with minimal modifications. Since these options change often, I cannot place them here with any expectation of longevity.

Virtual Private Network (VPN)

Similar to the Windows section, you should possess a VPN application within your macOS OSINT host machine. As stated previously, I rely on Proton VPN to provide protection to my home firewall, and therefore all devices on my network. When I am not at home, I rely on the Proton VPN application within my macOS host. When I am blocked from accessing a website, I connect my PIA dedicated IP VPN as needed, and return to Proton's protection when it is no longer needed. If you skipped the previous Windows section, please return to it and visit the VPN section. I keep further details on my site at **https://inteltechniques.com/vpn.html**.

Password Manager

Also similar to Windows hosts, I recommend and use KeePassXC on macOS machines. If you skipped the previous Windows section, please return to it and visit the Password Manager section. KeePassXC functions the same on all operating systems, and the password database can be shared across all platforms. Hopefully all Apple users now possess a machine ready for OSINT investigations. Figure 1.01 displays the desired result after a KnockKnock scan.

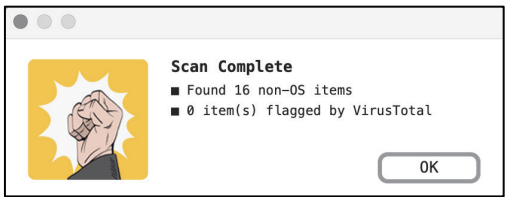


Figure 1.01: A KnockKnock scan result.

Linux Host

I suspect that very few readers will possess a Linux host machine for their OSINT investigations. Of those who do, you probably do not need my technical advice. However, let's finish this chapter with some recommendations for Linux machines.

Complete System Wipe

Most Linux systems have an embedded reset function, but I never use them. Whenever I want to reinstall a Linux operating system, I do so from a bootable USB drive which contains Ventoy. Ventoy allows you to place various Linux ISO files at the root of a USB drive and boot to it. After boot, you can select which OS you want to execute or install. A complete explanation of Ventoy is available in issue 004 of my magazine called UNREDCATED, and the direct link of https://unredactedmagazine.com/issues/004.pdf will download that entire issue. Once you have booted into your desired Linux system (I recommend Ubuntu 22.04), follow the tutorial presented within the next chapter to reinstall a fresh copy of the operating system.

Antivirus & Antimalware

Much like the Apple section, I recommend possessing ClamAV on your Linux host. This is more about "checking the boxes" of a secure machine and less about the protection from Linux-targeted viruses. You are more likely to identify viruses which target Windows machines. These could be attachments within email messages which are not a threat to your Linux installation, but should still be removed. The following commands within Terminal installs ClamAV.

```
sudo apt update
sudo apt install -y clamav clamav-daemon
```

You are now ready to update your antivirus database and conduct a scan. Type the following commands into Terminal to stop the service, update the database, and restart the service.

```
sudo systemctl stop clamav-freshclam
sudo freshclam
sudo systemctl start clamav-freshclam
```

These commands download all virus definition updates and should be executed before each scan. We now have two options for a scan of our entire drive. The first scans your data and notifies you of potential viruses. However, it does not remove any files. I always execute this option first. The second command repeats the scan while deleting any infected files.

```
clamscan -r -i /
clamscan -r -i --remove=yes /
```

ClamAV may occasionally present a false-positive report of a virus. Do not panic. Research the file on the internet and identify the issues. If you receive reports of malicious files within email, simply delete those messages.

Telemetry

Linux machines, especially Ubuntu 22.04 collect very little telemetry and allow us to easily prevent data from being sent from our machine. During installation, Ubuntu presents options to disable common telemetry, but let's assume we did not modify the proper settings. The following commands disable telemetry, and there is no harm executing these if telemetry has already been disabled.

```
sudo apt purge -y apport apport-symptoms popularity-contest ubuntu-report
whoopsie
sudo apt autoremove -y
```

We will also use this within our new Linux VMs created within the next chapter.

System Cleaner

I recommend BleachBit as my weekly Linux system cleaner. Type the following into Terminal to install the application.

```
sudo apt install bleachbit
```

Clicking the nine dots in the lower left will present two BleachBit applications. The second icon executes the software with administrative privileges and is the option I choose. Upon first launch, click "Close" to accept default configuration. Select every option except the "Free disk space" feature. Click "Preview" to see a report of recommended cleaning. Click "Clean" to execute the process. I run this program at least once a week to remove unwanted files.

Virtual Private Network (VPN)

Similar to the Windows and macOS sections, you should possess a VPN application within your Linux OSINT host machine. As stated previously, I rely on Proton VPN to provide protection to my home firewall, and therefore all devices on my network. When I am not at home, I rely on the Proton VPN application within my Linux host. When I am blocked from accessing a website, I connect my PIA dedicated IP VPN as needed, and return to Proton's protection when it is no longer needed. If you skipped the previous Windows section, please return to it and visit the VPN section. I keep details on my site at **https://inteltechniques.com/vpn.html**.

Password Manager

Also similar to Windows hosts, I recommend and use KeePassXC on Linux machines. If you skipped the previous Windows section, please return to it and visit the Password Manager section. KeePassXC functions the same on all operating systems, and the password database can be shared across all platforms.

Hopefully all Linux users now possess a machine ready for OSINT investigations. Figure 1.02 displays BleachBit after a scan.

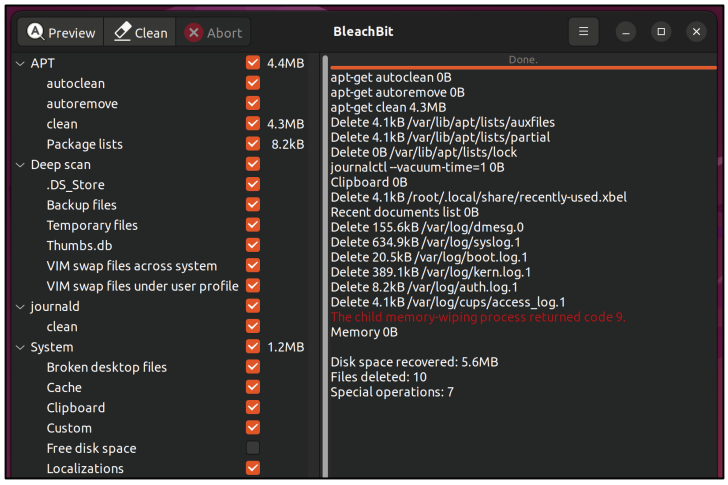


Figure 1.02: A BleachBit scan.

CHAPTER TWO
LINUX VIRTUAL MACHINES

Linux operating systems have been a part of my OSINT investigations and trainings for many years. They are lightweight, run on practically any hardware, cost nothing, and provide a level of security that cannot be obtained through traditional operating systems such as Microsoft Windows. I can navigate to any malicious website, download every virus possible, and eliminate all traces of my activity by simply deleting the VM and cloning a new machine. Upon reboot, there are no viruses and everything works exactly as intended when the system was created.

This chapter presents ways that you can harden your security by using a Linux operating system during your investigations. Many years ago, this may have been intimidating to non-technical users. Today, implementing Linux into your investigations is extremely easy. This chapter is intentionally at the beginning of the book in order to better protect you and your investigations right away. Once we start exploring the world of online search techniques, you will likely encounter malicious software or viruses at some point. If you investigate cyber criminals, this will be sooner rather than later. The malicious code will almost always target Windows machines. By choosing Linux as your investigations system, you greatly lessen the concern about infections. This chapter will vary from basic tutorials through advanced technologies. I present material on Linux before online searching because I want you to have a safe and secure environment for your research, without the fear of exposing your personal computer.

In 2015, I actively taught methods that would take a standard Linux system, such as Ubuntu or Mint, and install it to a USB device. This device could be inserted into a computer, booted, and a native Linux system would be present. When the machine was turned off, all history of that session was eliminated. This was a quick and easy way to conduct high-risk investigations while protecting the integrity of a personal computer. Unfortunately, this was slow, mostly due to the speed bottleneck of the USB device. It was a valid practice with good intentions, but not extremely applicable to most OSINT investigators today. Previous editions of this book had an entire chapter devoted to creating these USB boot devices. Today, I discourage it.

In 2016, I was contacted by David Westcott. We knew each other through our OSINT work, and he asked if I was interested in creating a custom OSINT virtual machine. I had always considered this, but had concerns about my skills at hardening Linux systems and pushing out finished builds. David had worked on other public Linux releases, and was much more comfortable distributing custom systems. I began designing my dream OSINT build, sending him weekly requests, and he began taking the ideas and executing them within a test product. By 2017, the first public version of our new operating system was released and titled Buscador (*Seeker* in Spanish).

This concept is not new. Many readers are likely familiar with Linux systems such as Kali. This is a custom Linux build which includes hundreds of security testing tools for the cyber security community. It is considered an all-in-one operating system that has everything pre-configured upon installation. We wanted that same experience for the OSINT community. Buscador was designed from the ground up with considerations for OSINT investigations. The web browsers were pre-configured with custom settings and extensions, and numerous OSINT software applications were already set-up to accept your search queries.

An important part of Buscador was the applications. On many Linux builds, launching software is not similar to traditional operating systems. While the software is installed, you must still launch a Terminal window and type the specific commands required. This can be very difficult and unforgiving. There are seldom point-and-click icons that launch similar to Windows. This has always created a barrier between the geeks and the norms. Either you know how to issue Linux commands or you do not. If you don't, then you never get to take advantage of the power of Linux and Python.

Order 0138471223

We wanted to eliminate that barrier. We wanted to make powerful Linux programs easily accessible to everyone. My initial thought was to create Bash scripts similar to batch files in Windows, but David came up with a much easier and more appropriate way. Every tool inside Buscador had its own icon in the Dock, executed by clicking with a mouse, which walked the user through the menus. After collecting the required data, each program executed the proper commands behind the scenes and delivered the content directly to the user. We believed this to be unique in our community. Every person, at any skill level, could use Buscador as a Linux virtual machine for immediate OSINT work.

Today, my views of pre-built virtual machines have changed slightly. While I am proud of our work with Buscador, I believe we should no longer rely on systems from third parties. Buscador is no longer updated with new versions, and there is no online repository to apply updates to the many applications within the virtual machine. I no longer offer a direct download link to Buscador from my site because security patches have not been applied since 2019. Many of the people using Buscador now receive errors due to outdated applications and most users do not have the training to apply their own updates in order to correct any issues. Creating a virtual machine which was user friendly had many benefits, but also some unintended consequences.

My goal in this chapter is to help you easily create and maintain your own OSINT Linux virtual machine with all of the features of Buscador. Overall, we should never rely on a single source for our OSINT tools. It is easier than ever to build your own version of Buscador while keeping it updated. By the end of this section, you will possess a custom OSINT Linux, Windows, or Mac machine which rivals any pre-built options available through various providers. Better yet, you will have built it yourself and can replicate your steps whenever desired, while being able to explain your actions.

Virtual Machines

Virtual machines (VMs) conduct virtualization of a particular computer system. They are computer operating systems on top of computer operating systems. Most commonly, a software program is executed within an operating system, and individual operating systems can launch within that program. Each virtual machine is independent from the other and the host operating system. The environment of one virtual machine has no impact on any others. Quite simply, it is a way to have numerous computers within your single computer. When finished, you can safely investigate a single target within a secure environment with no contamination from other investigations. You will be able to clone an original VM in minutes and will no longer need to worry about persistent viruses, tracking cookies, or leftover evidence.

Before creating a virtual machine, you must possess virtual machine software. There are several programs which allow you to create and execute virtual machines. Some of these are paid programs, such as VMWare. However, I will focus only on free utilities here. All recommended options will be completely free and easy to operate.

This is where I deviate heavily from previous editions. In the past, I relied solely on VirtualBox for all Windows, macOS, and Linux machines. While VirtualBox still works well for Windows and Linux, it is not recommended for Apple devices. There are two reasons, and the first is Apple silicon chips. All modern (and future) macOS machines now use Apple's own processors based on the ARM architecture. VirtualBox does not fully support this, and likely never will. Second, Apple's operating system security can be a hurdle when installing VirtualBox. With almost every application update, I need to confirm that I want the software installed; modify system security settings; and allow my system to reboot. I present a much simpler and more stable option for Apple devices within this chapter.

The following pages in this chapter will be split into "Windows & Linux" and "macOS". Each section is completely independent from the other, but I always recommend reading through all of them to know the benefits and burdens of each operating system. The overall lessons from the next section also apply to Apple devices. All commands presented throughout this chapter are available within the "linux.txt" file in your online OSINT portal at **https://inteltechniques.com/osintbook10**. When prompted, enter the username of "**osint**" and your password of "**book1443**" (without quotes).

Windows & Linux Host Virtual Machines

For Windows and Linux hosts, I still recommend **VirtualBox** (virtualbox.org). Volumes could be written about the features and abilities of VirtualBox. I will first explain how to install the application and then ways to configure a virtual machine. VirtualBox installation instructions can be found at virtualbox.org but are not usually straightforward. At the time of this writing, the following steps installed VirtualBox within Windows.

- Navigate to virtualbox.org/wiki/Downloads.
- Click the first appropriate link for your OS, such as "Windows Hosts".
- Select the highest version NOT including "Beta", such as "7.0.0".
- Download the appropriate "exe" file for Windows.
- Download the Extension Pack for your version, such as "7.0.0.vbox-extpack".
- Install the exe file with all default settings.
- Double-click the Extension Pack and install it to VirtualBox.

The following commands within Terminal installed VirtualBox on my Ubuntu Linux host.

```
sudo apt update
sudo apt install virtualbox virtualbox-ext-pack -y
```

While VirtualBox is free for personal use, make sure you or your organization meets the requirements for usage of the software, especially the free Extension Pack license. The only requirement for VirtualBox to function is a computer that supports virtualization. Any modern Windows or Linux host should work without any modification. Most mid-range and high-end computers made within the past five years should have no problem, but may require you to enable virtualization support in the BIOS (Basic Input / Output System) during startup. Netbooks, older machines, and cheap low-end computers will likely give you problems. If you are in doubt about meeting this requirement, search for your model of computer followed by "virtualization" and you should find the answers. The rest of this section will assume that your Windows or Linux host computer meets this requirement and now possesses VirtualBox.

Ubuntu Linux

This may be quite controversial to some Linux enthusiasts, but I recommend Ubuntu as an ideal Linux operating system for our OSINT machine. While I use Pop!_OS for my personal machine due to the privacy and security of that operating system, Ubuntu is more appropriate for a wider audience. Pop!_OS can present complications when migrating our commands intended for Ubuntu. If you are comfortable with Pop!_OS and insist on using it versus Ubuntu, go for it. For those who want an easier option, I will only explain the process using Ubuntu.

You may have heard of Ubuntu Linux. It is one of the most popular Linux distributions, and it is based on Debian as its backbone. Ubuntu is the product of a corporation called Canonical. Some still criticize Ubuntu for previously including Amazon software in the default downloads, which provided affiliate funding to Ubuntu when you made an online purchase, but this has been removed in current versions. The following could be replicated with other "flavors" of Linux if desired, with minimal differences within installation menus. As new versions of Ubuntu are released, you may see minor changes to the following tutorials. However, the general functions should remain the same. Before we continue, I should explain Ubuntu versioning.

As I write this, the current stable release of Ubuntu is 22.04 LTS, and my entire build is based on that release. LTS refers to Long-Term Support. This version will receive updates and patches until April of 2027. As you read this, you may see a "newer" version such as 22.10 or 23.04. While this may seem more appropriate for daily use, it is not. Versions which begin with an odd number or end in ".10" are interim releases with a short support cycle. Only versions which begin with even numbers and end in ".04" should be considered. If you are reading this after April of 2024, you will likely see a version labeled 24.04. This should be NOT used instead of 22.04,

as most, if not all, of the following tutorials may not function. We will release a new digital PDF VM supplement once Ubuntu 24.04 LTS is released. **Always use the latest 22.04 LTS release for these tutorials**.

First, we need to download an ISO file containing the installation files for Ubuntu. The official website is very user-friendly, and can be reached at ubuntu.com/download/desktop. This presents a Long-Term Support (LTS) option, which is currently 22.04.1. Clicking the "Download" button should prompt you to download the ISO file required to install Ubuntu. This will be a large file with an extension of iso. During this writing, my file was titled "ubuntu-22.04.1-desktop-amd64.iso". This file behaves similarly to a physical CD which would install Windows or any other operating system. The 64-bit version should apply to most readers and their computers. If you know you need a 32-bit version for an older computer, you will find it in the "Alternative Downloads" section of the site. Save the file to your Desktop or anywhere else to which you have easy access. Expect an approximate file size of 4 GB. Next, open VirtualBox and click on the button labeled "New". The following steps create a VM appropriate for our needs.

- Provide a name of "OSINT Original".
- Choose your desired location to save the machine on your host (I chose my Documents).
- Select "Linux" as type, "Ubuntu 22.04 (64-bit)" as version, and click "Next".
- In the Memory size window, move the slider to select 50% of your system memory.
- In the Processors window, move the slider to select 50% of your resources.
- Click "Next" and then "Create".
- Leave the hard disk file type as "VDI" and click "Next".
- Select the default option of "Dynamically allocated" and click "Next".
- Choose the desired size of your virtual hard drive. If you have a large internal drive, 50 GB should be sufficient. If you are limited, you may need to decrease that number.
- Click "Create".

Your VM has been created, but it will do nothing upon launch. We need to tell it to boot from the ISO file which we downloaded previously. Select your new machine in the menu to the left and complete the following.

- Click the Settings icon.
- Click the Storage icon.
- Click the CD icon which displays "Empty" in the left menu.
- Click the small blue circle to the far right in the "Optical Drive" option.
- Select "Choose a disk file".
- Select the Ubuntu ISO previously downloaded then click "Open".
- Click "OK" and then "Start" in the main menu.
- If prompted, confirm your choice by clicking "Start".

Your Ubuntu installation process should now start within a new window. You should be booting to the ISO file previously downloaded, which is behaving as if you had placed an Ubuntu install CD into the virtual computer. This is your first virtual machine running on top of your host operating system. If the window is too small, click "View", "Virtual Screen", then "Scale to 200%" within the VirtualBox menu. Avoid this step if possible, as it may cause viewing issues later. Consider reversing this setting once you are finished if things appear large. We can now finish the installation of your Ubuntu OS with the following steps within VirtualBox.

- Select "Install Ubuntu", select your desired language and location, then click "Continue".
- Select "Normal Installation", "Download Updates", and "Install third party…".
- Click "Continue".
- Select "Erase disk and install Ubuntu", then "Install Now". Confirm with "Continue".
- Choose your desired time zone and click "Continue".
- Enter a name, username, computer name, and password of "osint" (without quotes and lowercase) for each field. **This is a mandatory step as part of the tutorials and scripts presented in this book.**

While any other username could work at first, you will face issues as you proceed. Please make sure the **username and computer name is "osint".**

- Since this is a virtual machine inside a secure computer, minimal security is acceptable.
- Choose "Log in to automatically" and click "Continue".
- Allow Ubuntu to complete the installation and choose "Restart Now".
- Press "Enter" when prompted.

Your device should now boot to the login screen. In my example, it booted directly to the Ubuntu Desktop. The following will finish the default configuration.

- Click "Skip" then "Next".
- Select "No" and then "Next" when asked to help improve Ubuntu.
- Click "Next" then "Done" to remove the welcome screen.
- If prompted to install updates, click "Remind me later".

You now have a functioning virtual machine which contains the basic programs we need to use the internet. By default, it is using your host computer's internet connection, and taking advantage of your host's VPN if you have it connected. Technically, we could start using this machine right away, but the experience would get frustrating. We need to take some additional steps to configure the device for optimum usage. The first step should be to install VirtualBox's Guest Additions software. This will allow us to take advantage of better screen resolution and other conveniences. Conduct the following steps.

- In the VirtualBox Menu, select "Devices" > "Insert Guest Additions CD Image".
- Click "Run" when the dialogue box pops up.
- If no dialogue is present, click the CD in the dock, then right-click "autorun.sh".
- Choose "Run as Program".
- Provide your password (osint) when prompted.
- Allow the process to complete, press Return, and restart the VM (upper-right menu).

You should now have VirtualBox Guest Additions installed. You can test this by resizing the screen. If you make the Ubuntu VM full screen, you should see the overall screen resolution change with it. If you previously changed the scaling of the window in VirtualBox previously, change it back to 100%. If the screen is still too small, right-click the Ubuntu desktop and choose "Display Settings". If the option is present, choose a "scale" of 200% and see if that works better for you. Overall, I would rather modify the resolution within the Ubuntu VM and not within the VirtualBox settings. I like the Ubuntu VM to possess maximum resolution possible, in full screen mode, with a choice of the higher scale if required.

If all appears to be functioning, you can right-click the CD icon in the left Dock and choose "Eject". If not, double-click the CD icon and choose "Run Software" in the upper right corner to repeat the process. Next, we should make some modifications within the VirtualBox program in order to experience better functionality. Shut down the Ubuntu VM by clicking on the down arrow in the upper right and choosing the power button, followed by "Shut down". In VirtualBox, select your Ubuntu VM and click the "Settings" icon. Next, conduct the following steps.

- In the "General" icon, click on the "Advanced" tab.
- Change "Shared clipboard" and "Drag n' Drop" to "Bidirectional".
- In the "Display" icon, change the "Video Memory" to the maximum.
- In the "Shared Folders" icon, click the green "+".
- Click the drop-down menu under "Folder Path" and select "Other".
- Choose a desired folder on your host to share data back and forth.
- Select the "Auto-mount" option and then "OK".
- Click "OK" to close the settings window.
- Restart your Ubuntu VM.

- Click the nine dots in the lower-left to open the "Applications" menu. Search "Terminal" and open the application. In it, type `sudo adduser osint vboxsf` and press enter. When prompted, provide your password, which is not revealed in the window as you type, and press enter.

You should now have a more robust display, copy and paste capabilities, and a new shared folder connecting your VM to your host on the Desktop. You can copy files from your VM directly to your host and vice versa. This has improved a lot of function, and now it is time to personalize the machine. Personally, I do not like the default colors and wallpaper of Ubuntu, and prefer something more professional. Ubuntu 22 removed the ability to easily change wallpaper to a solid color, so we will do it all through Terminal. I conducted the following on my new VM.

- Launch Terminal from the Applications menu.
- Enter `gsettings set org.gnome.desktop.background picture-uri ''` and press return to remove the background image.
- Enter `gsettings set org.gnome.desktop.background primary-color 'rgb(66, 81, 100)'` and press return to set a neutral background color.

Next, we should consider some of the privacy and security settings within Ubuntu. The first two Terminal commands disable Ubuntu's crash reporting and usage statistics while the remaining steps within Ubuntu's operating system harden our overall privacy and security.

- `sudo apt purge -y apport apport-symptoms popularity-contest ubuntu-report whoopsie`
- `sudo apt autoremove -y`
- Launch "Settings" from the Applications Menu.
- Click "Notifications" and disable both options.
- Click the "Privacy" option, then click "Screen" and disable all options.
- Click "File History & Trash", then disable any options.
- Click "Diagnostics", then change to "Never".
- Click the back arrow and click "Power", changing "Blank Screen" to "Never".
- Click "Automatic Suspend" and disable the feature.
- Close all Settings windows.

It is important to keep the software on this original VM updated. There are different ways to do this, but I will focus on the easiest way within the operating system applications. While we do this, it may be a good time to add some commonly used applications to our Dock. Conduct the following steps.

- Launch the Applications menu (nine dots in lower-left).
- Type Terminal into the search field.
- Right-click on the application and select "Add to Favorites".
- Type Software into the search field and right-click on "Software Updater".
- Select "Add to Favorites".
- Press escape until all windows are gone.
- Launch the Software Updater icon from the Dock; click "Install Now"; and update all options.

You now have the Terminal and Software Updater in your Dock for easy access. Check for updates weekly and keep your original copy ready for usage. This brings us to a conversation about the term "Original". Ideally, you will keep a copy of this VM clean and free of any internet usage or contamination. There are two ways to achieve this, and both have unique benefits. First, let's discuss Snapshots.

Snapshots

A great feature of virtual machines is the use of Snapshots. These "frozen" moments in time allow you to revert to an original configuration or preserve an optimal setup. Most users install the virtual machine as previously detailed, and then immediately create a snapshot of the unused environment. When your virtual machine eventually becomes contaminated with remnants of other investigations, or you accidentally remove or break a feature, you can simply revert to the previously created snapshot and eliminate the need to ever reinstall. Consider how you might use snapshots, as detailed in the following pages.

Upon creation of a new Ubuntu virtual machine, apply all updates as previously mentioned. Completely shut down the machine and open the Snapshots option within your virtual machine software. Create a new snapshot and title it "Original". Use this machine for a single investigation, and export all evidence to an external USB device, such as a flash drive. You can then "restore" the Original snapshot, and it overwrites any changes made during the previous investigation. Upon reboot, all history and evidence is eliminated. This ensures that you never contaminate one investigation with another. When there are substantial updates available for Ubuntu, you can load the default configuration, and apply all updates. You can then shut the machine down completely and delete the Original snapshot, without saving it, and create a new snapshot titled Original. This new snapshot possesses all of the updates. If using this technique, I usually delete and create a new snapshot weekly. The use of snapshots is very similar between VirtualBox and VMWare, but let's take a look at the minor differences.

Snapshots (VirtualBox): Completely shut down the Virtual Machine and conduct the following.

- In the VirtualBox Menu, click on the "Snapshots" button in the upper right.
- Click on the blue camera icon to "take a snapshot".
- Create a name to remind you of the state of the machine, such as "New Install", and click "OK".

You can now use your virtual machine as normal. If you ever want to revert to the exact state of the machine that existed at the time of the snapshot, follow these instructions:

- Completely shut down the Virtual Machine.
- In the VirtualBox Menu, click on "Snapshots" and select the desired snapshot to apply.
- Click on the blue camera icon with arrow to "restore snapshot".
- Deny the option to save the current data, and click "Restore".

Optionally, if you ever want to remove a snapshot, simply click the icon with a red "X". This will remove data files to eliminate wasted space, but you cannot restore to that image once removed. It will not impact the current machine state. Many users remove old, redundant snapshots after creating newer clean machines. Today, I rarely use snapshots, as I believe they are prone to user error. I much prefer Cloned machines. These require more disk space, but provide a greater level of usability between investigations. Examples are explained later.

VM Exports and Clones

If you ever want to preserve a specific state of Ubuntu, you can export an entire session. This may be important if you are preserving your work environment for court purposes. When I am conducting an investigation that may go to trial, or discovery of evidence will be required, I make an exact copy of the operating system used during the investigation. At the end of my work, I shut down the machine. I click on "File" and then "Export" within my virtual machine software and create a copy of the entire operating system exactly as it appeared at shutdown. This file can be imported later and examined. After a successful export, I restore my clean "Original" snapshot and I am ready for the next case. The exported file is added to my digital evidence on an external drive. I now know that I can defend any scrutiny by recreating the exact environment during the original examination. As stated previously, I prefer Clones over Snapshots. I create an exact replica of my original VM for every investigation, and never use Snapshots within these unique VMs. For clarity, consider my routine for every OSINT investigation I conduct, which takes advantage of the "Clone" option within VirtualBox.

- Launch the Original VM weekly to apply updates or global changes, then close the VM.
- In the VirtualBox menu, right-click on the Original VM and select "Clone".
- Create a new name such as Case #19-87445 and click "Continue" (or "Next") then "Clone".

This creates an identical copy of the VM ready for your investigation. You have no worries of contaminating your original VM. You can keep this clone available for as long as needed while you continue to work within several investigations. You can export the clone in order to preserve evidence, or delete it when finished. Neither action touches the original copy. It is similar to possessing a new computer for every case and having all of them at your disposal whenever you want to jump back into an investigation.

Issues

I wish I could say that every reader will be able to easily build virtual machines on any computer. This is simply not the case. While most computers are capable of virtual machine usage, many demand slight modifications in order to allow virtualization. Let's take a look at the most common errors presented by VirtualBox.

VT-x is disabled: Any version of this error is the most common reason your VMs will not start. This indicates that the processor of your computer either does not support virtualization or the feature is not enabled. The fix for this varies by brand of machine and processor. Immediately after the computer is turned on, before the operating system starts, enter the BIOS of the machine. This is usually accomplished by pressing delete, F2, F10, or another designated key right away until a BIOS menu appears. Once in the BIOS, you can navigate through the menu via keyboard. With many Intel processors, you can open the "Advanced" tab and set the "Virtualization (VT-x)" to "Enable". For AMD processors, open the "M.I.T." tab, "Advanced Frequency" Settings, "Advanced Core" settings, and then set the "SVM Mode" to "Enable". If none of these options appear, conduct an online search of the model of your computer followed by "virtualization" for instructions.

VT-x is not available: This is usually isolated to Windows 10 machines. Navigate to the Windows Control Panel and open "Programs and Features". Click "Turn Windows features on or off" and uncheck all "Hyper-V" features. Click "OK" and reboot. If the Hyper-V option is not enabled, enable Hyper-V, restart the computer, disable Hyper-V, and reboot again. Attempt to start your VM with these new settings. This may seem backwards, but it makes sense. Previous versions of VirtualBox cannot run if you are using "Hyper-V" in Windows. Basically, both systems try to get exclusive access to the virtualization capabilities of the processor. Hyper-V within Windows receives the access first and impedes VirtualBox from the capabilities. The latest version of VirtualBox attempts to correct this. If the previous setting did not help, try to re-enable all of the Hyper-V options within Windows, reboot, and try to boot your VM again. If you are still experiencing problems, read the troubleshooting chapter of the VirtualBox manual at virtualbox.org/manual/ch12.html. Expand any errors received and search the provided error codes to identify further solutions.

VirtualBox Displays: Some readers of previous editions have reported the inability to resize VM windows within VirtualBox and the "Auto-resize Guest Display" menu option greyed out. The following commands within Terminal of the Linux VM should repair this issue. There is no harm running these if you are unsure.

```
sudo apt update
sudo apt install -y build-essential dkms gcc make perl
sudo rcvboxadd setup
reboot
```

Virtual Machine Size & Shrinking

If you are following my protocol, you keep your OSINT Original Ubuntu VM updated and ready to be cloned. You may notice the size of this VM will be much larger than the actual data within it. This is because of the way Linux VMs resize as they are used. If you have a 10 GB brand new virtual machine, and then apply 1 GB of updates to it, the size of the VM might be 11 GB. However, the size of the Ubuntu operating system may only be 10.1 GB. This is because the 1 GB of updates replaced several files, which only cause a 100 MB growth. However, the virtual machine sees all 1 GB of that update as "new" data even though it is not being used. Because of this, I often reduce or "shrink" my virtual machines. My protocol follows.

The following only applies to VirtualBox VMs!

First, I open my Original OSINT VM and apply all updates, especially those explained within Chapter Four. Once I reboot the VM and it is in a dormant state, I open Terminal within the Ubuntu VM and execute the following commands.

```
cd ~/Downloads
dd if=/dev/zero of=zerofillfile bs=1M
rm zerofillfile
```

These commands change to the Downloads folder; fill all "empty" space on the VM drive with a "zero file"; and then delete the file from the VM. It may take a few minutes to complete each step. I then shut down the VM and execute the following commands within either Command Prompt (Windows) or Terminal (Linux) within the HOST operating system.

```
VBoxManage modifyhd --compact (path to VDI file)
```

You will need to replace the (`path`) portion with the absolute path to the VDI file of your VM. It is easiest to drag and drop the file from the Files (Linux) or Explorer (Windows) viewer into Terminal (Linux) or Command Prompt (Windows). This will shrink the VM to the actual amount of data used. I have had VMs grow to 100 GB in size, and then shrink down to 20 GB. This can save valuable space of your drive.

Your original VM should only be used to install new software and apply updates. Throughout the next several chapters, I will reveal more about my usage protocols. We will install several applications, scripts, shortcuts, and customizations. We will often apply updates to our primary VM. When we discuss maintenance of your Original OSINT VM, please remember these shrinking tips.

We have a long way to go before we are ready to conduct an actual investigation, and we want to make sure our working environment is pristine. Don't worry about anything you have done to your new VM just yet, in a moment we are going to delete everything and start over with abbreviated steps to make sure we all have the same machine.

Hopefully, you now have VirtualBox installed on your Windows or Linux host and an Ubuntu installation created as a virtual machine. You have chosen to either use Snapshots or Clones as part of your investigation (I prefer Clones). Before we get into custom software, let's tackle Apple devices and operating systems.

Apple Host Virtual Machines

A new Mac computer with an Apple M1, M2, or later processor may cause a lot of virtualization frustration and confusion. This is due to the conflicts between the chip and the VM operating systems, and sometimes the VM software itself. **I no longer recommend any macOS users rely on VirtualBox.** This is for two reasons. First, VirtualBox does not officially support M1, M2, and newer computers, and full compatibility could take years, if ever at all. Second, VirtualBox issues are common even within Intel-based Apple computers. Installation and updates often require additional authorizations and reboots, and the software just does not run as well as it does on Windows and Linux machines. However, none of these matter much to me anymore. Apple machine owners have something better.

UTM (mac.getutm.app) is a free open-source virtualization and emulation program which allows macOS users to launch practically any virtual system within machines which have either Intel or ARM (Apple) processors. This means it will work with any Apple computer, regardless of the hardware. UTM employs Apple's Hypervisor virtualization framework to run ARM operating systems on Apple Silicon at near native speeds. On Intel-based machines, traditional x86/x64 operating systems can be virtualized. In addition, lower performance emulation is available to run x86/x64 on Apple Silicon as well as ARM64 on Intel. This allows us practically any option desired, and is unique to this program. Even the paid alternatives do not offer all of these features. The software relies on QEMU, which has always been otherwise difficult to configure. I now rely solely on UTM for all VMs on my macOS machine, and I find it to be superior to a Windows or Linux host. If you have Brew installed, the following Terminal command downloads and installs UTM.

```
brew install --cask utm
```

Upon opening UTM, you have the option to "Create a New Virtual Machine". If this is ever not visible, you can replicate the action by clicking "File" > "New" within the program's menu. Choosing this selection presents options which may be new to readers. You can select to either "Virtualize" or "Emulate" your new VM. We should understand the difference.

Virtualize: This process is accomplished with the help of hardware, typically the hypervisor. It virtually shares the hardware resources of a single physical computer into multiple virtual devices by allocating dedicated resources from the host system to the newly created virtual system. This is typically much faster than emulation and the option we will choose for our new VM. If you have an M1, M2, or newer Apple processor, you cannot virtualize x86/x64 operating systems, you can only virtualize ARM-based systems. Similarly, you cannot virtualize ARM-based systems from a x86/x64 machine. When using virtualization, the operating system must be created for the processor present within the device.

Emulate: This process is much more forgiving, but can be slow. It uses software to emulate specific hardware. This means that the VM needs a software interpreter translating its code into the host system's language. This eats up a lot of resources and can make things drag. Since the VM does not run on the host's physical hardware, emulation is slower when compared to virtualization. By contrast, in virtualization, the guest system gets direct access to the host's allocated resources, resulting in better speed. It is great to have this option, but we will not use it within this chapter.

Let's build our first Ubuntu Linux VM within UTM together. Choose the "Virtualize" option and then select "Linux". This brings us to our need to choose the proper path based on our hardware. You must choose the appropriate version of Ubuntu for your processor. If you have an Intel device, you can use the same Ubuntu ISO which we may have previously downloaded. If you have an M1, M2, or later processor, you need the ARM version of Ubuntu. You can click the Apple logo in the upper-left of your device and select "About This Mac" to identify your version. Below are the current download links for each.

Ubuntu 22.04 Desktop LTS for Intel Devices: https://ubuntu.com/download
Ubuntu 22.04 Desktop Daily for ARM Devices: https://cdimage.ubuntu.com/jammy/daily-live/current/

If choosing the ARM version, make sure you do not accidentally select the AMD version. Let's talk about these versions a bit more before we proceed. The Ubuntu 22.04 Desktop LTS is the same ISO we used previously for Windows and Linux hosts. It is the long-term version with support until 2027. It is the easy choice for Intel-based machines. Apple M1, M2, and newer machines are tricky. They require an ARM version of Ubuntu, but Ubuntu does not currently offer an official ARM LTS release. However, they do support a daily ARM build which began with the official 22.04 LTS release. We could download the LTS Server edition and modify it for our needs, but I find that unnecessary. The ARM Desktop daily build, based on the current LTS version works well for me. However, I would never use the daily versions of upcoming releases such as 22.10, 23.04, etc. Those tend to have more bugs and issues. If you are reading this before April of 2024, you should download the daily ARM build of 22.04. In 2024, hopefully we will have an official LTS ARM version.

Once you have downloaded the appropriate version of Ubuntu for your system, click the "Browse" button within UTM and select the downloaded ISO file. Leaving the "Use Apple Virtualization" option unchecked is the safe and stable way to go. However, M1, M2, or newer devices may function better in the future with this option selected. Leave it unchecked for now, as you can always build another machine with this option later to compare the performance. It was not functioning with Linux during this writing. Click "Continue".

Much like the tutorials for Windows and Linux, choose half of your system's memory and CPU cores. If you had 16 GB of RAM and an eight-core processor, you would change the RAM to "8192" and the CPU Cores to "4". Never leave the cores as "Default", as it can confuse the operating system. Click "Continue" when complete. The size of the drive should be set to the maximum you will ever need. This is not the size of the VM as it grows, it is only the max. I set mine to "100" GB.

I prefer to enable file sharing, as it makes it easier to extract evidence from your investigation onto your host. Browse to your desired shared folder (I chose Downloads) and click "Continue". Finish through the process and click the arrow icon to start your new Linux VM. You are now ready to install Ubuntu with the following steps. Upon initial boot of your new VM within UTM, conduct the following steps.

Ubuntu 22.04 LTS Desktop Only:

- Select "Try or Install Ubuntu".
- Select "Install Ubuntu", select your desired language and location, then click "Continue".

Ubuntu 22.04 ARM Daily Desktop Only:

- Select "Try or Install Ubuntu".
- Enter the username of "ubuntu" and password of "password" if prompted.
- Double-click "Install Ubuntu 22.04.1 LTS" on the Desktop.
- Choose your language and keyboard layout and click "Continue".

All Ubuntu Installations:

- Select "Normal Installation", "Download Updates", and "Install third party…".
- Click "Continue".
- Select "Erase disk and install Ubuntu", then "Install Now". Confirm with "Continue".
- Choose your desired time zone and click "Continue".
- Enter a name, username, computer name, and password of "osint" (without quotes and lowercase) for each field. **This is a mandatory step as part of the tutorials and scripts presented in this book.** While any other username could work at first, you will face issues as you proceed. Please make sure the **username and computer name is "osint".**
- Since this is a virtual machine inside a secure computer, minimal security is acceptable.
- Choose "Log in automatically" and click "Continue".
- Allow Ubuntu to complete the installation and choose "Restart Now".

- Click the "CD" icon in the upper-right of the UTM VM window; highlight the CD/DVD option, and click "Eject". THEN press "Enter" to restart the VM or the "left triangle" within UTM.
- If necessary, choose the "power" or "restart" icons in the UTM window.

Your device should now boot to the login screen. If it boots into the installation ISO again, shut the machine down. In the main UTM window, select your new machine and click the settings icon in the upper-right. Under "Drives", identify the CD/DVD drive and change "Image Type" to "None". Reboot the VM and boot into the Ubuntu Desktop. The following will finish the default configuration.

- Click "Skip" then "Next".
- Select "No" and then "Next" when asked to help improve Ubuntu.
- Click "Next" then "Done" to remove the welcome screen.
- If prompted to install updates, click "Remind me later".
- Continue to the section titled "Ubuntu Customization".

Ubuntu Customization

You should now have an Ubuntu virtual machine executed within your Intel or ARM macOS device. Unlike the steps with VirtualBox, UTM does not require any special extension packs or software to be installed within the VM for the display to resize. You should be able to adjust your screen as desired, or make it full screen for best view. However, do not enter full-screen mode just yet. We still have some work to do and I want you to have easy access to the menu bar. Remember to always use the highest resolution available within Ubuntu and then choose the 100%, 200%, or additional options present within the Ubuntu display preferences. You can right-click the Ubuntu desktop and choose "Display Settings" to see this menu. Also, there are no concerning license restrictions. This is a much better scenario for all of us.

Some desired capabilities, such as clipboard and file sharing, require the installation of UTM's Spice daemon. Launch Terminal from the Applications menu within Ubuntu, and execute the following, which will also modify our background to match the previous tutorial. You may already have the required software, but let's make sure.

```
sudo apt install spice-vdagent spice-webdavd
gsettings set org.gnome.desktop.background picture-uri ''
gsettings set org.gnome.desktop.background primary-color 'rgb(66, 81, 100)'
```

Copy and paste capabilities should now be working, but you may not see your shared folder within Files on Ubuntu. The following should fix this.

- Shut down the VM and close the window.
- Within the main UTM window, click the "Settings" icon in the upper-right for your VM.
- Select "Sharing" on the left.
- Change "Directory Share Mode" to "Spice WebDAV" and click "Save".
- Reboot the VM.
- Click the "Shared folder" icon in the upper right of the UTM Ubuntu VM window.
- Confirm your desired shared folder location.
- Open Files within Ubuntu and click the "Other Locations" option in the left menu.
- You should see a folder titled "Spice client".
- Single click that folder and wait for your system to recognize the share.
- Confirm you can access the shared folder within the left menu of Files.

Sometimes, it can take several minutes for the share to become available to Ubuntu. Once it does, it should appear until the VM reboots. Whenever you need to access the shared folder, simply click the Spice folder under "Other Locations" for that session. You can now copy files from your VM directly to your host and vice versa. I typically only do this after the VM has been booted for a while and I need the shared folder.

Next, we should consider the same privacy and security settings within Ubuntu which were discussed in the previous pages for Windows and Linux hosts. The first two Terminal commands disable Ubuntu's crash reporting and usage statistics while the remaining steps within Ubuntu's operating system harden our overall privacy and security.

- `sudo apt purge -y apport apport-symptoms popularity-contest ubuntu-report whoopsie`
- `sudo apt autoremove -y`
- Launch "Settings" from the Applications Menu.
- Click "Notifications" and disable both options.
- Click the "Privacy" option, then click "Screen" and disable all options.
- Click "File History & Trash", then disable any options.
- Click "Diagnostics", then change to "Never".
- Click the back arrow and click "Power", changing "Blank Screen" to "Never".
- Click "Automatic Suspend" and disable the feature.
- Close all Settings windows.

It is important to keep the software on this original VM updated. There are different ways to do this, but I will focus on the easiest way within the operating system applications. While we do this, it may be a good time to add some commonly used applications to our Dock. Conduct the following steps.

- Launch the Applications menu (nine dots in lower-left).
- Type Terminal into the search field.
- Right-click on the application and select "Add to Favorites".
- Type Software into the search field and right-click on "Software Updater".
- Select "Add to Favorites".
- Press escape until all windows are gone.
- Launch the Software Updater icon from the Dock; click "Install Now"; and update all options.

USB Connection

UTM offers simple USB device access. While any USB drive is attached to the host computer, select the "USB Devices" icon in the upper-right of the UTM VM window. This presents that drive within the dock of Ubuntu for file transfer. If you plug in a USB device while the VM is running, it should prompt you to choose where you want it used.

Snapshots

UTM offers a Terminal-based snapshot option, but it is not similar to VirtualBox's offering. Personally, I never use snapshots and always rely on clones, as previously explained and further detailed next.

VM Exports and Clones

While a VM is shut down and the VM window is closed, you will see options within the main UTM application for the selected VM in the upper-right. We will focus on the "Clone selected VM" and "Share" menu options. Clicking the clone option simply presents an option to make a full clone. This allows us to preserve our original OSINT VM and make copies whenever we want to conduct an investigation, as previously explained. The share option allows us to export an entire VM within a ".utm" file for easy archiving or sharing. The dialogue will

prompt you to choose the export location. If you ever want to see the location of all VMs within UTM, navigate to the following directory within the macOS host.

/Users/[your macOS username]/Library/Containers/UTM/Data/Documents/

If this location is not convenient, and I believe it is not, you can move your VM to any folder desired with the "Move selected VM" option within the main windows. I keep all of mine within a dedicated folder on my macOS host. Note that you can only move them once using the button option.

Issues

Unlike using VirtualBox within various Windows and Linux machines, UTM within a macOS system has very few issues. The software was designed only for macOS which reduces some of the typical cross-platform bugs. Since it is only made for Apple hardware, it seems much more fluid than other software.

Virtual Machine Size & Shrinking

Similar to VirtualBox, your VMs will keep growing unnecessarily. Fortunately, UTM makes the shrinking process easy. While a VM is shut down, select it within the main UTM window. Go to the Settings for that VM and select the IDE drive of your operating system within the left menu. Click the "Reclaim Space" button and confirm the option. You will see the size decrease if free space was available. I do this after every update to my original OSINT VM.

macOS Virtual Machines

If you own an Apple computer with an M1, M2, or later Apple processor, you can easily run multiple macOS virtual machines within UTM. This provides a "test" copy of macOS for any experimentation without modifying your host operating system. The following applies only to Apple processors. First, you must download the appropriate "IPSW" file from a legitimate Apple source. I prefer to visit the following website and download the latest "Final" IPSW file, which was UniversalMac_13.0_22A380_Restore.ipsw at the time of this writing.

https://mrmacintosh.com/apple-silicon-m1-full-macos-restore-ipsw-firmware-files-database/

The source of the file was as follows.

https://updates.cdn-apple.com/2022FallFCS/fullrestores/012-92188/2C38BCD1-2BFF-4A10-B358-94E8E28BE805/UniversalMac_13.0_22A380_Restore.ipsw

Once you have the IPSW file on your machine, you are ready to continue with installation.

- Launch UTM and click "Create a New Virtual Machine".
- Choose "Virtualize" and then "macOS 12+".
- Click "Browse" and select your downloaded IPSW file.
- Modify the RAM to 50 % of your current resources.
- Modify the CPU to half of the "Performance" cores. My 10-core M1 Pro has 8 performance cores, so I chose "4" for this option.
- Click "Continue" and specify the desired size of the drive.
- Click "Continue" then "Save".
- Click the "Settings" icon and click the "Display" menu.
- Choose the display of your device, enabling "HiDPI".
- Click "Save" and exit the settings.

If you want a shared drive for accessing files across each OS, complete the following.

- From the macOS host, open "System Settings" > "General" > "Sharing".
- Enable "File Sharing" and click the information button.
- Click the "+", browse to the desired shared folder, and click "Add" then "Done".
- In the UTM program, open the settings for the macOS VM.
- Ensure "Network" is set to "Shared Network" and click "Save".
- In the UTM VM settings, click the shared directory menu and select "Browse".
- Select the same folder previously shared and click "Open".
- Launch the macOS VM and open Finder.
- Go to "Finder" > "Settings" > "Sidebar" and enable "Connected Servers".
- Click "Network" within Finder and double-click the host.
- Click "Connect As…" and enter your host OS credentials.
- Enable "Remember this password…" and click "Connect".

Once complete, you have a fully-functioning macOS VM which should be as smooth and responsive as the native host. You have the ability to copy files to and from the VM when needed. It could be used to test software before committing on your macOS host or to play with the macOS OSINT machine explained in Chapter Six. Note that the drive for your macOS VM will be 100% full. Whatever size you set will create a file that size for the VM. There is currently no shrinking option. UTM can also host Windows VMs, but they must be based on ARM builds if you want them to load in M1 or newer processors. Windows VMs within Windows, Linux, or older macOS machines was previously explained in Chapter Two.

Summary

The most vital virtual machine decision to make is choosing the virtualization software. This will be dependent on your hardware. Below is my summary.

- Linux Host Computers: VirtualBox
- Windows Host Computers: VirtualBox
- Apple Host Computers (Intel Processors): VirtualBox or UTM
- Apple Host Computers (Apple Processors): UTM

I began this chapter with praise for newer Apple computers as OSINT machines. I currently rely on a 2022 MacBook Pro (M1) for all of my OSINT investigations. I only use virtual machines within UTM and possess no other virtualization software on my machine. I simply find UTM to be much more stable than the equivalents for other operating systems. I can quickly launch multiple Linux, Windows, and macOS VMs for my OSINT needs. A Linux VM boots in less than 10 seconds; a Windows VM in less than 19 seconds; and macOS VMs somehow boot in only 6 seconds. I can easily test software and customizations without much of a time commitment. Using UTM's "Run without saving changes" option allows me to avoid snapshots when I want a disposable machine. We are spoiled OSINT professionals.

However, Apple is not required for OSINT work! Whether you are virtualizing Ubuntu within VirtualBox, UTM, or paid options such as Parallels or VMWare, the rest of this book applies to you. It is agnostic of your hardware or software choices.

CHAPTER THREE
WEB BROWSERS

If you are a Windows user, your default web browser is either Internet Explorer or Microsoft Edge. Apple users are presented Safari by default. I believe OSINT professionals should avoid these at all costs. All are inferior in my opinion, and you will encounter difficulties with some of the websites and services mentioned later. Therefore, we need a better browser.

Firefox (mozilla.org)

The most vital application in this chapter is the Firefox web browser. Most of the search methods that you will learn throughout this book must be conducted within a web browser. The Firefox browser has enhanced security and a feature called "Add-ons" or "Extensions". These are small applications which work within the browser that perform a specific function. They will make searching and documentation much easier. I also use the Chrome web browser when necessary, and will explain some customizations later. However, many of the extensions that I need are only compatible with Firefox. The following chapter applies to any recent version of Firefox, including builds for Windows, Mac, and Linux. However, Ubuntu users should make some modifications, as explained in a moment.

Downloading and installing Firefox is no different than any other application. Detailed directions are readily available on their website. The browser will not look much different from the browser you were previously using. When installing and executing, choose not to import any settings from other browsers. This will keep your browser clean from unwanted data.

Ubuntu users may notice that Firefox is installed as a "Snap", which I do not like. This installation tends to break many scripts and can cause issues with the custom Firefox profile. I conduct the following within Terminal to switch Firefox within Ubuntu from a Snap installation to a standard APT package. Copy and paste the following commands from the linux.txt file within your online portal at **https://inteltechniques.com/osintbook10**. When prompted, enter the username of "**osint**" and your password of "**book1443**" (without quotes).

```
sudo snap remove --purge firefox
sudo add-apt-repository -y ppa:mozillateam/ppa
echo '
Package: *
Pin: release o=LP-PPA-mozillateam
Pin-Priority: 1001
' | sudo tee /etc/apt/preferences.d/mozilla-firefox
sudo apt install firefox -y
```

For the purposes of this chapter, I will assume you are configuring the Firefox application now included in your new Ubuntu VM. However, all of the methods explained here could be replicated within your host or any other computer. If you will be using your host computer for any web browsing, Firefox is highly recommended as the default browser. Regardless of where you will be conducting your online investigations, have a properly configured Firefox application.

Before identifying Firefox resources which will aid in our OSINT research, we must first secure our browser to the best of our ability. While the default Firefox installation is much more private and secure than most other browsers, we should still consider some modifications. I personally use Firefox for all of my OSINT investigations in my VMs, and as my default web browser on my personal laptop. I no longer possess multiple browsers for various tasks. I believe that Firefox is the most robust, secure, and appropriate option for almost any scenario. However, I recommend changing the following settings within Firefox.

- Click on the menu in the upper right and select "Settings", "Options", or "Preferences".
- In the "General" options, uncheck both "Recommend extensions as you browse" and "Recommend features as you browse". This prevents some internet usage information from being sent to Firefox.
- In the "Home" options, change "Homepage and new windows" and "New tabs" to "Blank page". This prevents Firefox from loading their services in new pages and tabs.
- Disable all Firefox "Home Content" options.
- In the "Privacy & Security" options, enable "Delete cookies and site data when Firefox is closed". This cleans things up when you exit the browser.
- Uncheck all options under "Logins and Passwords".
- Change the History setting to "Firefox will use custom settings for history".
- Uncheck the box titled "Remember browsing and download history".
- Uncheck the box titled "Remember search and form history".
- Check the box titled "Clear history when Firefox closes".
- Do NOT check the box titled "Always use private browsing mode", as this will break Firefox Containers.
- Uncheck "Browsing history" from the "Address Bar" menu.
- Disable "Suggestions from the web" and "Suggestions from sponsors".
- In the "Permissions" menu, click "Settings" next to Location, Camera, Microphone, and Notifications. Check the box titled "Block new requests…" for each of these options.
- Uncheck all options under "Firefox Data Collection and Use".
- Uncheck all options under "Deceptive Content and Dangerous Software Protection". This will prevent Firefox from sharing potential malicious site visits with third-party services. This leaves you more exposed to software attacks, but protects your internet history from being shared with Google.

This is where I again deviate from previous editions. I often walk a fine line between OSINT and privacy. In my book ***Extreme Privacy***, I took many actions in order to protect every facet of our online privacy and security. This included a complete hardening of the Firefox browser which I use as a daily driver on my personal machine. In previous editions of this book, I carried over many advanced Firefox settings into the recommended OSINT machine. Today, I no longer feel these actions are appropriate.

Over the past year, many of the Firefox "about:config" modifications which I had previously presented began causing minor issues within investigations. Some invasive sites which needed documented as part of an investigation could not be viewed due to overprotective settings. I have had to remind myself of the scope of this book. It is designed to assist with online investigation and collection of digital evidence.

Our OSINT machine is not designed for personal internet usage and simply does not need the same security through modifications which I believe are appropriate for a personal computer. Therefore, I have removed all advanced Firefox settings from this book. Doing so eliminates the likelihood that problems will occur, and enables the browser to be seen as another common configuration ready to view any website.

Firefox natively respects your privacy and security more than other browsers. These recommendations are for those that want to tweak additional settings that may provide a new layer of protection, even if minimal.

I want to stress that the enhanced Firefox privacy and security settings presented within ***Extreme Privacy*** are still very appropriate for personal machines. I just don't believe they are necessary here. For those who follow both sides of my work (OSINT and privacy), please continue to use the advanced strategies within your personal machines. With this book I want to strive harder to isolate these two worlds. While OSINT and privacy complement each other, they should not be lumped into one bucket when it comes to our digital strategies.

Next, I will discuss the biggest benefit of Firefox, which is the abundance of helpful browser extensions called Add-ons, occasionally called Extensions.

Firefox Add-ons (Extensions)

There are thousands of extensions available for Firefox. Some are helpful, some are worthless, and some are just fun. This chapter will discuss several of them. The Firefox Add-ons, sometimes called extensions, detailed here will include a website for each option. You can either visit the website and download the add-on, or search for it from within Firefox. The latter is usually the easiest way. While Firefox is open, click on the menu in the upper right and then "Add-ons". This will present a page with a search field in the upper right corner. Enter the name of the extension and install from there. The following are my recommendations, in order of importance.

- **Firefox Containers**: Isolate specific sites within tabs which do not see settings from other sites.
- **uBlock Origin**: Block undesired scripts from loading.
- **DownThemAll**: Download bulk media automatically.
- **Bulk Media Downloader**: Download bulk media automatically.
- **VideoDownloadHelper**: Download media from a page with a click of a button.
- **FireShot**: Generate screenshots of partial and entire web pages.
- **Nimbus**: Alternative screen capture for large web pages.
- **SingleFile**: New alternative screen capture option for HTML files.
- **Exif Viewer**: Identify metadata embedded inside a photograph.
- **User-Agent Switcher and Manager**: Emulate various browsers and devices.
- **Image Search Options**: Conduct automatic reverse image searches.
- **Search By Image**: An alternative tool for reverse image search.
- **Resurrect Pages**: Enable historical search on deleted websites.
- **Web Archives**: Search archived versions of the current website.
- **Copy Selected Links**: Quickly copy all hyperlinks from a website.
- **OneTab**: Collapse or expand tabs into a single resource.
- **Stream Detector**: Identify embedded video streams for archiving.

The following pages provide explicit instructions for installing and configuring each of these Add-ons. At the end, I will explain how you can export your settings and replicate your work across practically any Firefox installation. This will preserve your work and allow you to receive an identical experience if conducting investigations across multiple computers. I will also present an option to run a few commands to replicate the setup explained within this chapter. This will also benefit your virtual machines. Ideally, you would complete all browser configurations within your original VM before cloning, exporting, or use of snapshots.

Order 0138471223

Firefox Multi-Account Containers (addons.mozilla.org/addon/multi-account-containers)

The first Firefox Add-on which I use daily is the Multi-Account Containers option from Mozilla. Multi-Account Containers allows you to separate your various types of browsing without needing to clear your history, log in and out, or use multiple browsers. These container tabs are like normal tabs, except the sites you visit will have access to a separate slice of the browser's storage. This means your site preferences, logged-in sessions, and advertising tracking data will not carry over to the new container. Likewise, any browsing you do within the new container will not affect your logged in sessions, or tracking data of your other containers. Below is an example.

On my personal laptop, I have a container tab open which I use to log in to my email provider. I have my inbox open in this tab. I want to order a product from Amazon, but I do not want them to see any cookies stored by my email provider. I also want to conduct a Google search, but do not want Google to see any data present from my Amazon search. I simply open a unique container tab for each of these events. Each sees the session as unique, and no data is shared from one service to another.

OSINT investigators can use this technique in many ways. With a traditional browser, you can only be logged in to one instance of a social network. If you are logged in to a covert Facebook account, then open a new tab and navigate to Facebook, you will be presented with the same logged-in account used in the previous tab. With containers, we can isolate this activity. You can log in to one Facebook account in one container, another Facebook account in a second container, and any additional accounts in their own containers. This applies to any service, such as Twitter, Reddit, or others. This allows us to simultaneously access multiple accounts within the same service without logging out or opening a different browser. This is a substantial update from previous editions. Let's configure it for optimal use.

Once installed, you will see a new icon in the upper right in your Firefox browser which appears as three squares and a "+" character. Click on it and select the container you want to open. Default options include choices such as Personal and Shopping, but you can modify these any way you desire. You can create, delete, and edit containers from the main menu. When you click the "Manage Containers" option, you can change the color or icon associated with a container or change the container name. The following tutorial replicates my configuration for OSINT investigations.

- Open the Multi-Account Containers menu and click the "Manage Containers" option.
- Delete all containers by selecting each and clicking "Delete This Container".
- In the "Manage Containers" menu, click the + in the upper left.
- Enter the name of your new container, such as "Alias 01".
- Choose a desired color and icon.
- Repeat this process to create the number of containers desired.

On my machine, I have the following containers.

Personal
Alias 01
Alias 02
Alias 03
Alias 04
Leaks
Google
Harmful

You can now either open a new container as a blank page or open links in a new or different container. The following are a few of my usage examples.

Multiple Logins: While in Firefox, I want to open Facebook inside a unique container. I click on the containers menu and select Alias 01. This opens a new blank tab within this container. I navigate to Facebook and log in to an alias account. I then want to log in to a second Facebook account, so I click on the containers menu and select Alias 02. This opens a new tab in that container. I then navigate to Facebook and receive a login prompt. I log in to my second account and can switch back and forth between tabs. You should note that Facebook can see you have the same IP address for each login, but they cannot see your cookies from one session to the other. You could replicate this process for any other social network or service. You could also have numerous Gmail accounts open within one browser.

Safety: While I am viewing my target's Twitter profile, I see a link to an external website from his page. I am logged in to a Twitter account within this container and I do not know what this linked website will try to load. I do not want to jeopardize my investigation. I right-click on the link and choose "Open link in New Container", and then select the desired container tab, such as "Harmful". The new tab will open within this container created for questionable websites. The page I open cannot see any cookies stored within my container associated with my Twitter login.

Dedicated Container: I assign specific websites to a container so they will always open in that container. I use this for Google because I do not want my search history associated with other investigation activities. If I ever visit Google as part of an investigation, the site will open in a new container tab which I designated "Google". This is regardless of which tab I try to use. The following steps configure this option.

- Create a Containers tab titled "Google".
- Click on the Containers menu and open a new Google tab.
- Connect to google.com and click the Containers menu and select "Always open This Site in…".
- Select the desired container.
- Navigate to google.com from a standard tab.
- Select "Remember my decision…" and then "Open in…" the desired container.

When complete, you have created a rule within Firefox. Any time you connect to google.com, regardless of the container you are in, or if you have Google set as your default search from within the URL field, Firefox will open a new "Google" tab to complete your connection. This isolates your Google traffic from any other tab, and applies to any Google sites, such as Google Voice, Gmail, etc. If you regret making this type of rule, you can either delete the entire container or just the policy for that site. In this example, I can go to the Containers menu; click the "Manage Containers" option; then select the Google container; then click the delete option.

When I first installed this add-on, I went a bit too far with customized containers. I wanted all Facebook pages to load in their own container, which prevented the ability to log in to multiple accounts. I removed this option and established the rule mentioned previously which allowed me to have multiple logins, but lost the isolation from Facebook to other websites. I created containers for most of the sites I visited, which was overkill. There is no perfect solution. Evaluate your needs and create the most appropriate set of containers vital to your investigation. If you want to isolate a container to only open designated sites, click "Manage Containers"; select the desired container; and enable the "Limit to Designated Sites" option. This prevents accidental openings of undesired sites within a secure container. On my personal laptop, my containers are focused on privacy and isolate invasive services such as Amazon, Google, and online payment services. I also isolate financial websites and personal email tabs. I highly recommend applying these same strategies to your personal devices. I possess the following containers on my personal laptop.

- Communications: Personal email and calendar accounts
- Financial: Banking and credit card accounts
- Search: All Google queries
- Alias: Any social network accounts in another name

uBlock Origin (addons.mozilla.org/firefox/addon/ublock-origin)

I have previously recommended NoScript, Adblock Plus, Privacy Badger, and Disconnect as privacy Add-ons that would help stop unwanted ads, tracking, and analytics. These are no longer present on any of my systems. I now only use uBlock Origin, as it replaces all of these options. This section may seem a bit overwhelming, but experimenting with the advanced settings should help you understand the functionality. Let's start with the basics.

Install uBlock Origin from the Firefox Add-ons page or directly by navigating to the application's website at https://addons.mozilla.org/en-US/firefox/addon/ublock-origin/. You are now protected on a basic level. By default, most known invasive advertisements, tracking code, and malicious content is blocked. This step alone would provide much needed protection from the internet. However, we can take it a step further.

Click on the uBlock Origin icon in the menu and select the "Dashboard" icon to the right, which appears as a settings option. This will open a new tab with the program's configuration page. On the "Settings" tab, click the option of "I am an advanced user". This will present an expanded menu from the uBlock Origin icon from now forward. Click on the "Filters" tab and consider enabling additional data sets that may protect your computer. I find the default lists sufficient, however I enable "Block access to LAN" under "Privacy". You now have extended protection that will be applied to all visited websites without any interaction from you. When you encounter a web page with a lot of advertisements, such as a news media website, it should load much faster. It will block many of the pop-ups and auto-play media that can be quite annoying when conducting research. This protection will suffice for most users, but dedicated OSINT analysts may choose to take a more advanced approach.

After you have enabled the Advanced settings as explained above, clicking on the uBlock Origin icon should now present an expanded menu which will change as you visit different sites. In order to explain the function of this menu, I will conduct a demonstration using the website cnn.com. Figure 3.01 displays the default view of uBlock Origin with the site loaded. While this book is printed in black and white, your view will be in color, and likely all options will appear grey. Scrolling down this list of scripts that have either been loaded or blocked, you can see several questionable scripts such as Twitter, Amazon, and Turner. These scripts allow tracking across multiple websites and are the technology responsible for monitoring your interests, web history, and shopping habits.

This menu is split into three columns. The first simply identifies the type of code or domain name of the script. The second column is global settings. Anything changed here will apply to all website visits. The third column contains settings for the current website. A single plus sign (+) indicates that less than ten scripts were allowed from that specific option. Two plus signs indicate that between ten and one hundred scripts were allowed. The single minus sign (-) indicates that between one and nine scripts were blocked from that domain, while the dual minus signs tell us that ten to one hundred scripts were blocked. In Figure 3.01, we know that over ten scripts were allowed to run from cnn.com, and at least one script was blocked from sending data to Twitter. This is all default behavior and provides a balance of functionality and security. uBlock Origin decides which content should be allowed and which should be blocked.

Using this same page, let's modify the options. In Figure 3.02 (left), I have clicked on the far-right portion of the first cell in the third column. This turned the entire third column red in color. This action activated an option to refresh the page (arrows) and an option to save the change (padlock). Clicking the padlock and then refreshing the page presented me with the example in Figure 3.02 (right). Since I blocked every script, the page would not fully execute. It could not load images, design scripts, or any JavaScript. This is not useful at all, so I disabled my actions by clicking on the left (grey) section of the top cell in the third column, which turned the entire column back to grey in color. Saving these changes and refreshing the page brought me back to the example in Figure 3.01.

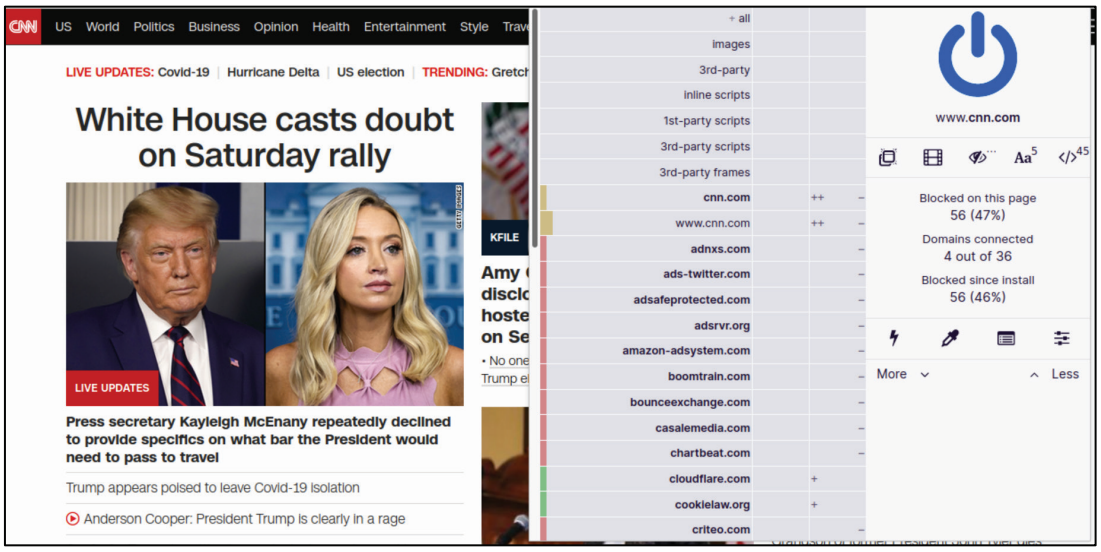


Figure 3.01: An advanced view of uBlock Origin.

We can also take this to the opposite extreme. In Figure 3.03 (left), I clicked on the "power button" in the upper-right. This turned the entire left edge green in color, and allowed all scripts to load on cnn.com. This includes the dozens of intrusive scripts that could load advertisements on the page. You can also see that small plus signs confirm that scripts were allowed to run while the minus signs in Figure 3.03 (right) state the opposite. For most users, this allowance would seem irresponsible. However, there is a specific reason that we want the ability to allow all scripts. If you are collecting evidence, especially in criminal cases, you may want to archive a page exactly as it was meant to be seen. When we block scripts, we are technically modifying the page (evidence). By intentionally allowing all scripts before the collection of the screen capture, we know that we are viewing the page in an unmodified format. This may be overkill for many investigators, but you should know your options.

Next, we will modify the second (middle) column, which will apply settings globally. By default, all options are grey in color, which is desired by most users. This indicates that the default block list is applicable, and only invasive scripts will be blocked everywhere. For demonstration, I clicked on the right (red) portion of the top cell in the second column. This turned the entire column red, and indicates that all scripts across all websites will be blocked. After I saved my changes, every website will only load the most basic text content. This will prohibit much of our research.

Loading a page such as a Twitter profile resulted in no usable content. By clicking on the uBlock Origin icon and clicking the left (grey) sections of specific cells within the third column, I enabled those scripts without allowing everything on the page. While you cannot see the colors in Figure 3.03 (right), you can see the difference in shading. In this example, the entire second column is red. This indicates that all scripts are blocked globally. The third column is mostly red, but the options for twitter.com and twimg.com are grey. Those scripts will be allowed, if approved by uBlock Origin's rules, only for that domain. If I load a blog that has scripts from Twitter, they would still be ignored.

These are extreme examples. Let's bring this back to some sanity. The following is how I recommend using uBlock Origin. Install, enable advanced options, and proceed with your work. When you arrive at a website that is blocking something you want to see, open the menu and click on the left (grey) section of the top cell in the third column. That will allow everything to load on that page, and that page only. When you are about to navigate to a questionable site that may try to install malicious code on your machine, click on the right (red) section of the top cell in the second column. That will block all scripts on all pages. Conduct your research and reverse the change when you are finished. Remember to click the save button (padlock) after each change and refresh the page.

Hopefully, you are practicing these settings and learning how this program functions. It is an amazing option that has protected me many times. If you are doing things right, you have likely completely messed-up your settings and are now blocking things you want while allowing things you do not. Don't worry, we can reverse all of our mistakes by first changing the global (second column) settings back to grey (left section of top cell). Next, return to the dashboard settings of the add-on, and click on the "My Rules" tab. In the second column (Temporary Rules), select all of the text and press the delete key on your keyboard. Click the "Save" button in this same column and then the "Commit" button to apply these settings everywhere. This resets our extension and brings us back to default usage regardless of your modifications. This is important in the event you go too far with settings in the future. Removing and reinstalling the extension does not always wipe this data out of your system.

The primary benefit of uBlock Origin over other options is the simple ability to block malicious scripts without customization, while having an option to allow or block any or all scripts at our disposal. This is a rarity in these types of Add-ons. Another benefit is the ability to bypass website restrictions, such as a news site blocking articles unless the visitor has a subscription service. Consider the following example with the Los Angeles Times. Visiting the page allows you to view three articles for free, but you must have a paid subscription in order to continue using the site. If I click on the uBlock Origin menu while on this page, select the right (red) option on the right (third) column under the setting for "3rd party scripts", then the padlock icon, and reload the page, I see a different result. I am now allowed to see the article. This is because this website relies on a third-party script to identify whether a visitor is logged in to the service. This modification presents unlimited views of articles without registration on this and thousands of other websites.

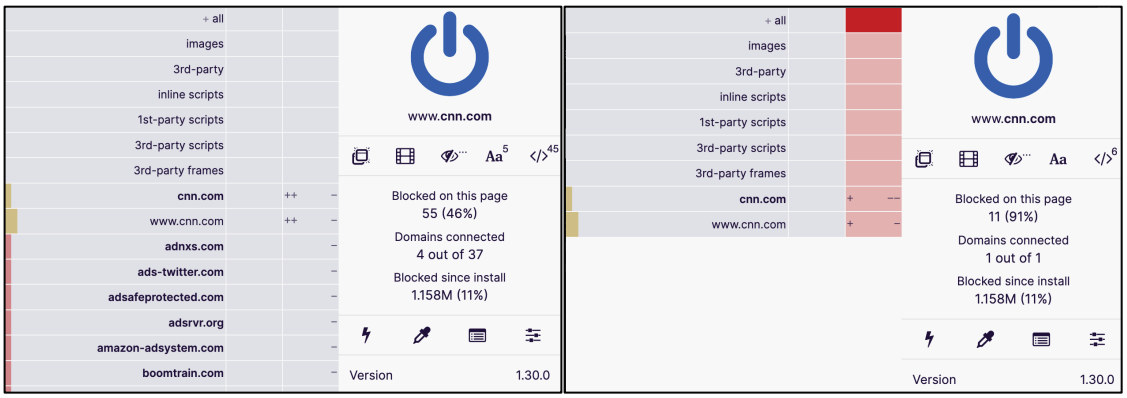


Figure 3.02: Disabled scripts within uBlock Origin.

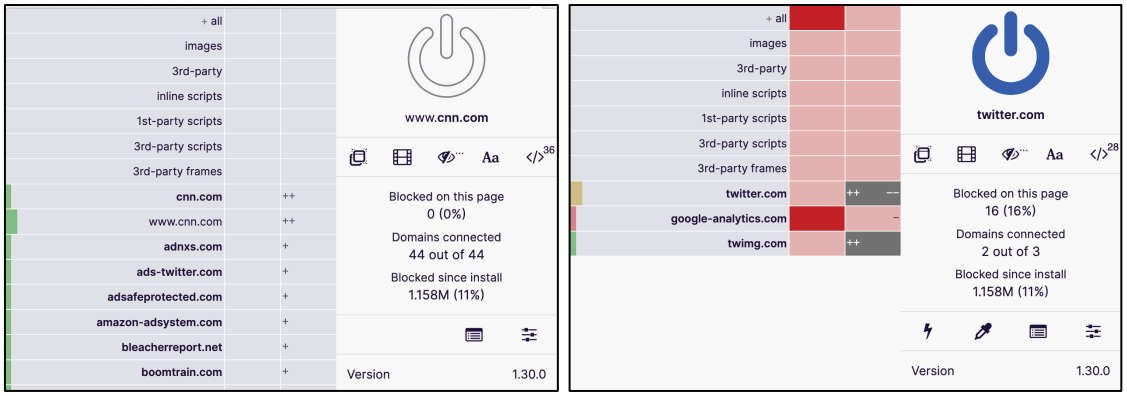


Figure 3.03: Fully and partially enabled scripts with uBlock Origin.

The final example of uBlock Origin within this chapter, which I rely on with my daily browsing, is the Inline Scripts blocker. For this demonstration, I will navigate to cnn.com. Clicking on almost any article presents a new page, including new annoyances. An undesired video begins playing while scrolling down the page, multiple images from unrelated articles populate over some of the article text, a pop-up advertisement interrupts my view, and over 56 scripts attempt to monitor your activity. uBlock Origin blocks the malicious scripts, but not all of the annoyances. We could block each script individually, but that is time consuming. Instead, consider a simple modification of the inline scripts setting.

Click on the uBlock Origin menu while on the desired page, select the right (red) option on the right (third) column under the setting for "inline scripts", then the padlock icon, and reload the page. The site should load much faster and block all of the inline scripts being pushed to you by the provider. You should notice that all pages on cnn.com load immediately, and without all of the undesired media interfering with your research. Clicking the grey area in this same box reverses the action. I apply this feature to practically every news website I visit. It blocks vital information if your desire is to obtain a screen capture as evidence, but provides a much more pleasing environment if you simply want to read the articles.

DownThemAll (addons.mozilla.org/firefox/addon/downthemall)

DownThemAll simplifies the process of extracting bulk data from a web page. It attempts to identify linked video, audio, images, documents, or any other type of media within a site. It then allows you to easily download everything at once. Consider the following example.

Archive.org possesses a copy of a website which once offered several gigabytes of marketing data containing millions of records on Americans, including full names, addresses, telephone numbers, and interests. The archive can be found at the following URL.

https://web.archive.org/web/20151110195654/http://www.updates4news.com:80/kyledata/

This URL presents hundreds of large CSV and TXT files. Later in this book, I discuss how to properly parse through this content and create your own searchable file with this example. For now, I simply need to download each file. While on the page, click on the DownThemAll toolbar menu and select "DownThemAll". In the new window, you should see all of the data links present on this site. Clicking the "All Files" box near the bottom selects each of the files. Clicking "Download" in the lower right begins the process of downloading all of the data from the page (which may take hours), and places each file in the default download location for your operating system. Please do not download this data set yet, as it will fill your disk space in your VM. We will discuss external storage methods later. This add-on is a requirement for any browser I use. I find it to work better than the next option, but consider all alternatives for your own needs.

Bulk Media Downloader (addons.mozilla.org/firefox/addon/bulk-media-downloader/)

Similar to DownThemAll, this add-on can make downloading a large number of media files easy. It should serve as a backup in the event you find a page which DownThemAll will not function properly. If you locate a page of several audio or video files, it can be time consuming to save them all manually. Additionally, you run the risk of accidentally skipping a file. Bulk Media Downloader provides a solution. As an example, I navigated to Twitter and searched the word Video. This presented hundreds of embedded videos within a single page. I launched Bulk Media Downloader, which displayed a pop-up option over my browser. In this pop-up, I can select specific file types such as Video or Audio. I chose only the Video option and reloaded the Twitter page in the background. The Bulk Media Downloader tool began populating video links as I scrolled down the Twitter page. Clicking the Download button retrieved all of the videos in MP4 format. This utility works well on sites that have a large number of embedded audio or video files, as well as those that contain numerous documents. You can easily select or deselect entries individually, or select categories at the bottom that fit your needs.

Video DownloadHelper (addons.mozilla.org/en-US/firefox/addon/video-downloadhelper/)

This extension will assist with downloading media that is located during your search. It works well with videos such as those found on Vimeo and other hosts. It does **not** work well with YouTube, and we will use a better option in the next chapter. When this extension is enabled, an icon will appear within your browser that looks like three grey circles. Any time you open a website that includes media content, such as a video, these circles will turn to full color. This is an indication that the media on the page can be extracted. While this add-on will work immediately after installation, I have found specific configuration changes to be helpful to OSINT investigators.

- Click on the icon placed in your menu bar and select the Settings icon.
- Click the "Behavior" tab and change the "Max concurrent downloads" to 20.
- Change the "Max Variants" to 99.
- Select the "Hide ADP Variants" option and click "Save".

When downloading videos, especially from YouTube, the ADP format requires secondary conversion software to be installed. I do not like this option as it introduces unnecessary software to my machine. Furthermore, I never want to convert video evidence. I simply want to extract the options available directly from the source. Therefore, eliminating the ADP options from our view as explained above reduces the chance of downloading undesired content. In Figure 3.04 (left), the ADP options are present and would not be ideal download choices. In the example on the right, I have eliminated these choices and I am presented with more appropriate options.

You can now extract embedded media files from websites by clicking the icon and selecting the appropriate file. If your desired media is going to be used in court, I recommend downloading all sizes available. If you only want a personal archive, the largest size should be downloaded. You will now have a pure digital extraction of the target video. This is better than a screen capture or recording of the video because there is less loss of data from analog conversion. If downloading a large number of videos, consider the script which will be explained soon.

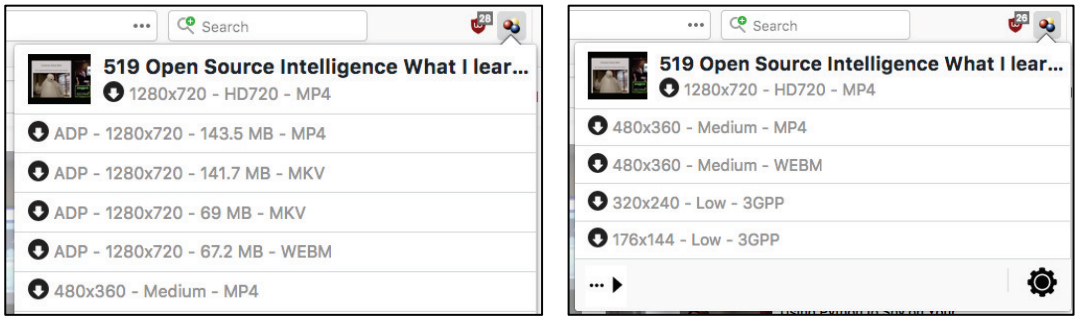


Figure 3.04: Menu options from Video DownloadHelper.

Fireshot (addons.mozilla.org/en-us/firefox/addon/fireshot/)

Documenting and archiving your progress with an OSINT investigation is as important as the intelligence discovered. The general rule is that if you do not have proof of your findings, then they never existed. Full Web Page Screenshots, also known as FireShot, provides you with an easy solution to capturing all of your results. When enabled, this extension is a button in the upper right portion of your browser. It appears as a blue square containing the letter "S". Clicking the icon presents a menu with options. The best option is to select "Capture entire page" and then "Save to PDF". This will create a PDF document of the entire page exactly as it appears in your browser and then save it to anywhere you choose. The file can later be archived to a removable storage device. The title of the document will match the title of the web page and it will include the URL of the page.

This method is preferred over a standard screen capture for several reasons. A typical screen capture only captures the visible area and not the entire page. You must then open a program into which you "paste" the data and then save the file. This extension automates this and saves it in a format that is difficult to edit. This can be beneficial during testimony.

By accessing the "Options" area of the menu, you can assign customized naming features. Click "Show filename template settings" in the options page and change the default value to the following.

%n-%u-%t-%y-%m-%d-%H-%M-%S

This setting will change the default name of each page capture. Each file will be named a numerical value, followed by the website URL, followed by title, and followed by the date and time of capture. Changing the %n value to 0 and the Pad option to 3 will ensure that your captures always start with a numerical value of 0 and ascend chronologically. This can help determine the order of evidence you retrieved. Be sure to "Apply" and then "Save" after you have made your desired changes. Notice that you can quickly see the order captured (first three digits), target website, description, and date & time.

Nimbus (addons.mozilla.org/firefox/addon/nimbus-screenshot)

While FireShot is my preferred screen capture utility within Firefox, there are some instances where it does not perform well. If you have a target's Facebook page that has a lot of activity present, this may create a screen capture too large for FireShot. The rendering process will likely expend all of the computer's video memory and fail to create the file. When this happens, I use Nimbus as my first backup.

Nimbus allows you to specify whether you want to capture only the visible portion of the page, the entire page, or a custom selection from the page. The drop-down menu presents these choices and the result is saved as a PNG file. This is not optimal for online investigations, but is better than no capture at all. Another feature of Nimbus is the ability to manipulate captures. I believe that this is bad practice as we usually want to provide the most authentic and accurate evidence as possible. I do not want to manipulate any potential evidence. Therefore, I recommend the following configurations.

- Click on the Nimbus icon and choose the "gear" icon in the lower-right.
- In the "File name pattern" field, insert {url}-{title}-{date}-{time}. This will name every capture with the URL and title of the target website along with date and time of capture.
- Check "Enable Quick Screenshot" and select the "Entire Page" option in the first row and "Download" option in the second row.

After these changes, clicking the Nimbus icon in the menu bar will no longer present a menu with options. Instead, it will automatically select the entire page, apply the proper file naming, and download the capture as a maximum quality PNG file to your Desktop. While a PDF file created with FireShot is the preferred file format, a PNG file has other advantages. The PNG file is more universal and does not require PDF viewing software such as Acrobat Reader. However, PNG files are easy to edit, and establishing the integrity of the file may be difficult. I believe that Nimbus should be used as a supplement to FireShot.

One common failure of both FireShot and Nimbus is the capture of extremely large Social Network pages. While this is rare on computers that have ample resources such as processing power and RAM, it can be quite common on older machines with low specifications. Surprisingly, I have found FireShot to work better on large Twitter profiles and Nimbus to be best for large Facebook pages. I have no logic to offer for this discovery. Again, having both at our disposal will make us better prepared for online evidence collection. When both fail, consider Firefox's own solution, displayed next.

Firefox Screenshot

You may not like either FireShot or Nimbus. In general, you get what you pay for with these (they are free). When I have an extremely long Facebook or Twitter page, I find both of those options mediocre at best. Lately, I find myself using the embedded Firefox Screenshot utility more than anything else. Consider the following example.

I loaded my own Twitter profile and scrolled back to posts from a year prior. This generated quite a long page and my computer fans increased speed due to the heat generated from my processor. I attempted a screen capture with both FireShot and Nimbus, and each failed. I then executed the following with Firefox.

- Right-click within the page and select "Take Screenshot".
- Choose "Save full page".
- Click "Download".

The process ran for about three minutes and saved an image .png file to my default downloads directory. It was several megabytes in size. The default filename includes the word Screenshot, date, and title from the webpage.

SingleFile (addons.mozilla.org/en-US/firefox/addon/single-file/)

The previous options have always been the standard way of saving content. However, we should always embrace new services. In late 2022, I was struggling to archive a large page, which led me to seeking alternative capture Add-ons. SingleFile saves any page as a HTML file, which can bypass many resource restrictions while generating a PDF. Before we test it, let's modify several settings. After right-clicking the SingleFile icon in the menu bar and choosing "Manage Extension", I conducted the following.

- Click the "Preferences" tab.
- Expand "User Interface" and enable "Open saved pages in a new tab".
- Expand "HTML Content" and disable "Remove hidden elements".
- Enable "Save original URLs of embedded resources".

You can now load any page and click the SingleFile icon to begin the process. The add-on will collect all of the content from the page and embed it into a single HTML file within the Downloads folder of your VM or host. You can open this file within your web browser to see the archive. Our changes simply apply settings which try to save an archive close to the original. By opening the result in a new tab after completion, we will know that everything worked.

We can also save multiple tabs all at once within the SingleFile menu. Executing this feature archived each page as a separate file and also opened each archive for immediate viewing. You should notice the file names now include the title of the page and full timestamp. The sizes assure us that all of the data is embedded into the file. Since this utility is embedding Base64 images directly into the HTML file, we only have one file to archive or export. Lately, this tool has been much more reliable than PDF export options, and the source of the HTML captures additional pieces of data for our investigation. The first few lines of every save now include details such as the following.

```
Page saved with SingleFile
url: https://inteltechniques.com/book1.html
saved date: Mon Oct 31 2022 15:27:21 GMT+0000 (Coordinated Universal Time)
```

Exif Viewer (addons.mozilla.org/en-us/firefox/addon/exif-viewer)

This extension provides right-click access to the Exif data embedded into images. Later chapters explain what Exif data is and how it can be useful. With this extension enabled, you can right-click on any full size image located on a web page. The menu option is "View Image Exif Data" and a new window will open when selected. This window will identify any available metadata about the image. Figure 3.05 (left) displays the right-click menu with a new option to View Image Exif Data. Figure 3.05 (right) displays partial results that identify the make and model of the camera used to capture an image.

Overall, most photos on social networks do not contain any metadata. They have been "scrubbed" in order to protect the privacy of users. However, many blogs and personal websites display images that still contain metadata. While Chapter Twenty will explain online websites that display this metadata, a browser add-on is much more efficient. In my experience, this extension will increase the number of times that you will search for this hidden content.

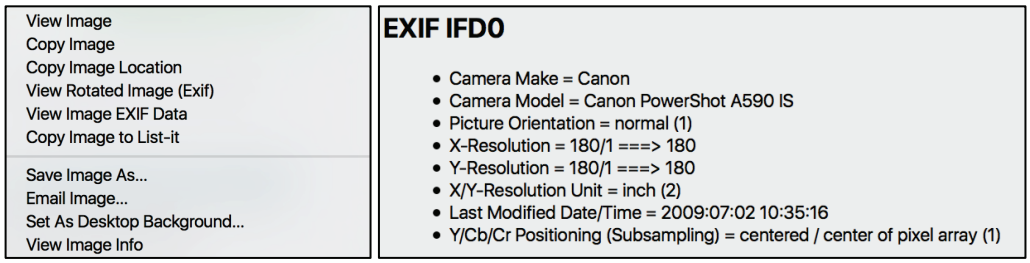


Figure 3.05: The right-click menu and result from an Exif Viewer search.

User-Agent Switcher and Manager (addons.mozilla.org/firefox/addon/user-agent-string-switcher/)

Occasionally, you may visit a website that does not want to cooperate with Firefox. Browsers notify websites of their identity and websites can alter or refuse content to certain products. One example is that some older websites require Microsoft's Internet Explorer to view the content. Even though Firefox is capable of displaying the information, a website can refuse the data to the browser. Another example is mobile websites that display different content if viewing from an iPhone instead of a computer. This can now all be controlled with User-Agent Switcher and Manager. Note that this is a different application than the recommendation within the previous edition of this book (User-Agent Switcher). User-Agent Switcher and Manager offers many new features, one of which we will discuss in the Instagram chapter.

When installed, you have a new option in your browser. The menu allows you to choose a mobile operating system, such as iOS or Android, or a desktop browser such as Internet Explorer or Chrome. It will also allow you to specify your operating system such as Mac or Windows. Whatever you choose, this data will be sent to any site that you visit. If you visit a website of a tech-savvy target, he or she may know that you were looking around. You may also be revealing that you are using a specific browser, such as Firefox, and a Windows computer (common in government). You could now change your agent to that of a mobile device or Google Chromebook which may not look as suspicious.

To do so, you would click on the menu bar icon; select the desired browser to emulate (such as Edge), select the desired operating system (such as Windows); choose exact offering (such as Edge 92.0.1 Windows 10); and click "Apply (container on window)". Click the "Test UA" button to visit a page which will confirm your new active user-agent. To return to the default Firefox option in your native operating system, click on the "Restart" button and refresh the page. Figure 3.06 displays an example where a mobile version of Yahoo was delivered to a desktop computer. Figure 3.07 displays the same page without any user-agent spoofing.

I have used this on several occasions to bypass poor security protocols. During one investigation, I had encountered a web forum of a hacking group that always appeared blank upon visit. Google had indexed it, but I could not see any content. By changing my default agent to Firefox on a Linux machine, I was allowed to see the content. The group had enabled a script that would only allow the page to be viewed on Linux computers.

In another situation, while still employed by the government, various mandated online training needed to be completed in order to maintain specific certifications. This government-hosted training was poorly designed and required users to access via Internet Explorer. Since I used an Apple computer, I could not connect until I changed my agent to Internet Explorer within my Firefox browser.

Please note that user-agent spoofing will not fool every website. If the target site includes JavaScript which scans for additional identifiers, such as touch points and video cards, the real details may be presented. I host a demonstration page at **https://inteltechniques.com/logger** which you can use to see how some sites may bypass your trickery. While this level of scrutiny is rare, it is a possibility. Always know what websites may be able to see about your computer and connection before any sensitive investigation. Refresh my test page as you make changes to this extension.

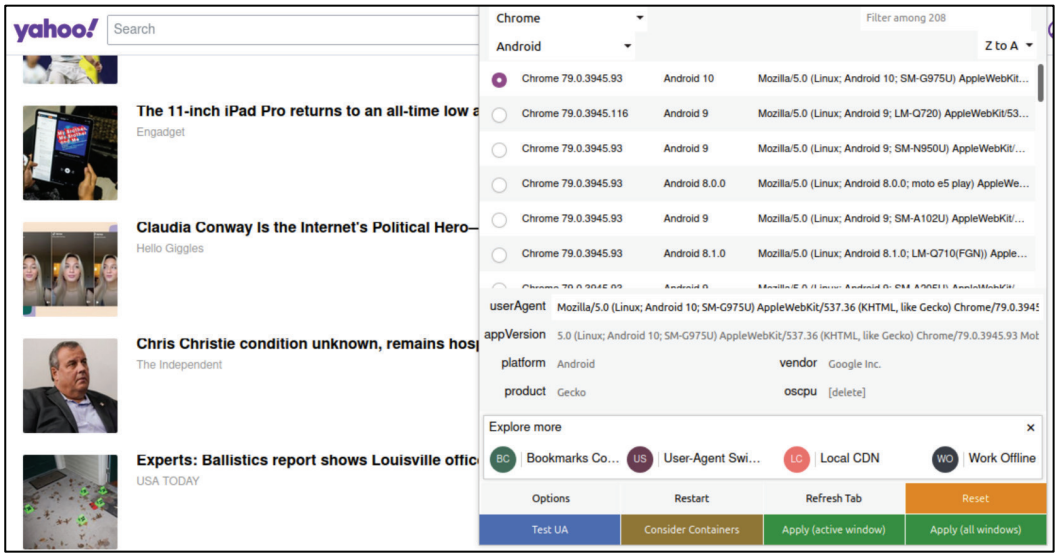


Figure 3.06: User-Agent Switcher and Manager disguising a desktop system as a mobile device.

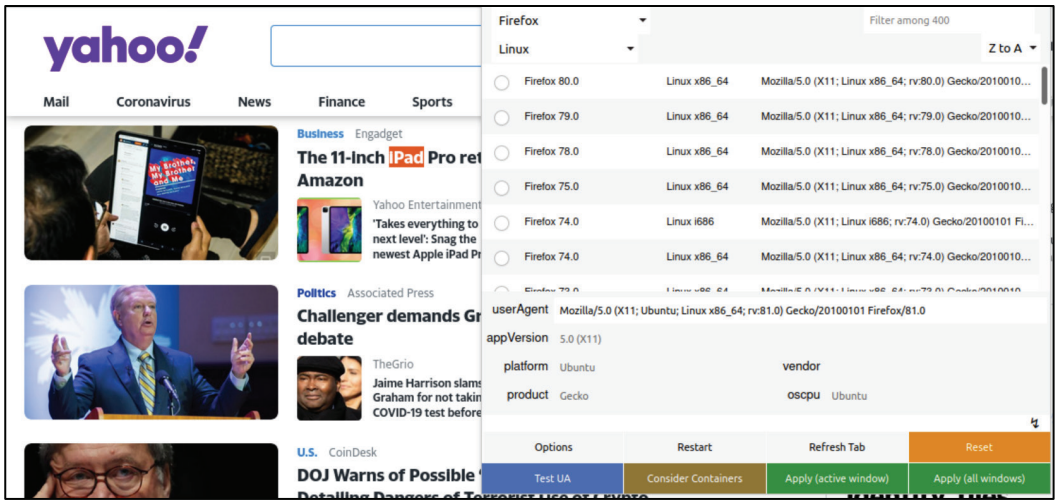


Figure 3.07: A page without user-agent spoofing.

Image Search Options (addons.mozilla.org/firefox/addon/image-search-options/)

A later chapter explains reverse image search engines and how they can identify more target photographs. Popular options include Google Images and TinEye. This extension automates the reverse search when an image is right-clicked. When installed, "Image Search Options" is present when you right-click on an image. Highlighting this option presents several reverse image search services including Google, Bing, TinEye, Yandex, Baidu, and others. You will later learn how my online search tool will execute an image search across all of those services at once. However, this tool can be beneficial due to the convenience and obscure services such as Karma Decay, which looks for copies of images on Reddit. This add-on removes any excuse to not always check reverse images on target websites. With this add-on enabled, you will be ready to enhance your searches during that investigation.

Search By Image (addons.mozilla.org/en-US/firefox/addon/search_by_image)

The previous option has been present in my OSINT VMs for a long time. Recently I added Search By Image as an alternative add-on. While it seems to be updated more often and has additional options, Image Search Options also has unique services missing from this one. I plan to keep them both available in case one should ever become blocked by a specific service. However, Search By Image has one huge advantage. It allows you to capture an image from a page without knowing the source URL or being able to right-click an image file. Consider the following scenario.

Assume you are on a website which has either restricted right-click access or has embedded images in a way which makes them inaccessible. This is very common within newer social networks. Search By Image allows you to crop any visible part of a page and then conduct a reverse image search analysis across multiple services. The following steps walk through the process.

- Left-click the Search By Image icon in the upper-right of the browser.
- Change the "Select URL" drop-down menu to "Capture".
- Select the desired service, such as Google.
- Use your cursor to draw a rectangle around the target image.
- Click "Search" in the lower popup menu.

A new tab should open with your reverse image search results. This can be a vital utility when you encounter sites which strive to prevent unauthorized copying of content. I have also used this on protected PDF files which contain images but do not allow right-click access or ability to extract images. You could even open a locally-stored PDF within Firefox in order to take advantage of this tool.

Resurrect Pages (addons.mozilla.org/en-US/firefox/addon/resurrect-pages/)

This extension provides a link to archived versions of websites whenever a page has been modified, is unavailable, or has been deleted. Right-clicking on any site will offer a new menu option of "Resurrect this page". That option will present the following archive services.

Google: A standard cache of the target address from Google
Google (Text Only): The text-only view of a standard Google cache
The Internet Archive: A link to the target page within The Internet Archive
Archive.is: Any captures of the target address domain on Archive.is
WebCite: Any captures of the target address domain on WebCite
Memento Timetravel: A standard cache of the target address from Memento

This add-on will not give you any content that you could not locate manually from these sources. Instead, it serves as an easy way to quickly identify interesting content. You will learn more about online archives later in this book.

Web Archives (addons.mozilla.org/en-US/firefox/addon/view-page-archive)

This option is redundant to the previous technique, but is much more robust (and more often updated). Let's change a few settings before we tackle usage. I clicked the Web Archives icon in the menu bar; expanded the three dots to see the menu; and selected "Options". From there I conducted the following.

- Enable "Google Cache (Text Only)".
- Enable "Memento Time Travel".
- Enable "Gigablast Cache".
- Enable "Megalodon".

We can now click the Web Archives icon and choose from multiple services. Each will query the current tab for archives from previous dates. We also have the "All" option which opens a new tab for each activated service. This can be fairly resource intensive and some VMs may struggle.

Copy Selected Links (addons.mozilla.org/en-US/firefox/addon/copy-selected-links/)

This simple add-on will identify any hyperlinks within the selected text of an individual web page. It will store the links within your operating system's clipboard, which allows you to paste them into any application of your choice. While only a small utility, it can quickly turn a large project into an easily completed task.

Using the utility is fairly straightforward. While on any website, select any or all text, right-click anywhere in the page, and select the "Copy selected links" option in the menu. The links will be stored in your clipboard and you can paste them into Notepad, Excel, or any other productivity application. There are unlimited uses for Copy Selected Links, and below are a few of my favorite.

Facebook: When I am on my target's list of Facebook friends, I will select all text and use Copy Selected Links to quickly record each hyperlink to an individual's profile. I will then paste these into Excel for later analysis. Comparison with previous captures identifies those that were "unfriended".

Twitter: When I am viewing a Twitter profile, I will use this utility to capture all links to external websites and photos.

YouTube: When viewing a person's YouTube videos page, Copy Selected Links allows me to paste the entire link collection of every linked video into a report.

eBay: While viewing results from a search for a specific fraudulent product, I can quickly copy the active hyperlinks to each auction and paste them directly into a report in seconds.

Human Trafficking: While viewing ad results for suspected human trafficking victims, I can copy all active hyperlinks and paste directly into a report, email, or memo for other investigators.

Documents: When I encounter a public FTP server or open web directory, this tool allows me to copy the native links to all files encountered. This is helpful for documentation after downloading all of the data.

PDFs: Since I can open PDF files within Firefox, I can also use this tool to extract all hyperlinks from within the file.

Performing screenshots during my investigation in these examples would never identify the direct links to the visible content. Only hovering over the links would temporarily identify the source. A combination of screen captures and link collection with this add-on provides a much more comprehensive report. I always press "ctrl-a" to select all of the content; right-click within the page; and choose "Copy selected links". I can then paste all hyperlinks from that page within any text editor.

OneTab (addons.mozilla.org/en-US/firefox/addon/onetab/)

This extension provides simple and effective management of browser tabs by allowing you to collapse any open pages into a list of bookmarks. Once installed, it is activated by clicking on the blue funnel icon on your toolbar. Doing so will close any open tabs and open the OneTab management page. The collapsed tabs are displayed on this page as a group of bookmarks. Each individual bookmark is made up of a page title, with the respective URL embedded as a link. Any previously saved tab groups are stored further down the page in reverse chronological order.

Interacting with the management page is straight forward. At the top of each tab set is an editable title which defaults to the number of tabs in that group. Left-click on the title to change it to something logical for that set of links, such as "Username Subject X". To the right of the title is a date and timestamp indicating when the list was saved. You can drag and drop individual bookmarks to change the order in each group or to move links from one group to another. Clicking on "Restore all" will re-open each of the bookmarks into its own tab.

The bookmarks are links, not saved pages, so you will be reloading any non-cached page content from the remote host. Selecting "Delete all" will destroy the bookmarks in that group. Selecting "More…" gives options to rename, lock, or star the tab group. "Star this tab group" pins that group to the top of your management page, independent of the date and time created.

Using "Share as web page" is not recommended as it creates an html page of your tab group on the OneTab servers. The preferred method for transferring OneTab bookmarks is to use the export feature, which is found in the upper right menu. This will allow you to copy links as plain text and paste them into a secure platform.

OneTab defaults to deleting the bookmarks from a group if you select "Restore all". To change this behavior, select "Options" and select "Keep them in your OneTab list". Right-clicking on a web page will bring up the OneTab context menu which allows for more granular tab selection. OneTab collects no user data unless you intentionally click on the "Share as web page" feature. Barring that feature, all data is stored locally.

Stream Detector (addons.mozilla.org/en-US/firefox/addon/hls-stream-detector)

This extension has become vital during my daily OSINT routines due to the prevalence of streaming video within my daily investigations. Whether pre-recorded or live, many websites now deliver video via embedded streams rather than traditional files. While we could once easily right-click an online video and download the file as an MP4, that is rare today. We must now be prepared to encounter video stream protocols such as playlists (M3U8) and transport stream (TS). Let's start with an example.

While writing this chapter, sources on Twitter reported the death of Sean Connery. The page displaying various news articles (https://twitter.com/i/events/1322516496564723713) included a post with video which linked to cnn.com. Right-clicking this video provided an option to play the video, but nothing to download it. Monitoring the network activity of the page through "Inspect Element" presented numerous small files, none of which were the entire video. However, clicking on the Stream Detector extension presented the options visible in Figure 3.21. The bottom "master" option is a link to the M3U8 stream which is responsible for the video playback. Clicking this link copies it to the clipboard. The full link is as follows.

https://cnnios-f.akamaihd.net/i/cnn/big/entertainment/2020/10/31/sean-connery-dead-james-bond-vpx-pkg-elam.cnn_3462706_ios_,650,1240,3000,5500,.mp4.csmil/master.m3u8

Downloading and opening this link within VLC media player, which is explained in the next chapter, plays the entire video within the software. The Linux-based Python tools which we create in the next chapter help us download the entire video for offline use. While this extension may seem complicated now, it will simplify archiving of online video streams in the next chapter. For now, the power of this utility is the ability to document our evidence.

I can now disclose the exact digital source of the video stream. This can be extremely valuable when a website is later removed or altered, but the media remains untouched on the media hosting server. We will explore online video streams much more later.

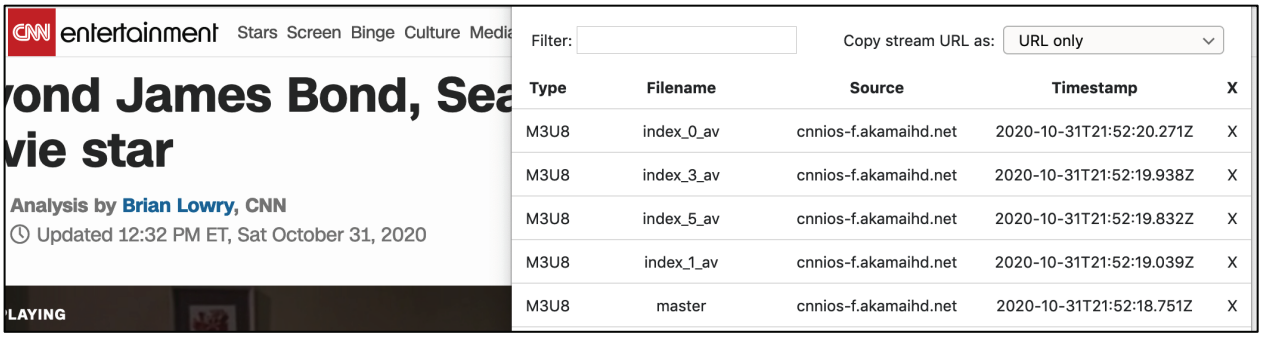


Figure 3.08: The Stream Detector extension displaying embedded video streams.

If this extension provides value to your investigations, but you find some sites which do not function, consider adding another Firefox extension titled **The m3u8 Stream Detector** (addons.mozilla.org/firefox/addon/the-m3u8-stream-detector). It may pick up missing streams. This add-on will be vital during the Broadcast Stream chapter much later in the book. We will use it to immediately identify files which stream live news coverage from all over the world, and that file will be used to populate our custom search tool. This is a great example of how one OSINT resource works in tandem with others.

Exporting and Importing a Pre-Configured Profile

At this point, it may seem overwhelming when thinking about the abundance of Add-ons and their proper configurations. I currently use several Windows, Apple, and Linux virtual machines and must keep my Firefox browser updated on all of them. I no longer manually update each browser. Instead, I maintain a single Firefox browser that includes all customizations that I desire. I then import these settings into any other browsers in order to replicate the experience across every computer I use. In a moment, I share my profile for easy import.

The following instructions allow you to export your final settings and import the same customizations into other investigative computers. The details will vary slightly based on your operating system and version. Only execute this tutorial on a new install of Firefox that has no saved settings, or within a version of Firefox that you want to be overwritten. Do not overwrite your current version if you have bookmarks, extensions, or other data that you want to keep. You should backup any settings if proceeding on an older install. As a final warning, the following steps will overwrite any custom options applied to your target Firefox installations.

- Open your CONFIGURED version of Firefox and click the menu button (three horizontal lines), then click "Help" and select "Troubleshooting Information". The Troubleshooting tab will open.
- Under the "Application Basics" section, click on Open (or Show) Folder (or Directory). A window with your profile files will open. Close Firefox, but leave this window open.
- Copy these files into a new folder on a removable drive.
- Open your NEW version of Firefox and click the menu button (three horizontal lines), click "Help", and then select "Troubleshooting Information". The Troubleshooting Information tab will open.
- Under the "Application Basics" section, click on Open (or Show) Folder (or Directory). A window with your profile files will open. Close Firefox, but leave this window open.
- Paste the content of the new folder on your removable drive into this folder. Overwrite any files when prompted. Restart Firefox.

The result should be a copy of Firefox which contains every add-on you configured from this chapter. This profile could be copied to an unlimited number of computers, as long as the versions of Firefox were identical.

I once visited a large fusion center which had created a custom Firefox profile for use on all computers. This made everyone's version of Firefox practically identical and ready for online investigations. When an employee was assigned to a different workstation, the look and feel of the browser was identical. This also included blocking of specific scripts globally throughout the organization with a custom version of uBlock Origin.

Custom Firefox Profile

While I prefer readers create their own Firefox configuration, I respect that a lot of effort is required to replicate these steps within multiple machines. Therefore, I have created a custom file which includes every configuration explained within this chapter. It can be used within any instance of Firefox on Windows, Mac, or Linux. It could even be used on your new custom Linux Ubuntu VM created previously. Let's walk through each option.

Ubuntu Linux: The following will only work if you installed an APT version of Firefox as previously explained. **Launch Firefox then close it**. Open Terminal and enter the following commands, striking enter after each.

```
sudo apt update && sudo apt install -y curl
cd ~/Desktop
curl -O https://inteltechniques.com/data/osintbook10/ff-template.zip
unzip ff-template.zip -d ~/.mozilla/firefox/
cd ~/.mozilla/firefox/ff-template/
cp -R * ~/.mozilla/firefox/*.default-release
cd ~/Desktop && rm ff-template.zip
```

The next chapter explains more about these types of commands, but let's digest these for future use. The first command installs curl, which we will use to download a file via Terminal. The second command changes the path within Terminal to our desktop. The next command downloads my custom Firefox template to the desktop. The fourth command decompresses the file and extracts all data. The next command changes our directory to the downloaded decompressed files and the final commands copy our Firefox template file into the appropriate folder and delete the downloaded file.

All of the commands within this chapter (and the rest of this book) are replicated within the online OSINT portal at **https://inteltechniques.com/osintbook10**. When prompted, enter a username of "**osint**" and password of "**book1443**" (without quotes). This will provide access to all of the digital resources within the book, and we will be using this portal often as we work through various strategies. I will also place any updates within this portal.

Windows & Mac: While Terminal commands are possible, and will be used within the Mac and Windows OSINT machine chapter, I find it easier to copy the profile based on the official tutorial presented within the previous page. You could also replicate these steps within Linux if you experienced any difficulty with the previous steps. Conduct the following.

- Download the file hosted at https://inteltechniques.com/data/osintbook10/ff-template.zip.
- Save it to your Desktop. Enter a username of "**osint**" and password of "**book1443**" (without quotes) if required.
- Extract the data by double-clicking it and saving the ff-template folder to your Desktop.
- Enter the folder, select all of the files, right-click, and select "Copy".
- Open Firefox and click the menu button (three horizontal lines), click "Help", and then select "Troubleshooting Information". The Troubleshooting Information tab will open.
- Under the "Application Basics" section, click on Open (or Show) Folder (or Directory). A window with your profile files will open. Close Firefox, but leave this window open.
- Paste the content of the new folder on your removable drive into this folder. Overwrite any files when prompted. Restart Firefox.

When you re-open Firefox, you should now see all the extensions installed and configured within the menu. All privacy and security settings should be applied and you are ready to begin your usage. If the extensions are missing, close Firefox and re-open. As a final reminder, these actions overwrite any existing bookmarks, extensions, settings, and other configurations. This tutorial is always preferred within a new instance of Firefox, such as your new Ubuntu VM. Figure 3.09 displays my final Firefox build.

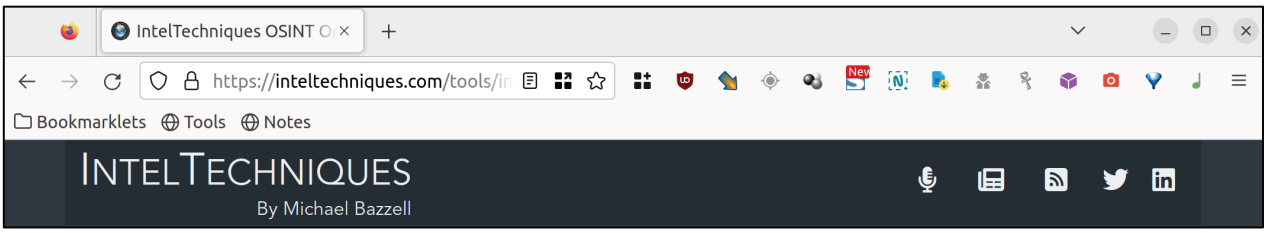


Figure 3.09: A custom Firefox build with Add-ons and tools.

Bookmarklets

In several editions prior to this version, I presented numerous bookmarklets for use with Facebook. They slowly lost value as Facebook constantly changed their code. Today, many bookmarklets can add features to the various websites we visit. These short lines of code do not open any specific web pages. Instead, they execute commands within the website you are currently viewing. They are stored as bookmarks within any browser, but never navigate away from the current page. Let's conduct a demonstration and then explain the usage of each option. In Chapter Ten, I explain the importance of obtaining a Facebook user's ID number. Searching through the source code displays this identifier, but pushing a button is easier. The following code could be added as a bookmarklet within your browser to display a popup notice of the user's Facebook ID number.

javascript:(function()%7Bvar html %3D document.documentElement.innerHTML%3Bvar subhtml %3D html.split('"userID"%3A')%5B1%5D%3Bvar output %3D subhtml.split('%2C"')%5B0%5D%3Balert(output)%7D)()

After saving this code as a bookmark and loading "zuck's" Facebook profile, I launched the bookmark and received a popup displaying "4". I now know his user ID is 4, which can be used for the methods explained later. I prefer to save all bookmarklets within the toolbar of my browser for easy access. The Firefox profile discussed previously already has them all configured for usage. Simply click the "Bookmarklets" toolbar shortcut and choose the most appropriate option for the page you are viewing. Below is a summary of every option.

FacebookID: While on any Facebook profile, this option displays the Facebook User ID.
FacebookGroupID: While on any Facebook group page, this option displays the Facebook Group ID.
FacebookExpand: Attempts to expand all comments on a profile. May be slow and could crash on large pages!
FacebookScroll: Loads and scrolls a Facebook feed before capture of a page.
TwitterScroll: Loads and scrolls a Twitter feed before capture of a page.
InstagramScroll: Loads and scrolls an Instagram feed before capture of a page.
PageScroll-Slow: Slowly scrolls through a static website for video capture.
PageScroll-Fast: Faster scroll through a static website for video capture.
PageScroll-Feed: Scrolls through a feed-style (social network) website for video capture.
ModifiedDate: Displays the date and time of modification to a static web page.
Cache-Google: Opens a Google Cache version of the current website.
Cache-Archive: Opens the Archive.org version of the current website.
Images: Opens a new tab with all images from the current website.
Links: Opens a new tab with all URL links from the current website.
WordFrequency: Displays all words on a page sorted by frequency to easily digest keywords.
Paywall: Opens a blocked news article within a new tab through Outline.com.
Right-Click: Enables right-click functionality on sites which block it.

TextSelect: Enables copy-paste functionality on sites which block it.
BugMeNot: Checks BugMeNot for public credentials to any website (explained later).
Tools: Opens your offline search tools on your Linux Desktop (needs modified for Windows/Mac).

Notes: Opens a blank page which can be used to type or paste notes about the current website. This data is stored within your local storage and is persistent. If you open a new website and click this option, the notes page will be blank. If you return to the previous page which contained notes, clicking this button retrieves those notes pertinent to that URL. This allows you to keep custom notes throughout your entire investigation about each site independently. Rebuilding your VM or Firefox profile removes all note data.

It should be noted that I did not create all of these. The code presented here has been floating around several OSINT websites for many years. As an example, the "ModifiedDate" code was originally released in the Third Edition of this book, but numerous OSINT practitioners now claim credit for it on their own pages. While I cannot determine the original source of each entry here, I thank those who help make these possible.

Chrome/Chromium (google.com/chrome)

I always prefer the previous custom Firefox configuration for my daily investigations. However, I also want the availability of an untouched browser when needed. This might be to appear as a standard user to whatever website I am accessing or to make sure I have not tweaked my browser to the point of blocking some online content. Sometimes, I want an untouched browser to document the exact way an intrusive website appears to the average user without any modification extensions. I believe every good OSINT machine should have a secondary browser option.

Chrome is an excellent browser that is known for being very fast and responsive. Chrome is also very secure by nature, but compromises privacy since Google receives a lot of data about your internet usage. Chrome is based on Chromium, but Chromium is a somewhat less invasive application. Both Firefox and Chromium "sandbox" each tab. Sandboxing restricts the content in that tab to that tab only, preventing it from "touching" other tabs in the browser, or the computer's hardware. This is a very important feature in preventing malware from being installed when you visit a malicious website.

While I always prefer Firefox as my browser for investigations and daily usage, Chromium is my browser used during live training events. This is due to stability when loading dozens of tabs, and your system should have a lot of RAM if you want to take advantage of Chromium's power. For investigative purposes, Chromium can use several of the Add-ons previously mentioned for Firefox. I highly recommend uBlock Origin as previously discussed on any browser that you use, including Firefox, Chrome, Safari, and Opera.

The only time that I use Chromium during an investigation is when I am forced because of a Chromium-specific utility. Before discussing any investigative resources, I suggest you harden your Chromium security. Enter the "Settings" menu; expand the upper-left menu; and consider the following changes.

Privacy and Security: Click the option labeled "Clear browsing data". This button will open a dialogue that allows you to clear any or all of the data stored during your sessions. You may erase information for a period of time ranging from the last hour to "the beginning of time". You may wish to use this function to clear all of your browsing data daily.

Autofill: I recommend disabling every feature in this section. If you have stored form-fill information or passwords in Chromium, I recommend removing any data before conducting investigations.

Chromium Extensions: To install Add-ons in Chromium, navigate to the settings menu and click "Extensions" on the left side of the interface. You will be presented with all the Add-ons that are currently installed in Chromium. I recommend uninstalling any Add-ons that you did not personally install or research for trustworthiness. Most extensions explained for Firefox can be installed in the same manner in Chromium.

We can install stock Chromium within our Ubuntu VM with the following commands.

```
sudo snap install chromium
```

I offer a final note about Chromium. I believe it can be somewhat invasive into your usage and investigations. If you are concerned about the privacy issues surrounding Google's capture of internet traffic occurring within Chromium, consider omitting it from your VM. If you do not need Chromium, and can complete all of your investigations within Firefox, I believe you will possess more privacy. However, having the stock Chromium option can be beneficial for some of the advanced OSINT strategies presented later in this book. I keep Chromium ready on my VM, but rarely open it.

Tor Browser (torproject.org)

Tor is an acronym for The Onion Router. Basically, it allows you to mask your IP address and appear to be browsing the internet from a false location. Normally, when you connect to the internet and browse to a website, that website can identify the IP address that was assigned to you from your internet service provider. This can often identify the city and state that you are in and possibly the business organization where you are currently located. In some instances, it can identify the building you are in if you are using public wireless internet access. The owner of the website can then analyze this information which may jeopardize your investigation. This is one of the many reasons that I recommend the uBlock Origin add-on for Firefox which was explained earlier. uBlock Origin will block most of the analytic code within websites that monitors your information, but it will not stop everything. Occasionally, you may want to change your IP address to make you appear to be someone else in a different country. This is where Tor excels.

For Windows and Mac, the Tor bundle is available for free from the site above and requires minimal default installation. Installation within Linux is discussed in the next chapter. Upon launching, the first task that Tor will complete is to create a connection to a Tor server. This connects you to a server, usually in another country, and routes all of your internet traffic through that server. After the connection is successful, it will load a custom version of the Firefox browser. Now, every website that you visit through this browser will assume you are connecting through this new IP address instead of your own. This provides a layer of privacy to stay hidden from a suspect.

This may be overkill for most investigations. If you are only searching and monitoring common services such as Facebook, Twitter, or YouTube, this service is not needed. If you are visiting personal websites and blogs of a tech savvy hacker, you should consider Tor. When using Tor, you may notice a drastic decrease in the speed of your internet. This is normal and unavoidable. This often improves the longer you are connected. To stop the service, simply close the browser. This will disconnect the Tor network and stop all services. Any activity conducted through the Tor browser is not associated with my real internet connection and appears to be originating in Canada. We can install Tor Browser into our Linux VM with the following command.

```
sudo apt install torbrowser-launcher -y
```

After installation, launch "Tor Browser Launcher" from your applications menu and click "Save & Exit". Then, Launch "Tor Brower" from your applications menu and allow it to complete the installation. I choose the option of "Always connect automatically" and then click "Connect".

Hard Page Refresh

I close this chapter with one last important tip. Internet cache can be a hurdle within your investigations. Any time you visit a website, files are downloaded to your computer. Many sites have instructions embedded into them which tell your browser how it should behave. It may instruct it to load images from your local cache from your computer instead of fetching them again from the web server in order to decrease demand on the network.

You can click the refresh button all day, but it might only renew the page with previously downloaded content. If images on your target website have changed, you may never know. This is where a hard refresh can assist.

The following steps force your browser to reload an entire page from the online resources instead of any locally-stored cache files. I make a habit of doing this often to make sure I am viewing the most recent content.

- Chrome or Firefox (Linux and Windows): Press Shift+CTRL+R.
- Chrome or Firefox (macOS): Press Shift+Command+R.
- Most browsers: Hold Shift while clicking the reload icon.

Your browser might be the most vital piece of your OSINT investigations. Make sure you spend some time customizing it the way you need it. If you choose to build an OSINT VM based on my tutorials within this book, all customizations within this chapter have already been applied.

Firefox Custom Profile Issues

As I was preparing to publish this edition, I noticed that Firefox within Ubuntu 22.04 occasionally reset either the profile or the bookmarks within the browser. If you are missing either the bookmarks or extensions (or both), you can conduct the following at any time to restore them.

Missing Bookmarks

- In Terminal, execute: `cd ~/Downloads`
- `curl -O https://inteltechniques.com/data/osintbook10/bookmarks.html`
- Open the Firefox menu and click "Bookmarks" > "Manage Bookmarks" > "Import and Backup" > "Import Bookmarks from HTML".
- Navigate to your Downloads folder and select the Bookmarks.html file.
- Click "Bookmarks Menu" > "Bookmarks Toolbar" and drag the items to the top "Bookmarks Toolbar".
- Delete any undesired bookmarks.

Missing Profile, Bookmarks, or Extensions

Close Firefox and execute the following within Terminal to delete the current profile and create a new build. Enter "A" if prompted to overwrite files. This can also be completed if you simply want to rebuild your Firefox installation to the default settings from the book.

- `cd ~/.mozilla/firefox/*.default-release`
- `rm -r *`
- `cd ~/Desktop`
- `curl -O https://inteltechniques.com/data/osintbook10/ff-template.zip`
- `unzip ff-template.zip -d ~/.mozilla/firefox/`
- `cd ~/.mozilla/firefox/ff-template/`
- `cp -R * ~/.mozilla/firefox/*.default-release`
- `cd ~/Desktop && rm ff-template.zip`

CHAPTER FOUR
LINUX APPLICATIONS

Hopefully, you now have a functioning Linux Ubuntu virtual machine and customized browsers. That alone provides a very secure environment for your online research. However, possessing customized software applications within your VM would greatly enhance your investigations. Ubuntu provides some basic applications in the default installation, but I want to take that to the next level. In this chapter, we are going to customize your Ubuntu VM with numerous OSINT applications which will automate many investigative tasks.

In Buscador, David and I provided all applications pre-configured for use. This included icons to launch programs and scripts to help execute specific queries. As previously stated, we should not rely on third parties to create and maintain these VM configurations (even my own public resources). The goal here is to teach you how to replicate that work and easily create your own custom VM. If conducting the following tutorials to your original VM, you will only need to take these steps once. Each clone you create will maintain all your hard work.

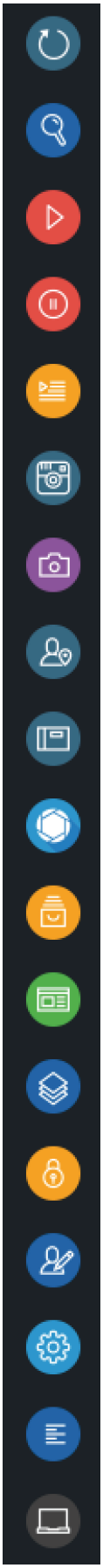
Previous editions of this book included a chapter focused on Windows-only software applications beneficial to OSINT analysis. I now place the emphasis on Linux for three reasons. First, many of those applications have become outdated, or they are no longer maintained, and do not function as originally intended. Second, I want to enforce better security within our investigations. As previously stated, I believe we should only conduct investigations within secure virtual machines which are free from any contamination of other cases. In almost every way, Linux is safer. Finally, there are simply many more OSINT-related applications natively available for Linux than Windows.

We can replicate practically every Windows-only application mentioned in the previous editions while protecting our investigation within a secure VM. There will be a learning curve if you are not familiar with Linux, but the data obtained during your investigations will be superior to the content retrieved from Windows application equivalents. In a later chapter, I offer options to replicate many Linux tools within both Windows and Mac operating systems.

Some of this chapter may seem complicated at first. I promise everything becomes easier as you practice. I will explain the entire manual process of downloading, installing, configuring, and executing each program. Furthermore, I will demonstrate how to create and download scripts in order to easily automate your queries without the necessity of entering the terminal commands and switches. This will be a crash course in Linux, but the lessons learned now will pay off in future usage.

Once we have our systems ready, then we can dive into various online search techniques. All scripts referenced throughout this chapter can be downloaded from my website at **https://inteltechniques.com/osintbook10**. When prompted, enter a username of "**osint**" and password of "**book1443**", without the quotes. **The next chapter automates this entire process which creates a 100% functioning OSINT VM within a few minutes.** The following page presents a summary of the custom applications which we will create together within your VM throughout the next two chapters, including the corresponding icons for each.

Consider this chapter the manual approach in order to understand every detail of our new virtual machine with custom tools. It is designed for those who want to know everything happening behind the scenes. Later, the automated installation will allow you to create machines with almost no effort. **Everything in this chapter could be applied to a dedicated Linux host computer without the use of VMs.** You lose the ability to quickly delete a VM, but you gain performance and a native Linux experience. I keep a dedicated Linux OSINT host for general research and "playing", but I do not use it for investigations. There is no wrong way to learn, and every scenario is unique. Practicing all Linux tutorials within a dedicated Linux host which can be reformatted afterward could be ideal for some readers. For others, VMs will be more appropriate.



Update Tool: Update all software including operating system files, Pip programs, and all custom applications.

OSINT Tools: Launch the custom OSINT Tools discussed later in this section through Firefox.

Video Download Tool: Download single videos or channels with captions, extract all YouTube comments, and display results.

Video Utilities Tool: Play, convert, extract, or compress any video, then display all results. Extract audio from any media.

Video Stream Tool: Easily play, save, convert, and archive live and pre-recorded video streams in real-time.

Instagram Tool: Provide an Instagram username and extract all images using Instalooter and Instaloader, or full content with Osintgram.

Galleries Tool: Provide a URL of a photo gallery and download all images, such as a Flickr account or Tumblr profile, and display all results, or launch RipMe.

Users-Emails Tool: Enter a username or email address and search through various online accounts to discover profiles through various services.

Captures Tool: Enter a single URL or list of URLs and receive data and a screen capture of each.

Domains Tool: Execute a search of any domain through Amass, Sublist3r, Photon, theHarvester, or Carbon14 and display all results.

Metadata Tool: Analyze metadata stored within media files or submit your own files and remove all metadata for safe transmission.

Archives Tool: Launch HTTrack's web service and clone static websites; extract site data from Archive.org; or monitor for website changes from your own machine.

Documents Tool: Enter a URL and receive any documents located on the domain, extracted metadata from the documents, plus a full report of all activity, as explained in Chapter 20.

Breaches-Leaks Tool: Search through breach data, analyze a hash, or enter the IP address of an Elasticsearch server to retrieve all data stored within a specific index, as explained in Chapter 28.

Reddit Tool: Enter a Reddit username and extract current and deleted posts, plus archive entire subreddits.

Spiderfoot: Launch Spiderfoot through your custom Firefox build in order to conduct full queries, as explained in Chapter 27.

Recon-ng: Launch Recon-ng in order to conduct advanced Terminal-based queries, as explained in Chapter 27.

API Tool: Query multiple APIs, including caller ID services, with embedded keys for easy access, as explained in Chapter 28.

Application Installation

Ubuntu possesses a software "store" in which you can point and click through various Linux applications and install them with ease. However, I discourage users from this method. Your choices are minimal and there are likely better alternatives available. Instead, we will use the Terminal for all of our application installations. If you followed the previous tutorials, you may already have the Terminal application in your software Dock within your Ubuntu VM created earlier. If not, you can always find the Terminal application within the "Applications" area of Ubuntu by clicking the nine dots within the Dock on the left of your screen. Open the Terminal application and leave it open while we install some required software. While I encourage readers to replicate this chapter manually, typing all of the commands directly, **I maintain a file with every step within this chapter at https://inteltechniques.com/osintbook10/linux.txt**. If anything should need to be changed, you will find updates there. Let's ease into things slowly.

VLC Media Player

By default, your Windows and Mac operating systems include media players which allow execution of audio and video files. Your default Ubuntu virtual machine may not have this luxury. However, this is easy to correct. VLC is an application which can play practically any media files you throw at it. You could find VLC within the Ubuntu Software application, but I prefer to manually install it. This also provides our first explanation of installation commands in Linux. Within Terminal, type the following commands, pressing return after each.

```
sudo apt update
sudo apt install -y vlc
```

Let's break down this command, as you will see similar instructions throughout this chapter.

`sudo`: This command executes any following text with elevated privileges. It is similar to running a program in Windows or Mac as the administrator. When using this command, you will be required to enter your password. Note that passwords entered within Terminal do not appear as you type them, but they are there. Simply press enter when finished typing. Any additional sudo commands in the same terminal session should not ask for the password again.

`apt update`: This command updates the Ubuntu lists for upgrades to packages which need upgrading, as well as new packages that have just come to the repositories. It basically fetches information about updates from the repositories previously mentioned.

`apt install`: This tells Ubuntu to install a specific software application, such as VLC. Since we added "`-y`" to the command, we do not need to confirm installation during the process.

After you have executed these commands, you should see VLC installed within the "Applications" menu by clicking the nine dots icon in the Dock to the left. You can launch the application within this menu and it should be set as the default option for opening most downloaded media files.

FFmpeg

This is another set of media tools, but these only work within Terminal. We will need them when we start adding utilities to manipulate and download videos. Enter the following into Terminal and press enter after each.

```
sudo apt install -y ffmpeg
```

Video Download Tool

This may be my most-used utility within my custom Linux OSINT VM. The Python script **YouTube-DL** is the backbone which will help us download bulk videos from YouTube and other sources. This is a Python script, and we will need to install "Preferred Installer Program", otherwise known as PIP. Enter the following into Terminal, which will provide the necessary configuration for the current default version of Python (3) in Ubuntu.

```
sudo apt install -y python3-pip
```

We can now install YouTube-DL via "Pip" by entering the following command.

```
sudo pip install -U youtube_dl
```

You may receive a warning about running Pip as Root. This is safe to ignore, and these types of warnings are common. **Since we are creating a disposable VM with many Python programs, I have chosen to always run Pip as Root for maximum functionality**. If this were a super-secret personal machine, I may think differently. Root is optimal for our OSINT VM.

We now have the necessary utilities installed for our media needs. This process is not too exciting, and you will not immediately see anything useful after these actions. However, you are laying the groundwork for the upcoming scripts. The YouTube-DL script is ready to use, but only through the Terminal application. Let's conduct an example in order to understand the features, and then discuss a way to automate queries.

Assume that we are looking for videos of Bob Ross teaching viewers how to paint with oils. After a search on YouTube, you found the Bob Ross video channel located at https://www.youtube.com/user/BobRossInc. Clicking on the "Videos" option on that page presents https://www.youtube.com/user/BobRossInc/videos. This page displays over 600 full episodes of his television program. Clicking one of these videos presents a URL similar to https://www.youtube.com/watch?v=lLWEXRAnQd0. You want to download the video from this page in the best quality possible in order to attach it to your case. The default YouTube-DL command to do this is as follows.

```
youtube-dl https://www.youtube.com/watch?v=lLWEXRAnQd0
```

This will download the video to whichever directory you are in within Terminal. By default, you are likely in your home folder, but that is not a great place to save a video. Therefore, let's change the saved location to your desktop with the following commands.

```
cd ~/Desktop
youtube-dl https://www.youtube.com/watch?v=lLWEXRAnQd0
```

You should now see the video file on your Desktop in Ubuntu. Playback will likely appear poor within the VM, but you could copy that file out to your host for proper playback through the shared folder on your Desktop. When you retrieved this video, you may have noticed that two files actually downloaded (one video and one audio). This is because YouTube-DL has detected that you have FFmpeg installed, and it chose to download the highest quality option for you. If you had not possessed FFmpeg, you would have been given a lower quality version. This is because YouTube presents separate audio and video files to the viewer when a high-quality output is chosen. YouTube-DL works in tandem with FFmpeg to download the best quality option and merge the result into a playable file. This is the best way to extract a single video from YouTube and other video sources. However, the true power is on bulk downloads.

Assume you downloaded the previous file in reference to your investigation. You then see that your target possesses hundreds of videos on his YouTube "Videos" page. You want all of these videos, located at https://www.youtube.com/user/BobRossInc/videos. YouTube-DL can download these with the following.

```
youtube-dl https://www.youtube.com/user/BobRossInc/videos
```

This will take some time to download everything, especially if the videos are long. If you want to cancel this command, press "Ctrl" and "C" on your keyboard at any time. This terminates any process within Terminal and may be beneficial throughout this chapter. Additional options to add to this command include the following.

`--all-subs` (Downloads all closed captioning subtitles associated with the videos)
`--all-formats` (Downloads all versions of a video of any quality)

Downloading all available formats of every video is likely overkill and could take several hours. However, obtaining the closed captioning for each video can be quite beneficial. The following command in Terminal will download the best quality video and audio tracks of the target video, merge them together, and download a text file of each of the embedded closed captioning files.

```
youtube-dl https://www.youtube.com/watch?v=lLWEXRAnQd0 --all-subs
```

The video appears on my Desktop, as that was the last place I had navigated within Terminal. Next to the video is two text files with a .vtt file extension. Below are the first two lines of each. The first represents the English subtitles, and the second are in Spanish.

```
00:00:29.988 --> 00:00:32.108
- Hello, I'm Bob Ross and I'd like to welcome
00:00:29.988 --> 00:00:32.108
- Hola, soy Bob Ross y me gustaría dar la bienvenida
```

Imagine the following scenario. You have a suspect on YouTube with hundreds of videos. You have been tasked to download every video and identify the exact files and timestamps of every time he said the word "kill". One command within YouTube-DL would provide all of the data you need to conduct a search within all text files, and the corresponding video evidence. In a later chapter, I will provide a single query which will search all of these text files at once.

This tutorial could be sufficient for your needs, as it will download the media you desire. It works well on most video host websites. However, I would like to take things a step further. There are a few minor annoyances with this process. The first is that you must manually open Terminal in order to replicate this technique. Next, you must navigate to a desired folder for saving. Finally, you are at the mercy of YouTube-DL on the naming scheme for your saved file. The following may seem complicated at first, but it will simplify our usage of this utility.

In both versions of the Buscador VM, we created scripts which would automate the various tools which we had installed. These were publicly available inside the VM and were the best feature of the OS in my opinion. They simplified the Terminal tools into point-and-click utilities. A YouTube-DL script would be as follows, with minor modifications due to changes in the command options. All scripts mentioned in this chapter are located within my online archive of "vm-files" available at **https://inteltechniques.com/osintbook10**. If prompted, enter a username of "**osint**" and password of "**book1443**", without the quotes. **Later in this chapter, we will incorporate all of the scripts explained in this chapter into our OSINT Original VM using only a few commands.** Before then, we should understand the code and function.

```
#!/usr/bin/env bash
timestamp=$(date +%Y-%m-%d:%H:%M)
url=$(zenity --entry --title "Video Downloader" --text "Target URL:")
youtube-dl "$url" -o ~/Videos/Youtube-DL/"$timestamp%(title)s.%(ext)s" -i --
all-subs
open ~/Videos/
exit
```

The first line (#) explains that this file is an executable script. The second line creates a current timestamp in order to prevent duplicate file names and display the capture time. The third line (url) creates a menu prompt for the user to input the URL of the target video. The next line (youtube-dl) instructs the script to launch the YouTube-DL command with the entered URL; defines a folder for output; and supplies a proper file name. The rest of the script (open) launches the folder containing the final download for convenient access. We will see this structure numerous times throughout this chapter. If you were to save this file to your Desktop as video-download.sh, you could launch it with the following Terminal commands.

`cd ~/Desktop` (Places Terminal in the Desktop directory)
`chmod +x video-download.sh` (Makes the script executable)
`./video-download.sh` (Launches the script)

The result would be the image seen in Figure 4.01 (left). If you entered either of the YouTube URLs used in the previous pages, the script would download the target data to your "Videos" folder. This solves many of the annoyances, but there is still one issue. You still need to open Terminal in order to launch the script. The solution to this is to create a ".desktop" file which we can launch from the Dock. You could conduct the following manual steps or automate this process with the tutorial presented in a moment. The following is provided in order to understand what will happen behind the scenes when we automate our full VM build.

- Open the Applications menu (nine dots) and launch Text Editor.
- Enter the following text and save the file as video-download.desktop to your Desktop.

```
[Desktop Entry]
Type=Application
Name=Video Download Tool
Categories=Application;OSINT
Exec=/home/osint/Documents/scripts/video-download.sh
Icon=/home/osint/Documents/icons/video-download.png
Terminal=true
```

- Close Text Editor, and in Terminal, type: `cd ~/Desktop`
- In Terminal, type: `sudo mv video-download.desktop /usr/share/applications/`
- Enter password if prompted.
- Open the Files program and click on "Documents" in the left area.
- Right-click in the empty area and select "New Folder".
- Create a folder titled "scripts" and enter this folder.
- Drag and drop the video-download.sh file from the Desktop to this folder.
- Click on "Documents" in the left area of the Files program.
- Right-click in the white area and select "New Folder".
- Create a folder titled "icons" and enter this folder.
- Download the Linux "vm-files" archive from https://inteltechniques.com/osintbook10.
- Enter a username of "**osint**" and password of "**book1443**" if required.
- Extract the archive within your new VM.
- Copy all images from the "icons" folder in the archive into the "icons" folder in the VM.
- Open the Applications menu again and scroll to "Video Download". If you do not see it, try searching "Video" within the search field. This should present the icon. You may need to repeat this step later.
- Right-click this program and select "Add to favorites".

You should now see a new icon in your Dock in Ubuntu. Clicking this should present the same dialogue as seen in Figure 4.01 (left). Entering your target URL from YouTube or other services should execute the appropriate options and save your evidence into your Videos folder, as seen in Figure 4.01 (right). The file name includes the date and time of download followed by the title of the video file. The subtitles are also included. The summary of these steps is that we created a desktop shortcut script; moved it to the proper system folder; created a new directory to store our custom scripts; moved our first script into it; downloaded the supplemental files; moved over data; and added our new launch icon into our Dock. If you feel lost on any of this, do not worry.

We will repeat the process in a moment with another program. As you practice entering these commands, the more comfortable you will become using Terminal for other configurations.

While this configuration took some time, it will save you much effort in the future. You can now download single videos, or entire video collections, without opening Terminal or typing a single command. We will replicate similar steps throughout this chapter, and automatically copy over all scripts, shortcuts, and icons with only a few direct commands soon. All configurations and files are available within the "vm-files" archive of custom Linux files on my website at **https://inteltechniques.com/osintbook10**. If required, enter a username of "**osint**" and password of "**book1443**", without the quotes, in order to access these resources. Alternatively, you can create your own scripts with the details provided in these tutorials. Please note that these examples assume you chose "osint" as your username for your VM, as explained in a previous chapter. All of the demos in this chapter will assume this. If you deviated from this example, you will need to replace "osint" with your chosen username within the shortcuts.

While named YouTube-DL, this utility works on most popular video websites. You should have no issues downloading individual or bulk videos from YouTube, Vimeo, and many others. The bulk download option has saved me numerous hours of manually downloading videos individually. I have yet to find any size or file number limitations. This utility is likely my most used program within Linux, aside from browsers.

In 2019, I discovered a potential issue with this script. Since many video websites are now offering "Ultra-HD" 4K resolution videos, the file size can be quite large. Downloading an entire channel can take hours and the files may contain over a terabyte of data. While I appreciate that YouTube-DL always defaults to the highest resolution video available, I may want to choose a lower resolution when downloading hundreds of files. The following command executes YouTube-DL; queries our target video; downloads the best quality video up to a resolution of 720p; downloads the best quality audio available; merges the two streams into one media file; and downloads all subtitles. Note that this is one command which requires two lines for display.

```
youtube-dl https://www.youtube.com/watch?v=lLWEXRAnQd0 -f
'bestvideo[height<=720]+bestaudio' --all-subs
```

You could enter this command manually when needed, but I prefer to add it as an option within our script. The following script provides a choice of best quality or best quality up to 720p.

```
#!/usr/bin/env bash
opt1="Best Quality"
opt2="Maximum 720p"
timestamp=$(date +%Y-%m-%d:%H:%M)
videodownloadmenu=$(zenity  --list  --title "Video Downloader" --text "Choose
Quality" --radiolist  --column "" --column "" TRUE "$opt1" FALSE "$opt2")
case $videodownloadmenu in
$opt1 )
mkdir ~/Documents/Videos/ && mkdir ~/Documents/Videos/Youtube-DL/
url=$(zenity --entry --title "Best Quality" --text "Enter Target URL")
youtube-dl "$url" -o ~/Documents/Videos/Youtube-DL/"$timestamp-%(title)s.%(ext)s"
-i --all-subs
open ~/Documents/Videos/Youtube-DL/
exit;;
$opt2 )
mkdir ~/Documents/Videos/ && mkdir ~/Documents/Videos/Youtube-DL/
url=$(zenity --entry --title "Maximum 720p" --text "Enter Target URL")
youtube-dl "$url" -o ~/Documents/Videos/Youtube-DL/"$timestamp-%(title)s.%(ext)s"
-i -f 'bestvideo[height<=720]+bestaudio' --all-subs
open ~/Documents/Videos/Youtube-DL/
exit;;
esac
```

If you saved this script and titled it "video-download.sh", you could launch it the same way as the previous option. It would provide a menu which allows a choice between the two commands previously explained. Figure 4.02 (left) displays the selector window when the script is launched and Figure 4.02 (right) displays the URL entry menu after a choice has been selected. Much of this script is similar to the previous example. However, there are some unique differences. In the first script, we only gave it one task. In this script, we have multiple options. If this seems overwhelming, please don't worry. Once we get to our final video download script, I will explain every section and its function. Please remember that this chapter is designed to explain the benefits of automated steps. Complete understanding of bash scripts is not required in order to use these new programs or continue with the tutorials throughout the remainder of the book.

You could either launch this file from within Terminal every time or repeat the previous steps to create a desktop icon which can be executed from within the Applications menu. However, I do not recommend these manual steps unless you simply want the experience. As another reminder, in a moment, I will present easy commands which download, extract, copy, and configure every script, icon, and shortcut within this chapter all at once. It will be much easier than performing all of the steps manually. However, it is important to understand the technology behind these scripts in case you ever need to modify the contents due to changes within the various applications or add new features and programs as they are released after this publication. I promise this will all come together easily in the next chapter. You will be able to replicate every detail presented here within only a few minutes.

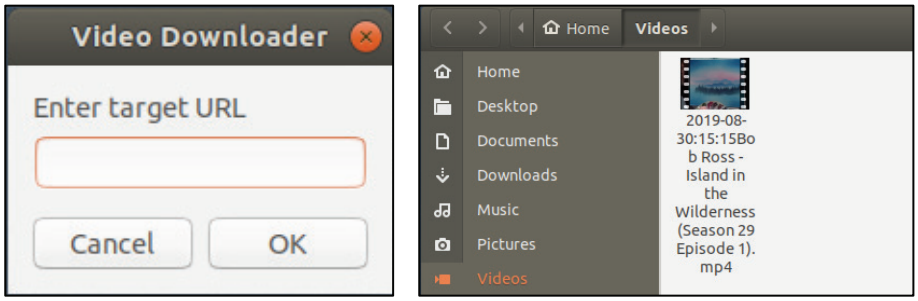


Figure 4.01: A Video Downloader prompt (left) and result (right).

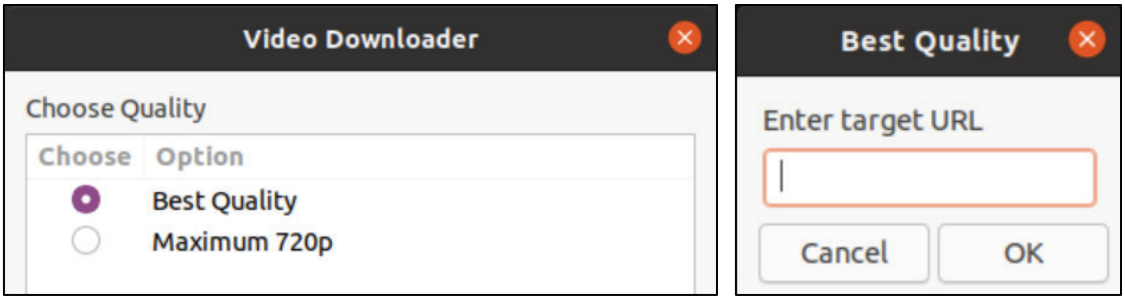


Figure 4.02: The Video Downloader selection and URL entry menu.

I like the way this video download script is coming along, but I want to add more features. I want an alternative faster download option; features which download all the comments from a YouTube video; and the ability to export data about videos, channels, and playlists. For all of that, we will need yt-dlp, as explained next.

yt-dlp (github.com/yt-dlp/yt-dlp)

YouTube-DL has been a staple in my OSINT work for many years. In 2021, I witnessed weekly updates turn into months without any patches. Since YouTube changes their own services regularly, we need alternatives for downloading their content. This is where yt-dlp can be helpful. This application is a fork of the original YouTube-DL, but it has a much more active community and regular updates. Today, I rely on yt-dlp for most tasks, but keep YouTube-DL available as a backup. Install yt-dlp with the following command.

```
sudo pip install -U yt-dlp
```

The previous commands for YouTube-DL function within this option. Our previous example is as follows.

```
yt-dlp https://www.youtube.com/user/BobRossInc/videos
```

The application yt-dlp now includes a feature to download other data associated with a YouTube video, including comments, subtitles, and metadata. Assume your target YouTube video is located online at https://www.youtube.com/watch?v=AFDU_R6ShmI. You can download all data associated with the video with the following commands.

```
yt-dlp https://www.youtube.com/watch?v=AFDU_R6ShmI --write-comments
yt-dlp https://www.youtube.com/watch?v=AFDU_R6ShmI --write-subs en
yt-dlp https://www.youtube.com/watch?v=AFDU_R6ShmI --write-info-json
```

These commands target the desired video page; load all of the data requested; export it to a JSON text file; then download the video file wherever your current Terminal path is set. If you only want to download the data without the entire video, you can add "--skip-download" to your command, as follows.

```
yt-dlp --skip-download https://www.youtube.com/watch?v=AFDU_R6ShmI --write-
comments
```

All files downloaded will be in JSON format. While the files can be opened within any text editor, the format will be difficult to digest. **I always recommend opening these JSON files with the Firefox browser.** Consider the following text generated by the comment extraction functionality of yt-dlp.

```
"text": "It looks like the newest version is the 8th edition:
https://inteltechniques.com/book1.html\n\nHere is a link to more information
on OSINT certification: https://www.giac.org/certification/open-source-
intelligence-gosi", "timestamp": 1635379200, "time_text": "1 year ago",
"like_count": 2,
```

Now, compare the same comment as seen through Firefox in Figure 4.03.

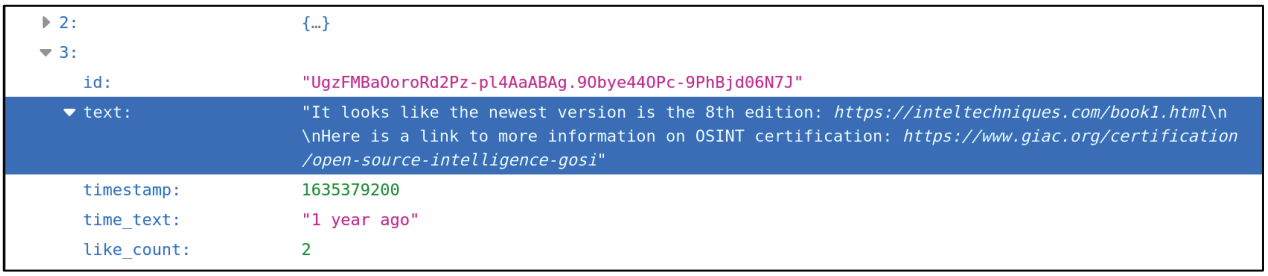


Figure 4.03: Archived YouTube comments viewed through Firefox.

I have added menu dialogue options for all of these tasks into our final video download script, which is presented in its entirety after the images, with a full explanation on the following page. It takes advantage of the best aspects of both YouTube-DL and yt-dlp. We also have redundancy in case a service stops working. Upon launch, you will see the menu in Figure 4.04 (left). If choosing the "Export YT Comments" option, you will then see the entry option within Figure 4.04 (right). Notice that this dialogue asks for the entire URL. Figure 4.05 displays archived YouTube metadata via Firefox, after executing the "Export YT Metadata" option within the menu. Remember that it is always best to open the JSON files within Firefox instead of the default text editor.

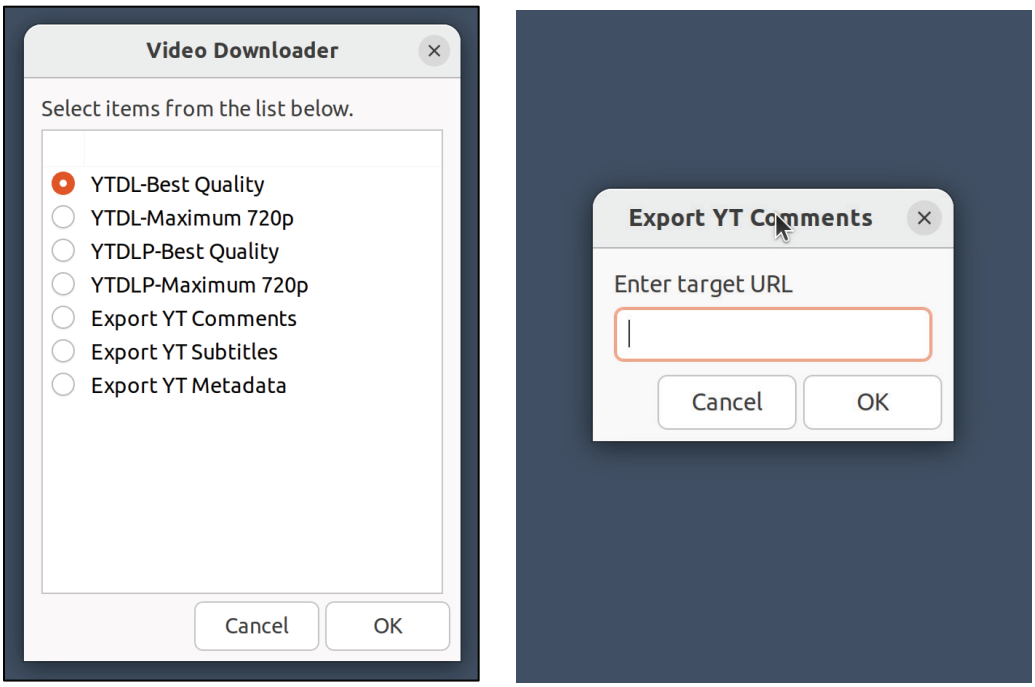


Figure 4.04: The final Video Downloader script dialogues.

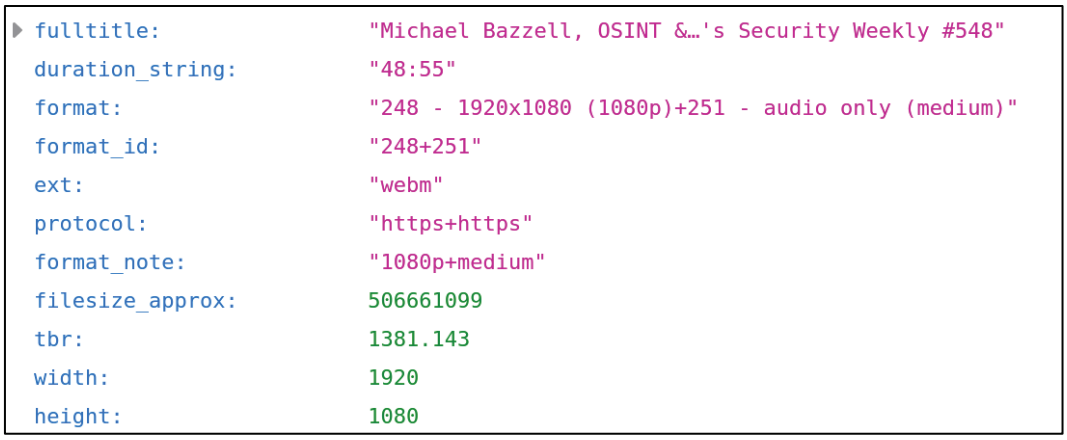


Figure 4.05: YouTube metadata viewed through Firefox.

```
#!/usr/bin/env bash
opt1="YTDL-Best Quality"
opt2="YTDL-Maximum 720p"
opt3="YTDLP-Best Quality"
opt4="YTDLP-Maximum 720p"
opt5="Export YT Comments"
opt6="Export YT Subtitles"
```

```
opt7="Export YT Info"
timestamp=$(date +%Y-%m-%d_%H_%M)
videodownloadmenu=$(zenity  --list  --title "Video Downloader" --radiolist  --
column "" --column "" TRUE "$opt1" FALSE "$opt2" FALSE "$opt3" FALSE "$opt4"
FALSE "$opt5" FALSE "$opt6" FALSE "$opt7" --height=400 --width=300)
case $videodownloadmenu in
$opt1 )
mkdir ~/Documents/Videos/ && mkdir ~/Documents/Videos/Youtube-DL/
url=$(zenity --entry --title "Best Quality" --text "Enter Target URL")
youtube-dl "$url" -o ~/Documents/Videos/Youtube-DL/"$timestamp-
%(title)s.%(ext)s" -i --all-subs
open ~/Documents/Videos/Youtube-DL/
exit;;
$opt2 )
mkdir ~/Documents/Videos/ && mkdir ~/Documents/Videos/Youtube-DL/
url=$(zenity --entry --title "Maximum 720p" --text "Enter Target URL")
youtube-dl "$url" -o ~/Documents/Videos/Youtube-DL/"$timestamp-
%(title)s.%(ext)s" -i -f 'bestvideo[height<=720]+bestaudio' --all-subs
open ~/Documents/Videos/Youtube-DL/
exit;;
$opt3 )
mkdir ~/Documents/Videos/ && mkdir ~/Documents/Videos/Youtube-DL/
url=$(zenity --entry --title "Best Quality" --text "Enter Target URL")
yt-dlp $url -o ~/Documents/Videos/Youtube-DL/"$timestamp-%(title)s.%(ext)s" -i
--all-subs
open ~/Documents/Videos/Youtube-DL/
exit;;
$opt4 )
mkdir ~/Documents/Videos/ && mkdir ~/Documents/Videos/Youtube-DL/
url=$(zenity --entry --title "Maximum 720p" --text "Enter Target URL")
yt-dlp "$url" -o ~/Documents/Videos/Youtube-DL/"$timestamp-%(title)s.%(ext)s"
-i -f 'bestvideo[height<=720]+bestaudio' --all-subs
open ~/Documents/Videos/Youtube-DL/
exit;;
$opt5 )
mkdir ~/Documents/Videos/ && mkdir ~/Documents/Videos/Youtube-DL/
url=$(zenity --entry --title "Export YT Comments" --text "Enter Target URL")
cd ~/Documents/Videos/Youtube-DL/
yt-dlp "$url" -o ~/Documents/Videos/Youtube-DL/$timestamp-comments --write-
comments --skip-download
open ~/Documents/Videos/Youtube-DL/
exit;;
$opt6 )
mkdir ~/Documents/Videos/ && mkdir ~/Documents/Videos/Youtube-DL/
cd ~/Documents/Videos/Youtube-DL/
url=$(zenity --entry --title "Export YT Subtitles" --text "Enter Target URL")
youtube-dl --all-subs --skip-download --write-auto-sub "$url"
open ~/Documents/Videos/Youtube-DL/
exit;;
$opt7 )
mkdir ~/Documents/Videos/ && mkdir ~/Documents/Videos/Youtube-DL/
url=$(zenity --entry --title "Export YT Info" --text "Enter Target URL")
cd ~/Documents/Videos/Youtube-DL/
yt-dlp "$url" -o ~/Documents/Videos/Youtube-DL/$timestamp-info --write-
comments --skip-download
open ~/Documents/Videos/Youtube-DL/
exit;;esac
```

The following explains the key elements.

- `#!/usr/bin/env bash`: This identifies the file as an executable script.
- `opt1="YT-DL Best Quality"`: This provides selectable options for the menu and identifies the text which will be displayed for each entry.
- `timestamp=$(date +%Y-%m-%d:%H:%M)`: This creates a timestamp to prevent duplicate file names.
- `videodownloadmenu=$`: This creates the menu and an identity for it.
- `zenity`: This launches Zenity, which provides the graphical dialogues.
- `--list --title "Video Downloader" --radiolist --column "" --column "" TRUE "$opt1" FALSE "$opt2" FALSE "$opt3" FALSE "$opt4" FALSE "$opt5" FALSE "$opt6" FALSE "$opt7")`: This creates a menu which displays the seven options visible in Figure 4.04. The "True" command selects a default option while "False" commands are unselected.
- `case $videodownloadmenu in`: The case statement is used to simplify things when you have multiple different choices. It is likely overkill for our scripts, but no harm in tidiness.
- `$opt1 )`: This begins the command for the first option.
- `url=$(zenity --entry --title "Best Quality" --text "Enter target URL")`: This instructs the script to create a menu and prompt the user for the URL of the target video.
- `youtube-dl`: This is the command to launch YouTube-DL.
- `"$url"`: This instructs the script to use the text which was input previously.
- `-o ~/Videos/Youtube-DL/"$timestamp%(title)s.%(ext)s"`: This finishes the command to output the evidence and title the file based on timestamp and actual video name.
- `open ~/Videos/`: This opens the Videos folder to display the evidence.
- `exit;;esac`: This instructs the script to exit after completion and closes the "case".

This final script offers the following features within your Linux VM without the need to open Terminal or enter any commands. The Desktop shortcut explained earlier would execute this script from your Applications menu.

- Download a video, entire playlist, or channel in the best possible resolution with YouTube-DL.
- Download a video, entire playlist, or channel in the best resolution up to 720p with YouTube-DL.
- Download a video, entire playlist, or channel in the best possible resolution with yt-dlp.
- Download a video, entire playlist, or channel in the best resolution up to 720p with yt-dlp.
- Export all comments from any YouTube page.
- Export all subtitles from any YouTube page.
- Export all text information from any YouTube page to a text file.

Fragmented Videos

It is very common for websites to present embedded videos which consist of hundreds of small pieces of streaming video frames. These files load seamlessly while viewing a video, but this delivery method can make the download process difficult. Some sites do this to prevent copying of proprietary content, while others eliminate unnecessary download of large single files if the video is stopped before the end. While YouTube-DL and yt-dlp are great programs, they are not perfect. Submitting a URL which contains a fragmented video is likely to fail. Therefore, we need to understand a manual way to approach this issue. Consider the following example. There is an embedded video at https://www.axs.tv/channel/movember/video/billy-gibbons-1/. Submitting this URL to the tools results in an error because they cannot properly detect the video.

Before playing the video, right-click within your Firefox or Chrome browser and select "Inspect". Click the "Network" tab within the new window at the bottom of your browser, then click the play button within the video. You should see numerous connections, similar to those in Figure 4.06.

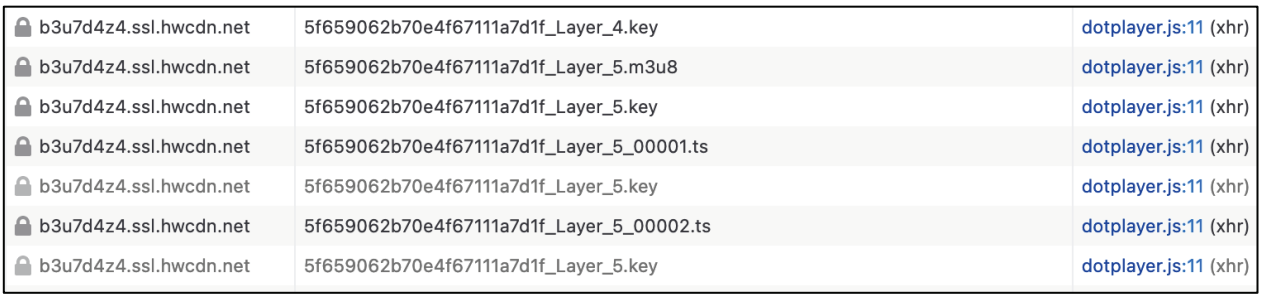


Figure 4.06: Fragmented video files.

The ".ts" files are small video fragments which are being streamed to your browser. This target video URL contains hundreds of these files. Instead of targeting the video fragments, we want the "m3u8" file which acts as a playlist for all of the video pieces. An example is displayed above in the second row. Right-click on this option; highlight "Copy"; and select "Copy URL". Paste this URL into your video download tool. I prefer the yt-dlp option. In this example, my URL is as follows.

https://b3u7d4z4.ssl.hwcdn.net/files/company/5e98e1f8bb2ee152df0229b2/assets/videos/5f659062b70e4f67111a7d1f/vod/5f659062b70e4f67111a7d1f_Layer_5.m3u8

Since you are providing a playlist of these video fragments to the download tool, the software can identify the list of videos and reconstruct them into a single video file. The previous URL resulted in a one-hour single video in MP4 format without any sign of spilt files. I rely on this technique weekly. It works for 99% of streaming videos, but I occasionally find a difficult target. I can usually bypass most download restrictions with the next technique.

Blocked Downloads

Some services generate much effort toward prevention of video downloads from their sites. If the previous download tactics resulted in errors, you may have encountered a blocked file or stream. In most scenarios, you can bypass this with a modified "referer". Consider the following example and modify for your own usage. In 2021, I was contacted by a former colleague who had a unique request. He was trying to download an old episode of the television show "Live PD" as part of an investigation. The episode originally aired in 2016 and pirated copies on torrent sites had all disappeared. It was the 56th episode of the first season and he had exhausted his search. The following query on Google displayed several sites which claimed to possess the video.

"live pd" "s01e56" "watch"

The first page of results loaded a video player but encountered an error when the stream was attempted. Basically, the video file no longer existed so it could not be streamed within the browser. I finally found the following website which would play a video stream of the target episode.

https://123watchmovies.co/episode/live-pd-season-1-episode-56/

I could watch the video, but submitting this URL to our script resulted in an "unable to download webpage" error. The Firefox plugin previously explained could not detect the actual video stream URL because this site embeds the video within a custom player which intentionally makes the download process difficult. I right-clicked the page while the video was streaming; selected the "Inspector" option; clicked the "Network" tab; and sorted the results by "Transferred" size. I immediately saw the following file titled "v.mp4" streaming through the browser and becoming a large file.

https://t13.gomoplayer.com/vxokfnh4w6alavf4ercyvnw4q6xevr24ybj4eyfwtedjogbrwjuewms52v2a/v.mp4

I submitted this new URL to the video download tool, but the result was a small empty video file without any content. This is because the target website includes a strict policy which prevents download of content from any source besides an approved URL. While this site is allowed to stream the video, I would be blocked if I tried to stream it on my own site. Therefore, our video download attempt appears to be an unauthorized source. This is becoming more common. I clicked on the "v.mp4" file within the Inspector menu in my browser and looked for the "Referer" field. It appeared as follows.

Referer: https://gomoplayer.com/

This instruction tells the target website that gomoplayer.com is referring its own content into its embedded player software and the video should be allowed to play on this site. We now know the required referer and can emulate this within our own computer. The following command was entered into Terminal.

```
cd ~/Downloads
yt-dlp https://t13.gomoplayer.com/vxokfnh4w6alavf4ercyvnw4q6xevr24ybj4ey
fwtedjogbrwjupgfk52v2a/v.mp4 --referer https://gomoplayer.com/
```

This changes our working directory to the Downloads folder; launches the yt-dlp software; provides the direct URL of the target video; and informs the service that the original referer is the same option as required (even though this is not true). After executing this command, the 750mb video file downloaded to my computer. I have successfully used this technique within many investigations. The next time your download is blocked, consider replicating this process with the data present within your own investigation.

YouTube-DL vs yt-dlp

At the time of this writing, YouTube was throttling video download speed to all video download utilities. They included code within their systems which would detect video download; throttle the speed to less than 50 kbps; and cause frustration to those trying to archive content. A five-minute video might require over an hour to download through YouTube-DL. Fortunately, yt-dlp updated their software to bypass this restriction. The same video download completed in a few seconds through their software. This is why I insist on access to both options. By the time you read this, the results may be reversed. Today, I always choose the yt-dlp options, but tomorrow could change my preference. Make sure you are comfortable with all options, and be sure to document the tool used in your final report.

IMPORTANT: I realize this is getting quite technical, and we are very early in the book. None of the steps in this chapter are required to complete the tutorials within the rest of the book. However, having your Linux VM customized and ready to go will be beneficial. In the previous example, you created a utility which will download videos from the internet. Once you see how this works and understand the benefits, you may not want to settle for the various web-based options presented later. **For convenience, let's go ahead and configure all of the custom scripts presented within this book. The following Terminal commands download, extract, relocate, and configure every Linux script, icon, and shortcut presented throughout this entire book. Copy and paste these commands from your portal at https://inteltechniques.com/osintbook10/.**

```
cd ~/Desktop
sudo apt install -y curl
curl -O https://inteltechniques.com/data/osintbook10/vm-files.zip
unzip vm-files.zip -d ~/Desktop/
mkdir ~/Documents/scripts
mkdir ~/Documents/icons
cd ~/Desktop/vm-files/scripts && cp * ~/Documents/scripts
cd ~/Desktop/vm-files/icons && cp * ~/Documents/icons
cd ~/Desktop/vm-files/shortcuts && sudo cp * /usr/share/applications/
cd ~/Desktop && rm vm-files.zip && rm -rf vm-files
```

You should now possess shortcuts and scripts for all of the techniques we are discussing, but the education is far from over. Let's conduct an additional example of creating your own execution script from the beginning. Next, we will tackle various video utilities. In the following chapter, we will issue a couple of commands and sit back while this entire chapter is automated. From now on, I will abbreviate the steps.

I present an important warning before we begin installing numerous applications via Terminal. Some will fail to install completely on the first attempt. This is often due to a broken or missing dependency which is required by the application. Sometimes, repeating the steps solves the issue, while other scenarios require us to wait until the program is updated by the developer. There is no harm in repeating the steps presented here if you run into complications.

You may see some Pip errors or warnings displayed while you install the following apps. These are often dependency version conflicts, and typically do not impact the functions of the program. While we will take steps to eliminate as many errors as possible, yellow warning and red error text within Terminal is common. Launching the "Updates" script, which is explained within the next chapter, often resolves most issues.

Video Utilities

The utilities in this section execute the previously installed tool FFmpeg. This powerful Terminal utility can manipulate locally-stored videos to assist with investigations. We will create scripts which will provide the following functions.

- **Play a video:** This option will force FFmpeg to attempt to play any video file with multiple video codecs. This will often play videos that would not otherwise play using standard media players such as Windows Media Player and VLC. This will also play many surveillance videos without the need for third-party programs. I once used this technique to play a corrupt video file which captured a homicide, but standard video players could not understand the data.
- **Convert a video to MP4:** This option simply converts any target video to a standard MP4 format. This is beneficial when the target video possesses a unique codec that prohibits playing universally. If the above option can play the video, this option can convert it so that any computer should be able to play it natively. I often used this to send copies of videos to prosecutors forced to use Windows computers.
- **Extract video frames:** This is likely the most used utility within this set of applications. After supplying a target video, this tool will extract the still images from the video. The result is a folder of uncompressed bitmap (bmp) image files depicting practically every frame of the video. This is beneficial when close examination of frames is necessary. I include this every time a crucial video will be used during prosecution of a crime.
- **Shorten a video (Low activity):** This version of the script removes single frames of a video that appear extremely similar to the frame preceding it. In other words, it takes out all of the frames which are the same (no action) and only leaves desired activity. This is great for surveillance video which contains hours of one scene.
- **Shorten a video (High activity):** This version of the script is identical to the previous with one exception. It is a bit more forgiving for videos with high activity. This might be outdoor surveillance of people walking in the distance or a video with a time counter printed within the feed. If the previous option does not remove enough content, this version will be a bit more aggressive. Both of these work well with surveillance video recorded in real-time (no motion detection).
- **Extract audio:** This option extracts the raw audio file out of any video, converts it to a 320k MP3 file, and saves the file. I have used this to extract audio from video confessions and interviews, and it works well on any online videos downloaded from the internet.
- **Rotate video:** You may encounter a video which has been captured from a mobile device and is rotated 90 degrees clockwise. This option allows us to rotate the video counterclockwise to a traditional view, and flips the video vertically to generate a traditional view of a cellular video. This is beneficial during courtroom presentation, especially when a jury is involved.

First, let's take a look at the manual commands which achieve the desired goals of each option. Assume that you possess a video titled "evidence.mpg" on your Desktop. After changing to your Desktop directory within Terminal (`cd ~/Desktop`), the following commands would be used.

Play a video:
`ffplay evidence.mpg`
This command simply plays the video inside a new window at full resolution.

Convert a video to MP4:
`ffmpeg -i evidence.mpg -vcodec mpeg4 -strict -2 evidence.mp4`
This command converts the video to a standard format and saves it alongside the original.

Extract video frames:
`ffmpeg -y -i evidence.mpg -an -r 10 img%03d.bmp`
This command saves the still images from the video to a new folder alongside the original.

Shorten a video (Low activity):
`ffmpeg -i evidence.mpg -strict -2 -vf "select=gt(scene\,0.003),setpts=N/(25*TB)" evidence.mp4`
This command converts the video to a version displaying activity, saving it alongside the original.

Shorten a video (High activity):
`ffmpeg -i evidence.mpg -strict -2 -vf "select=gt(scene\,0.005),setpts=N/(25*TB)" evidence.mp4`
This command converts the video to a version displaying activity, saving it alongside the original.

Extract audio:
`ffmpeg -i evidence.mpg -vn -ac 2 -ar 44100 -ab 320k -f mp3 evidence.mp3`
This command converts the video to a standard audio mp3 and saves it alongside the original.

Rotate video:
`ffmpeg -i evidence.mpg -vf transpose=0 evidence.mp4`
This command rotates the video 90 degrees counterclockwise and saves it alongside the original.

As you can see, these commands can be lengthy and technical. We want to create one single custom script which will allow us to choose which service we want to execute. This is a bit different than the previous example, as it will combine several options into one dialogue and allow the user to select a video upon execution. The following displays the script I use within the OSINT VM. Afterward, I explain some of the new portions which are important to our usage. This exact script is titled video-utilities.sh within your "vm-files" download.

```
#!/usr/bin/env bash
zenity --info --text="The next window will prompt you for a local media file." --
title="Video Utilities" --width=200
ffmpeg_file=$(zenity --file-selection --title "Video Utilities")
timestamp=$(date +%Y-%m-%d:%H:%M)
opt1="Play a video"
opt2="Convert a video to mp4"
opt3="Extract video frames"
opt4="Shorten a video (Low Activity)"
opt5="Shorten a video (High Activity)"
opt6="Extract audio"
opt7="Rotate video"
ffmpeg=$(zenity  --list  --title "Video Utilities" --radiolist  --column "" --
column "" TRUE "$opt1" FALSE "$opt2" FALSE "$opt3" FALSE "$opt4" FALSE "$opt5"
FALSE "$opt6" FALSE "$opt7"  --height=400 --width=300)
case $ffmpeg in
```

```
$opt1 )
ffplay "$ffmpeg_file"
exit;;
$opt2 )
mkdir ~/Documents/Videos/
ffmpeg -i "$ffmpeg_file" -vcodec mpeg4 -strict -2
~/Documents/Videos/$ffmpeg_file.mp4
open ~/Documents/Videos/
exit;;
$opt3 )
mkdir ~/Documents/Videos/ && mkdir ~/Documents/Videos/$timestamp-frames
ffmpeg -y -i "$ffmpeg_file" -an -r 10 ~/Documents/Videos/$timestamp-
frames/img%03d.bmp
open ~/Documents/Videos/$timestamp-frames/
exit;;
$opt4 )
mkdir ~/Documents/Videos/
ffmpeg -i "$ffmpeg_file" -strict -2 -vf "select=gt(scene\,0.003),setpts=N/(25*TB)"
~/Documents/Videos/$timestamp-low.mp4
open ~/Documents/Videos/
exit;;
$opt5 )
mkdir ~/Documents/Videos/
ffmpeg -i "$ffmpeg_file" -strict -2 -vf "select=gt(scene\,0.005),setpts=N/(25*TB)"
~/Documents/Videos/$timestamp-high.mp4
open ~/Documents/Videos/
exit;;
$opt6 )
mkdir ~/Documents/Videos/
ffmpeg -i "$ffmpeg_file" -vn -ac 2 -ar 44100 -ab 320k -f mp3
~/Documents/Videos/$timestamp.mp3 |
open ~/Documents/Videos/
exit;;
$opt7 )
mkdir ~/Documents/Videos/
ffmpeg -i "$ffmpeg_file" -vf transpose=0 ~/Documents/Videos/$timestamp.mp4
open ~/Documents/Videos/
exit;;esac
```

- The second "zenity" line prompts the user with a dialogue telling them to pick a file in the next window. The user then sees a file selection window which can be used to browse to the target video. The script then executes the commands as previously explained. It then defines that video within the script and obtains a timestamp for unique naming.
- The "opt" section identifies the options for the user, which match our previous manual entries. The script then displays the utility selection window to users which allows them to choose the appropriate action to take against the target video, such as play, convert, extract, etc.
- The "mkdir" entries ensure that we have a folder titled "Videos" present within our Documents folder. If one exists, this command will do nothing, if it does not, the script will create the directory.
- Each "ffmpeg" command outputs our new data into the Videos folder for easy access.
- Finally, the "open" commands display the Video folder so we can immediately access our generated content.

This utility is often overlooked in favor of scripts which conduct actions directly toward online content. However, it is a vital tool in our arsenal. It is the only utility which interacts with media files already stored locally within our operating system. This can be important when we need to modify something originally obtained from the internet. When I was conducting online investigations full-time, this utility was launched daily.

If following along manually, you will find this script within the "vm-files" archive you previously downloaded. You should already have this script in the "scripts" folder in your "Documents" folder. This should be the same location where you saved the YouTube-DL script. Next, we need a shortcut to open this script, just as we did previously. The previous steps moved a file titled "video-utilities.desktop" to your "/usr/share/applications" folder, and a shortcut should be visible in your Application menu. Below is the content of the video-utilities.desktop file.

```
[Desktop Entry]
Type=Application
Name=Video Utilities Tool
Categories=Application;OSINT
Exec=/home/osint/Documents/scripts/video-utilities.sh
Icon=/home/osint/Documents/icons/video-utilities.png
Terminal=true
```

This should look very similar to the previous shortcut. Again, this file assumes you have applied the same paths and usernames as my demonstration throughout this chapter. Next, let's add a shortcut icon to our Dock in order to have easy access to this new utility.

- Open the Applications menu again and scroll down to "Video Utilities".
- Right-click this program and select "Add to favorites".

You should now have a new icon in your Dock directly below the previous YouTube-DL option. You can drag and drop these in a different order if desired. The new icon launches a dialogue that first notifies you that "The next window will prompt you for a target media file. Click "Cancel" if entering a URL. After clicking OK, you are presented with a file manager window which prompts you to choose a target video file. This could be a downloaded YouTube video, an unplayable surveillance video, or any other downloaded video content retrieved from any source.

After you select the target video, you are presented with a menu of the options mentioned previously. After choosing the action, the script processes the video and creates the new evidence. All of this happens without opening Terminal or typing any commands. The previous examples explained the details of how these scripts and shortcuts function. The remaining utilities we need to add are quite similar and do not need the same attention to detail. I present a description of each utility; the content of each script and shortcut; the commands to include in your Ubuntu VM, and an example of usage.

Video Stream Tool

The previous utility possesses an option to download a video stream. This can be useful if you know the exact location of the data. In Chapter Three, I presented a Firefox extension titled Stream Detector. On that page, I explained how it detected a video stream URL for a primary video stream from CNN. If we copied and pasted that URL into the "Download Stream" option of the previous script, it would archive the video into MP4 format. This is great, but that feature is unforgiving. If you do not know the exact stream URL, you are out of luck. This is where **Streamlink** can assist. This powerful software can identify, play, and archive many live or pre-recorded video streams by simply providing the URL of the website in which it can be viewed. We will need to install it into our Linux VM via Terminal, but we should first have a conversation about our new installation method.

Python Virtual Environments

The installation of the previous Python applications using "pip" were very basic. However, many applications installed with Pip also install multiple dependencies which are required for each program. Some programs only want specific versions of these dependencies and may not work if newer versions are installed. This brings us to another major deviation from the previous edition. In the past, I lumped all Linux applications within the root operating system, which began to cause dependency conflicts. I no longer configure all applications and

dependencies within the root operating system. Today, I place most Python programs within their own virtual environment using Python venv. I will only explain this procedure once, but we will use it throughout the book. First, let's install the software we need with the following command.

```
sudo apt install -y python3-venv
```

We can now install Streamlink within our newly created Python virtual environment with the following.

```
mkdir ~/Downloads/Programs
mkdir ~/Downloads/Programs/Streamlink
cd ~/Downloads/Programs/Streamlink
python3 -m venv streamlinkEnvironment
source streamlinkEnvironment/bin/activate
sudo pip install streamlink
deactivate
```

Now, let's walk through each of these commands to better understand the strategy.

`mkdir:` Makes a new directory for our software.
`cd:` Changes the directory to our Programs folder.
`python3 -m venv:` Creates a virtual environment with a unique name only for this application.
`source streamlinkEnvironment/bin/activate:` Activates the environment for tasks.
`sudo pip install streamlink:` Installs the application within the virtual environment.
`deactivate:` Stops listening for instructions.

Basically, we created a new isolated folder which will contain the pip installation and all supporting files. We activated the environment before we started installing our software, and the installation was performed in a folder isolated from the other Python applications within our operating system. Once we were finished, we stopped the environment so that nothing else would be placed within it. This prevents many software conflicts, and is especially important as we continue to pour more Python applications into our VM.

We can now use the tool within Terminal. The following command plays the best quality version of a live stream from a Twitch user conducting a cooking lesson through VLC.

```
streamlink https://www.twitch.tv/shroud best
```

Instead of watching the stream, we can capture it to a file named "shroud-stream" within our default directory with the following command.

```
streamlink https://www.twitch.tv/shroud best -o shroud-stream
```

If we wanted to watch and archive the data in real-time, we can use the following command.

```
streamlink https://www.twitch.tv/shroud best -r shroud-stream
```

Finally, when the live stream is over, I can convert the archived data into a standard video file within my Videos folder titled shroud-stream.mp4 with the following command.

```
ffmpeg -i shroud-stream -c copy ~/Videos/shroud-stream.mp4
```

I have created a script titled "video-stream.sh" within the "vm-files/scripts" folder of your download. If you placed this file into your "Documents/scripts" folder and moved the shortcut "video-stream.desktop" from the "vm-files/shortcuts" folder to the "/usr/share/applications" folder as previously explained, you can now replicate all of these steps easily. Open the "video-stream.sh" file if you want to analyze the script.

After the stream is finished, or the user cancels by pressing "Ctrl" and "C", the Videos folder is presented in order to see the evidence file. This file can then be converted using the final menu option. I find this to be the cleanest method of viewing, recording, archiving, and converting live video streams. This utility will be automatically installed and configured using the steps in the next chapter. This utility works well with most live and pre-recorded video streams within popular platforms such as Twitch and Vimeo. However, some sites cause errors. I prefer to use this tool in tandem with the Stream Detector extension previously mentioned. Let's conduct a live demonstration.

After entering the live news video stream from KTLA at https://ktla.com/on-air/live-streaming into the Video Stream Tool, I was notified "No plugin can handle URL". However, when I loaded this site within my custom Firefox build, I could see that the Stream Detector extension captured the M3U8 stream from this live broadcast. I clicked on the stream to copy the link and then re-opened the Video Stream Tool. I chose the option to "Play and Record Live Stream" and entered this new URL. Streamlink immediately began playing and archiving the video since I had provided the direct stream. No video capture utility is perfect, and do not expect this option to work every time. However, it is one of the best options we have. If you investigate targets who stream live video, this is a tool you must possess at all times.

Instagram Tool

There are several independent programs which assist with bulk-download of Instagram data. The first is **Instalooter**, which has been a staple within Buscador since the beginning. Since it occasionally stops working, pending an updated release, we also want **Instaloader**, **Toutatis**, and **Osintgram** available at all times. The results of each are very similar, but it is good to have redundant options. Osintgram is the most powerful of all of them. First, let's install the applications, which all require Pip, with our new Python Virtual Environments. The first command installs Git, which is required by Osintgram and numerous additional programs throughout the chapter.

```
sudo apt install -y git
mkdir ~/Downloads/Programs/Instalooter
cd ~/Downloads/Programs/Instalooter
python3 -m venv instalooterEnvironment
source instalooterEnvironment/bin/activate
sudo pip install instalooter
deactivate
mkdir ~/Downloads/Programs/Instaloader
cd ~/Downloads/Programs/Instaloader
python3 -m venv instaloaderEnvironment
source instaloaderEnvironment/bin/activate
sudo pip install instaloader
deactivate
mkdir ~/Downloads/Programs/Toutatis
cd ~/Downloads/Programs/Toutatis
python3 -m venv toutatisEnvironment
source toutatisEnvironment/bin/activate
sudo pip install toutatis
deactivate
cd ~/Downloads/Programs
git clone https://github.com/Datalux/Osintgram.git
cd Osintgram
python3 -m venv OsintgramEnvironment
source OsintgramEnvironment/bin/activate
sudo pip install -r requirements.txt
deactivate
make setup
```

The Git option for Osintgram is new to our steps. The final Osintgram commands switch into the proper directory; download the Osintgram program; change into the Osintgram directory; create a virtual environment called "OsintgramEnvironment"; activate the environment; install the required dependencies into the virtual environment; and then deactivate the environment. The optional "make" step prompts you for your Instagram username and password which is required for this application. It only needs to be supplied once during installation, and the application will store the credentials. I find the following third-party utilities helpful to the overall function of the program.

```
sudo apt install libncurses5-dev libffi-dev -y
```

Now that you have all programs installed, we can test them. Start with the following commands for Instaloader and Instalooter within Terminal. Please note this is not my account but serves as a good demonstration with minimal content.

```
cd ~/Desktop
instalooter user mikeb
instaloader user mikeb
```

If you wanted to log in to your own Instagram account in order to see additional content which may be restricted otherwise, you could enter the following instead, replacing the content within brackets with your actual credentials.

```
cd ~/Desktop
instalooter login -u [username] -p [password] user mikeb
instaloader --login [username] -p [password] user mikeb
```

Instalooter should have downloaded several individual images to your Desktop, while Instaloader should have created a new folder titled mikeb on your Desktop containing the same images. Next, let's discuss Toutalis. This application provides Instagram account details, often including full email addresses and partial telephone numbers. This extra level of disclosure requires a unique piece of information from our account. Each time you run the command, or use the automated script, you are required to enter your Instagram "Session ID". The following steps will identify this information.

- Log in to an Instagram account from your Firefox browser.
- Right-click an empty area of the page and choose "Inspect".
- Click the "Network" tab in the new menu at the bottom.
- Navigate to any user's Instagram page.
- In the Inspector menu, click on an entry similar to "200 GET www.instagram.com".
- Click the "Cookies" tab in the Inspector menu to the right.
- Scroll down this list and find "sessionid:".
- Copy the alphanumeric entry.

Keep this Session ID somewhere which is easy to access. Since you previously installed the software, we can conduct a test with the following command for the target of "mikeb" and Session ID of 24316:Lh59ygrmY4N:4.

```
toutatis -u mikeb -s 24316:Lh59ygrmY4N:4
```

The result is similar to the following, which I have partially redacted.

```
Full Name: Mike Brandon | userID : 1144153003
Verified : False
Is buisness Account : False
Is private Account: False
Follower: 70 Following : 31
Number of posts: 18
Number of tag in posts : 0
Obfuscated phone       : +44 **** ****77
```

I find this information extremely valuable, and the process only takes a few seconds to complete. You could easily copy and paste these details into your report.

Next, let's look at Osintgram. First, navigate to the software folder with the following command.

```
cd ~/Downloads/Programs/Osintgram/
```

We can conduct numerous types of queries on "mikeb". The following would display this user's photos, stories, comments, captions, followers, followers' email addresses, followers' telephone numbers, followings, followings' email addresses, followings' telephone numbers, and general account information.

```
python3 main.py -c photos mikeb
python3 main.py -c stories mikeb
python3 main.py -c comments mikeb
python3 main.py -c captions mikeb
python3 main.py -c followers mikeb
python3 main.py -c fwersemail mikeb
python3 main.py -c fwersnumber mikeb
python3 main.py -c followings mikeb
python3 main.py -c fwingsemail mikeb
python3 main.py -c fwingsnumber mikeb
python3 main.py -c info mikeb
```

If you received any "challenge" errors, that means Instagram has found your activity to be suspicious. You would need to log in to your Instagram account and confirm the activity as non-malicious. If you execute the Instagram custom tools and receive no results, it may be because of this error. You may need to log in to your account from the OSINT VM first to pacify their servers.

Since these programs all save data in a unique way, our script will need unique options for each. The entire custom script is titled "instagram.sh" within the "vm-files" archive. In all of these options, all data will be saved to folders containing the target username within the folder titles. **Since we have already analyzed these types of scripts within the previous tutorials, I will no longer present them here within this chapter.** You can open any of the scripts within your Linux VM by navigating to the Documents/Scripts folder and double-clicking each. Let's now focus on the techniques.

While these applications may suffice for most needs without the requirement to log in to an active account, I have found that Instagram actively blocks this type of automated behavior. Furthermore, Instagram often delivers images with lower resolution when you are not logged in, while presenting higher quality media when an account is authorized. Therefore, we will add two choices to our script which gives us options to log in to an active account before submitting our commands.

Once you have built your final custom OSINT virtual machine, as explained in the next chapter, you will have all tools ready to launch via the custom scripts and shortcuts. The dialogue presented when choosing the Instalooter (With Login) option demands your username and password for an Instagram account. The script

populates these details in order to optimally perform. Your credentials are stored within your own VM and never transmitted to any service aside from Instagram.

While writing this chapter, both Instalooter and Instaloader downloaded the target data without any modification. However, a second attempt with Instalooter failed. It required Instagram credentials in order to function again, likely due to blockage from our suspicious automated activity. Overall, you will experience much better results from both programs if you authenticate a valid Instagram account. Note that you may need to log in again after a period of inactivity. The "instagram.desktop" file should be available in your Applications menu. Next, let's add a shortcut icon to our Dock in order to have easy access to this new utility.

- Open the Applications menu again and scroll down to "Instagram Tool".
- Right-click this program and select "Add to favorites".

The automated script will automate this entire process; output the results in the appropriate locations in your Documents folder; and open the results upon completion. Note that the last option is to populate your Instagram credentials for use with Osintgram. This is beneficial in the event you created your OSINT VM with the single command explained in the next chapter, which does not add your username and password as we did during the manual installation. In my experience, the options to retrieve email addresses and telephone numbers can take a long time to run, and often fail to finish. I only use these options when my target has a small number of followers or following.

You now have four Instagram applications at your fingertips, and no requirement to manually use Terminal or typed commands. This utility attempts to extract media from a target's Instagram profile. While screen captures of an account through your browser may suffice for investigations, you should consider downloading images directly from the source when possible. Options including credentials will always perform better than those without. In my experience, these four options will never all function when you need them. Just when I think Instalooter and Instaloader are worthless, they save me when Osintgram breaks. It should be your goal to become familiar with each and understand how they can work (and sometimes not work) in tandem.

In previous editions of this book, I presented complete steps to create a similar utility for Twitter using Twint. As of this writing, that utility is no longer maintained due to changes at Twitter which blocked all functionality. By the time you read this, Instagram may be blocking these scripts. OSINT is a constant game of cat-and-mouse. Be sure to monitor my website files, blog, Twitter, or podcast to stay current with any changes. When these tools break for you, they break for me too. We are in this together.

Galleries Tool

I previously explained YouTube-DL as an all-in-one bulk online video download solution. **Gallery-DL** (github.com/mikf/gallery-dl) is a similar option for image collections. You can easily install this within Terminal by entering the following commands, which also provide an isolated Python environment.

```
mkdir ~/Downloads/Programs/Gallery-DL
cd ~/Downloads/Programs/Gallery-DL
python3 -m venv gallerydlEnvironment
source gallerydlEnvironment/bin/activate
sudo pip install -U gallery-dl
deactivate
```

Let's conduct a demonstration. Assume you have located a Flickr account at flickr.com/photos/henrik_lindberg and you want to archive all of the images within a specific album. The following command would download the data to the current folder within Terminal (command is entered as one line).

```
gallery-dl "https://www.flickr.com/photos/henrik_lindberg/
albums/72157630549692722"
```

The result is all 187 images from the photo gallery. This is a very valuable and efficient resource. Gallery-DL currently supports over 150 services including Tumblr, 4chan, Flickr, Imgur, Reddit, and numerous "adult" websites. As a test, I submitted the following command to download all images from my own Twitter profile, which completed in ten seconds. The "Gallery Tool" icon in your custom OSINT VM prompts you for the target URL and executes the command for you. Notice I excluded the quotation marks around the URL and it appeared to work fine. However, including quotation marks is more stable.

```
gallery-dl https://twitter.com/IntelTechniques
```

Next, you might want a graphical solution for your investigation when you encounter a situation which presents a large number of photos. I previously mentioned browser extensions which help automate this process, but they have limits. My preference is always the Gallery-DL, but it may not function on the site which you have targeted. We should always consider other automated options. I have had great success with **RipMe** (github.com/RipMeApp/ripme). This application requires Java to be installed on your machine. I refuse to install Java on any Windows or Mac host due to security concerns, as it adds an additional layer of vulnerability. However, I have no objection to installing it within a Linux virtual machine. Enter the following within Terminal.

```
cd ~/Downloads
sudo apt install default-jre -y
wget https://github.com/ripmeapp/ripme/releases/latest/download/ripme.jar
chmod +x ripme.jar
```

You can now right-click this file within Downloads to launch the program via OpenJDK. However, that is a nuisance to do every time. Instead, right-click the program, and choose "Properties". Click the "Open with" tab and select "OpenJDK Java 11 Runtime". Click the "Set as default button" and close the menu. From now on, double-clicking this program should open it without issue. Fortunately, our custom scripts will bypass this entire requirement.

Once you have the program open, enter any target URL and click "Rip". This application supports image downloads from Imgur, Twitter, Tumblr, Instagram, Flickr, Photobucket, and Reddit. The script for these tools, titled "galleries.sh", and the desktop entry, titled "galleries.desktop", are in your "vm-files" download archive. I use these tools almost every day. As I wrote this section, I was tasked with investigating a hidden locker room video which had surfaced on a porn site. Once I identified a suspect, I was able to download their entire image gallery in a few minutes by clicking the Gallery Tool icon within my finished OSINT VM, which we will rebuild in the next chapter.

Users-Emails Tool

The previous versions of this book provided an automated tool for a Python program called Sherlock. It queried a username and tried to identify online accounts which might be associated with the target. While Sherlock is still valuable, there are additional tools which assist with this type of query. I currently rely on Sherlock, SocialScan, Holehe, WhatsMyName, Blackbird, Maigret, Email2Phone, and Ghunt, all of which are explained here. Let's start with **Sherlock** (github.com/sherlock-project). It is installed with the following commands.

```
cd ~/Downloads/Programs
git clone https://github.com/sherlock-project/sherlock.git
cd sherlock
```

```
python3 -m venv SherlockEnvironment
source SherlockEnvironment/bin/activate
sudo pip install -r requirements.txt
deactivate
```

We must now construct a proper execution of the command. We must be within the program folder in Terminal (~/Downloads/Programs/sherlock/sherlock) and launch Python 3 followed by the script. We must also specify the target username and any additional options. My command is as follows, with an explanation immediately after in parentheses.

```
python3 sherlock.py inteltechniques --csv -o ~/Documents/Report.csv
```

`python3` (Specifies Python version 3.x)
`sherlock.py` (The Python script)
`inteltechniques` (The target username)
`--csv` (Creates a CSV spreadsheet file)
`-o ~/Documents/Report.csv` (Specifies save location and file name)

Note that the version of Sherlock within the custom scripts exports a text file for easier viewing. Sherlock seems to occasionally display false positives, which is always an issue with these types of queries. Next, let's take a look at **SocialScan** (github.com/iojw/socialscan). It can be installed with the following Terminal commands.

```
mkdir ~/Downloads/Programs/socialscan
cd ~/Downloads/Programs/socialscan
python3 -m venv socialscanEnvironment
source socialscanEnvironment/bin/activate
sudo pip install -U socialscan
deactivate
```

Within Terminal, you can now execute queries of usernames and email addresses as follows.

```
socialscan inteltechniques
socialscan inteltechniques@gmail.com
```

It currently only queries a few services, but the on-screen results are reliable. The first query above confirmed that I have accounts on GitHub, Reddit, Snapchat, and Twitter. Next, we have the free utility **Holehe** (github.com/megadose/holehe). This program is more robust than the previous options. It queries dozens of services using several identification strategies such as registered users and password recovery prompts. Installation is completed with the following commands.

```
mkdir ~/Downloads/Programs/holehe
cd ~/Downloads/Programs/holehe
python3 -m venv holeheEnvironment
source holeheEnvironment/bin/activate
sudo pip install -U holehe
deactivate
```

The following command within Terminal queries an email address through all services.

```
holehe test@gmail.com
```

The on-screen results identify online accounts which appear to be associated with the email address. Highlighting these allows for easy copy and paste into a report. I no longer export these as a text file because the results are difficult to read. Next, we have **WhatsMyName** (github.com/WebBreacher/WhatsMyName.git). It is more thorough than the previous options, but takes more time to process, and commonly displays errors which prevent completion. I also typically see numerous false positives. We can install with the following commands.

```
cd ~/Downloads/Programs
git clone https://github.com/WebBreacher/WhatsMyName.git
cd WhatsMyName/whatsmyname
python3 -m venv WhatsMyNameEnvironment
source WhatsMyNameEnvironment/bin/activate
sudo pip install -r requirements.txt
deactivate
```

We can submit a query with the following commands.

```
cd ~/Downloads/Programs/WhatsMyName
python3 whats_my_name.py -u inteltechniques
```

The results with these commands are presented directly on your screen. However, in the automated script, I chose to export the results directly to a text file within your Documents folder by adding the additional text of `"> inteltechniques-WhatsMyName.txt"` to the end of the command. While Sherlock, SocialScan, Holehe, and WhatsMyName have been staples within previous editions, I prefer the reporting of **Blackbird** (github.com/p1ngul1n0/blackbird). We can install it with the following.

```
cd ~/Downloads/Programs
git clone https://github.com/p1ngul1n0/blackbird
cd blackbird
python3 -m venv blackbirdEnvironment
source blackbirdEnvironment/bin/activate
sudo pip install -r requirements.txt
deactivate
```

We can submit a query with the following command.

```
python3 blackbird.py -u inteltechniques
```

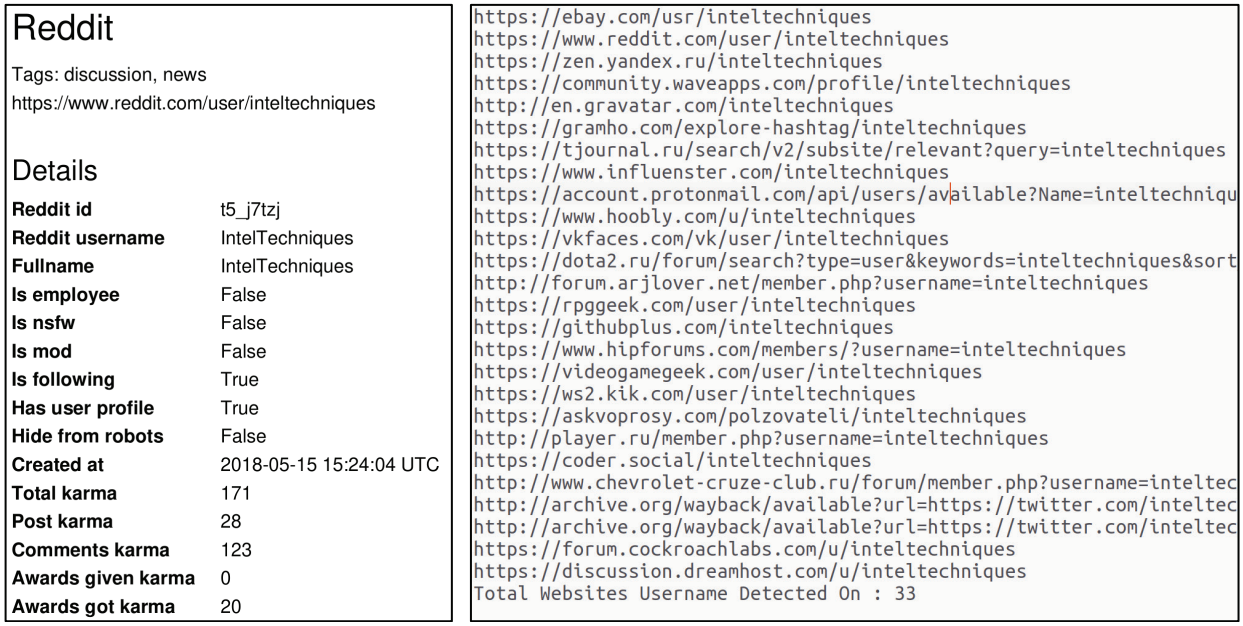
This presents the results on the screen, but also creates a file called inteltechniques.json in the "results" folder within the Blackbird package. The automated script, which is presented in a moment, automatically launches this file within Firefox upon completion.

Maigret (github.com/soxoj/maigret) was a last-minute addition to this book and the tools. It is a fork of Sherlock, which was previously explained, but currently scans over 2800 sites. This makes it the most thorough of all options. It also includes text, CSV, HTML, JSON, and PDF reporting, which is a great benefit to this utility over the others. Install with the following.

```
mkdir ~/Downloads/Programs/Maigret
cd ~/Downloads/Programs/Maigret
python3 -m venv maigretEnvironment
source maigretEnvironment/bin/activate
sudo pip install maigret
deactivate
```

We can submit a query with the following command, which forces all modules to be used, and outputs data to both a text file and PDF file within the "reports" folder for the program. The custom tool then opens this location. The text file only contains links to profiles associated with the target. The PDF file contains much more information, including URLs, follower counts, account creation dates, and additional details. Figure 4.07 (left) displays a partial result of a Reddit account associated with my name, as exported to a PDF. Compare that with the generic account details presented in the text file in Figure 4.07 (right).

```
maigret -a -P -T inteltechniques
```



Reddit

Tags: discussion, news
https://www.reddit.com/user/inteltechniques

Details

Reddit id	t5_j7tzj
Reddit username	IntelTechniques
Fullname	IntelTechniques
Is employee	False
Is nsfw	False
Is mod	False
Is following	True
Has user profile	True
Hide from robots	False
Created at	2018-05-15 15:24:04 UTC
Total karma	171
Post karma	28
Comments karma	123
Awards given karma	0
Awards got karma	20

```
https://ebay.com/usr/inteltechniques
https://www.reddit.com/user/inteltechniques
https://zen.yandex.ru/inteltechniques
https://community.waveapps.com/profile/inteltechniques
http://en.gravatar.com/inteltechniques
https://gramho.com/explore-hashtag/inteltechniques
https://tjournal.ru/search/v2/subsite/relevant?query=inteltechniques
https://www.influenster.com/inteltechniques
https://account.protonmail.com/api/users/available?Name=inteltechniqu
https://www.hoobly.com/u/inteltechniques
https://vkfaces.com/vk/user/inteltechniques
https://dota2.ru/forum/search?type=user&keywords=inteltechniques&sort
http://forum.arjlover.net/member.php?username=inteltechniques
https://rpggeek.com/user/inteltechniques
https://githubplus.com/inteltechniques
https://www.hipforums.com/members/?username=inteltechniques
https://videogamegeek.com/user/inteltechniques
https://ws2.kik.com/user/inteltechniques
https://askvoprosy.com/polzovateli/inteltechniques
http://player.ru/member.php?username=inteltechniques
https://coder.social/inteltechniques
http://www.chevrolet-cruze-club.ru/forum/member.php?username=inteltec
http://archive.org/wayback/available?url=https://twitter.com/inteltec
http://archive.org/wayback/available?url=https://twitter.com/inteltec
https://forum.cockroachlabs.com/u/inteltechniques
https://discussion.dreamhost.com/u/inteltechniques
Total Websites Username Detected On : 33
```

Figure 4.07: A Maigret PDF report (left) compared to the text output (right).

Next, we have **Email2Phone** (github.com/martinvigo/email2phonenumber.git). This tool queries an email address within various online services in an attempt to display any partial telephone numbers associated with the account. I rarely receive results here, but the occasional phone number identification justifies the efforts. The following installs the application.

```
cd ~/Downloads/Programs
git clone https://github.com/martinvigo/email2phonenumber.git
cd email2phonenumber
python3 -m venv email2phonenumberEnvironment
source email2phonenumberEnvironment/bin/activate
sudo pip install -r requirements.txt
deactivate
```

The following command searches test@gmail.com.

```
python3 email2phonenumber.py scrape -e test@gmail.com
```

Much like the Instagram tool, we see a lot of redundancy within these username options. This is never a bad thing. Sometimes, one or more of these lookup services fails or blocks my VPN. When that happens, I move on to another option. While I prefer Blackbird over the others, you may have a different favorite. I like Blackbird because of the JSON output and easy readability within Firefox. You may prefer a tool which compresses the data right into Terminal. Please be familiar with each and never expect them all to work well on any given day. Having multiple options allows us to raise or lower our confidence with the displayed results.

GHunt (github.com/mxrch/Ghunt)

This Terminal-based program requires some work to configure it properly, but the rewards justify the efforts. First, let's install everything with the following commands.

```
sudo pip install pipx
pipx ensurepath
pipx install ghunt
pipx ensurepath
```

Next, you must possess valid login cookies from an active Google account. This is because the Google API, which you will use to find information about various accounts, requires you to be logged in to an account in order to access these details. Therefore, you must acquire several pieces of data from your own Google account. I recommend using an account which you rarely access. As long as you do not log in to this account from a web browser AFTER you obtain these cookies, they should stay valid for long-term usage. Conduct the following.

- Navigate to gmail.com and log in to an active Google account.
- Right-click within this page and choose "Inspect".
- Click the "Network Tab" in the lower box and navigate to mail.google.com within this tab.
- Click the "Cookies" tab in right window.

You should now see several identifiers related to your Google account. Find each of the fields which are listed below, and document the data associated with your own account. I have provided sample data below.

SID: EthaBDv-SLCzi5-fGYFsgQEpsniuXp4vFdhSyxbfsgtJhePrKh7HWRhgfX42I4MoDa2Da
SSID: AhMQD6hufgMRsthCsVh
APISID: 90zgIYkLxDhgsLusoOl/AfJvEF8ihm_TfHOh86C1s
SAPISID: XsCNhC7fDVIE8hNY4Sr/AJfwurhpUkbFvsfUqBrP
HSID: Afmo3fgHjaUfpPhmfgRxfhZFQ
LSID: o.chat.google.com|o.mail.google.com|s.youtube:EQhaBIp83rHin-E_4hhykliVKWSZqPSfWl
__Secure-3PSID: EQjhaBDv-SCzi5-fGWYFsEpp4vFSf65yxbsgtJePrK7iOvBWuPnsdBiyw

Now that you possess the various account details required by GHunt, you can supply this data to the program. As long as these settings are not changed or refreshed by Google, you can reuse them for several queries. If you have followed my custom VM tutorials, the "Username/Email" tool is already configured to assist you with this process. Simply launch the tool and choose the option titled "GHunt Configuration" and select "[3] Enter manually all cookies". If you want to do this manually, you would enter "`ghunt login`" within Terminal. This will prompt you to provide your own SID, SSID, APISID, SAPISID, HSID, LSID, and __Secure-3PSID as previously obtained. Allow the application to complete the sign in process with these cookies.

In late 2022, GHunt released a major revision including a companion which attempts to populate your identifiers from a logged-in account. I originally had better success with the manual method above, but you may find the option of "[1] (Companion) Put GHunt on listening mode" easier by the time you read this. Google constantly changes the way they store your session, and it is common to find the "LSID" option to be missing when you need it. For most readers, the companion option will work better. You will need to install the GHunt browser extension if you choose this option. The following should help configure your GHunt application.

Open Firefox within your Linux VM; search "GHunt Companion"; install the extension; log in to your Google account; and press the "Synchronize to Ghunt" button in the extension. Execute the configuration option again and allow the application to acquire the session details. Once you have your Google session associated with GHunt, you can either launch the "Username/Email" tool from your custom VM or enter commands manually. The following command queries an email address from within the GHunt folder.

```
ghunt email larry@google.com
```

The result is as follows.

```
[+] 1 account found !
Name : Larry Page
[+] Custom profile picture !
=> https://lh3.googleusercontent.com/a-
/AOh14GiUjlWnt4MNgr7Wmeyb3PzXlka4E8PFEIlF27oIxIA
Profile picture saved !
Last profile edit : 2021/11/27 10:01:21 (UTC)
Email : larry@google.com
Gaia ID : 111627209495762463002
Hangouts Bot : No
[+] Activated Google services :
- Hangouts - Photos - Maps
[+] YouTube channel (confidence => 37.5%) :
- [Larry Page] https://youtube.com/channel/UCmpDzlgzPdbzShSzH48mCHg
- [Larry Page] https://youtube.com/channel/UCJuR7fG13KEpEPr8EH7bsgw
- [Larry Page] https://youtube.com/channel/UCNXk_sA4Kv3rDIYo8vI-gLQ
- [Larry Page] https://youtube.com/channel/UCefpKs_q0UsJV5_1n5_H13A
Google Maps :
https://www.google.com/maps/contrib/111627209495762463002/reviews
[-] No reviews
Google Calendar :
https://calendar.google.com/calendar/u/0/embed?src=larry@google.com
[-] No public Google Calendar.
```

Let's dissect our findings. We now know the name of the user and his profile photo has been saved within the "profile_pics" folder inside the GHunt installation directory. We are provided a date and time of his last profile update; his Google Accounts and ID Administration ID (GAIA); his four YouTube channels; and confirmation that he uses Hangouts, Photos, and Maps. We also know that he has not written any map reviews. This summary was provided almost instantaneously. If using the custom OSINT VM, the "Username/Email" tool with the "GHunt Email" option replicates this process. However, the custom script also outputs the data to a text file within your Documents folder and opens all results upon completion. The following Terminal command would search a Google Drive ID, and a fictitious result follows.

```
ghunt drive BxiMVs0XRA5nFMdKvBdBZjgmUUqptlbs74OgvE2upms
```

```
Drive ID : BxiMVs0XRA5nFMdKvBdBZjgmUUqptlbs74OgvE2upms
[+] Creation date : 2011/05/12 18:29:28 (UTC)
[+] Last edit date : 2011/05/12 18:29:28 (UTC)
Public permissions :
- reader
[+] Owner found !
Name : A Googler
Email : gdocsteam@gmail.com
Google ID : 02845897149113753960
[+] Custom profile picture !
=> https://lh3.googleusercontent.com/a-
/AOh14GhXaVAhS8Ci08Xito5iVJVsooEhsgUIGhZ45NjTfQ=s64
Profile picture saved !
```

We now know the document creation date/time; the last modified date/time; owner name; owner email; owner Google ID; and owner Google profile image. The automated version of this can also be found in your "Username/Email" tool within your OSINT VM.

Next, we can query a Google Accounts and ID Administration ID (GAIA) with the following command.

```
ghunt gaia 105144584335156066992
```

The result follows.

```
Name : Thrice
Gaia ID : 105144584335156066992
[+] YouTube channel (confidence => 50.0%) :
- [Thrice] https://youtube.com/channel/UC_gH-AQqoOUZ4ykZBCLdvww
Google Maps :
https://www.google.com/maps/contrib/105144584335156066992/reviews
```

Any time I encounter a Google email address, Google Drive account, or GAIA, I launch this tool. However, it can be quite finnicky. It commonly fails and requires me to enter fresh Google account identifiers. I often refer to this as an advanced tool, but wanted to include it here because of its presence within the custom tools.

The Users-Emails script for your OSINT VM is titled "users-emails.sh" within your downloads and is already configured to execute each of these options automatically. You should have it present within your "scripts" folder within "Documents". This script presents all options upon launch, as seen in Figure 4.08 (left). Selecting your desired choice presents a username or email address input box, as seen in Figure 4.08 (right). The script then executes the proper commands mentioned here and exports the results.

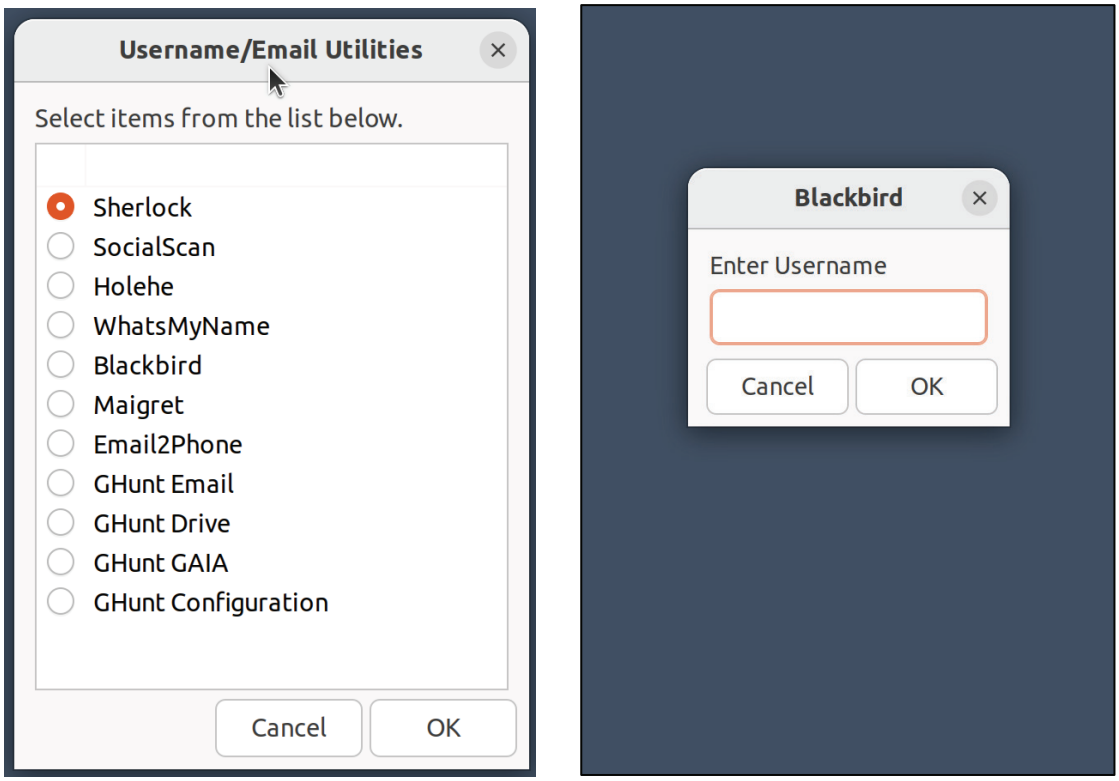


Figure 4.08: The Username Tool options.

EyeWitness

This Python script automates the collection of screen captures from websites. Imagine the following scenario. You are investigating a long list of website addresses that was provided to you as a lead. Maybe they were websites visited from a suspect's computer; a list of social network profiles discovered during your previous Twitter scrapes; or just a self-created list of URLs associated to your investigation. Manually visiting each site and using a screen capture tool can be overwhelming. Instead, let's automate the task. Install EyeWitness to your VM with the following commands within a new instance of Terminal.

```
cd ~/Downloads/Programs
git clone https://github.com/ChrisTruncer/EyeWitness.git
cd EyeWitness/Python/setup && sudo ./setup.sh
wget https://github.com/mozilla/geckodriver/releases/download/v0.32.0/
geckodriver-v0.32.0-linux-aarch64.tar.gz
tar -xvzf geckodriver* && chmod +x geckodriver
sudo mv geckodriver /usr/local/bin
```

In these steps, we entered the new folder you created during the installation of the Username Tool (Programs). This will be where we store applications which do not traditionally install into your operating system. You can now execute EyeWitness, but you must first navigate to the folder where the Python script is located. We will correct this with a custom script in a moment, but let's first test the application within Terminal.

- Open your Applications menu and launch Text Editor.
- Type or paste URLs, one per line, and save the file to your Desktop as "sites.txt".
- Open Terminal and enter the following commands.
- `cd ~/Downloads/Programs/EyeWitness/Python`
- `./EyeWitness.py -f ~/Desktop/sites.txt --web -d ~/Documents/EyeWitness/`

When finished, you should have a new file within your Documents/EyeWitness folder titled Report.html. Double-clicking this file opens it within Firefox. Figure 4.09 displays the first two results from my "sites.txt" file which contained exactly the following data.

```
https://inteltechniques.com
https://computercrimeinfo.com
https://instagram.com/mikeb
```

The results include screen captures of each target website and detailed information including the server IP address, page title, modification date, and full source code of the page. This is very beneficial in two ways. First, it automates the task of researching multiple target websites. Second, it saves each page as it existed at a specific moment in time. If your suspect deletes or modifies a website after you have captured it, you possess evidence of a previous state. The annoyances of this method include the requirement to create a text file and need to articulate the location of the data within Terminal. We can make this easier with a custom script.

The script on the following page will prompt you to choose between a single URL or multiple URLs. If you choose single, it will prompt you for the URL and then it will execute EyeWitness. The report will be saved into a new folder titled with a current timestamp within the Documents/EyeWitness folder. At the end of the process, you will be asked if you would like to open the report. If you choose the option for multiple URLs, you will be prompted to choose the text file containing all of the target addresses. EyeWitness will then conduct the queries and generate the report as described in the single URL option. The script and shortcut file are included within the "vm-files" archive previously mentioned, titled captures.sh and captures.desktop.

This utility can be very beneficial when you have dozens, hundreds, or even thousands of domains of interest. I once converted a list of over 300 suspect Twitter accounts into a folder of screen captures of each. This allowed me to quickly identify which accounts were truly valuable to my investigation by simply viewing the evidence

similar to photos in an album. The report was several hundred pages but was generated in only a few minutes. Once you complete your custom VM within the next chapter, you can launch EyeWitness at any time with the "Captures Tool" icon in the Dock.

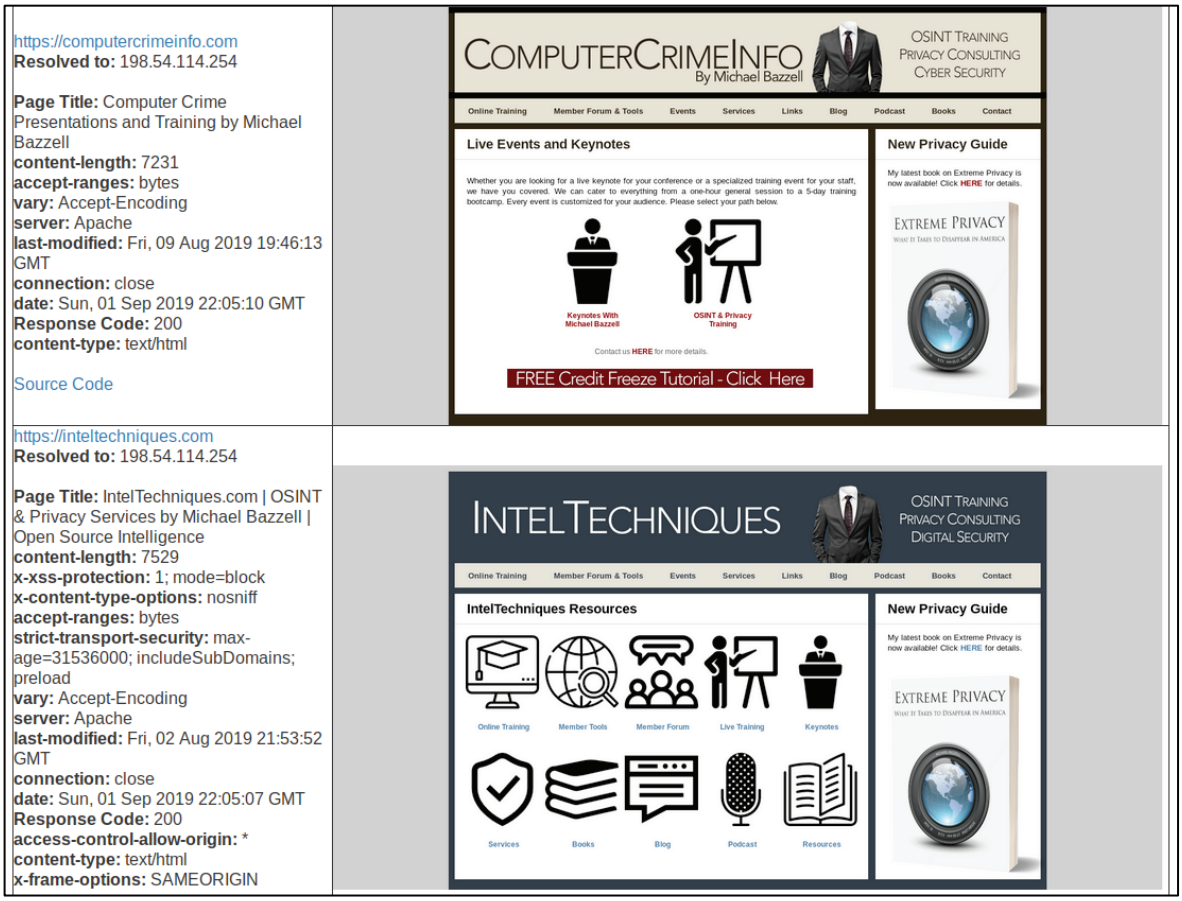


Figure 4.09: A partial EyeWitness report.

Domain Tool

Later chapters will explain manual techniques for analysis of a domain. We can automate much of that work with various scripts and programs. First, let's gather all of the required software with the following commands within a new instance of Terminal.

```
sudo snap install amass
cd ~/Downloads/Programs
git clone https://github.com/aboul3la/Sublist3r.git
cd Sublist3r
python3 -m venv Sublist3rEnvironment
source Sublist3rEnvironment/bin/activate
sudo pip install -r requirements.txt
deactivate
cd ~/Downloads/Programs
git clone https://github.com/s0md3v/Photon.git
cd Photon
python3 -m venv PhotonEnvironment
source PhotonEnvironment/bin/activate
sudo pip install -r requirements.txt
deactivate
```

```
cd ~/Downloads/Programs
git clone https://github.com/laramies/theHarvester.git
cd theHarvester
python3 -m venv theHarvesterEnvironment
source theHarvesterEnvironment/bin/activate
sudo pip install -r requirements.txt
deactivate
sudo pip install testresources
sudo pip install webscreenshot
cd ~/Downloads/Programs
git clone https://github.com/Lazza/Carbon14
cd Carbon14
python3 -m venv Carbon14Environment
source Carbon14Environment/local/bin/activate
sudo pip install -r requirements.txt
deactivate
```

You should now have new folders within a folder titled "Programs" inside your "Downloads" folder. These folders contain the Python scripts necessary to launch the programs. Next, let's take a look at the manual commands required to execute each program. I will use my own website during the demonstrations. You would need to execute each script from within the proper folder containing the Python files. However, our custom script allows us to use a menu without the need for Terminal commands.

```
amass intel -whois -ip -src -active -d inteltechniques.com
amass enum -src -ip -passive -d inteltechniques.com
python3 sublist3r.py -d inteltechniques.com
python3 photon.py -u inteltechniques.com -l 3 -t 100
python3 theHarvester.py -d inteltechniques.com -b bing
python3 carbon14.py https://inteltechniques.com
```

The automated script which automates all of these tasks and commands is included in your digital downloads, titled "domains.sh" within the "scripts" folder. It should be present within your Linux VM as long as you downloaded the files as previously explained. We will also run through all of the commands again in the next chapter. Next, let's add a shortcut icon to our Dock in order to have easy access to this new utility.

- Open the Applications menu again and scroll down to "Domain Tool".
- Right-click this program and select "Add to favorites".

In a later chapter, I explain investigative techniques for domain names, and the importance of searching for subdomains. A subdomain is a dedicated address that is part of a larger domain, such as pics.inteltechniques.com. While inteltechniques.com might be a main landing page, pics.inteltechniques.com could possess hidden content not available within the main website. Programs within this script, such as **Amass** and **Sublist3r**, request a domain name and then fetch subdomains associated with the target. This is a vital step for any investigation into a domain name. The following explains each utility in this new menu, including the results for a test query of cnn.com.

Amass: This option takes the longest to run, but it is the most thorough. It uses a brute force option which attempts to determine any possible subdomains. It creates two reports, both of which will be located in the Documents/Amass folder. In my test with cnn.com, it found 282 subdomains, such as the following.

footballclub.cnn.com
edition.cnn.com
fave.edition.cnn.com

Sublist3r: This program scans much faster but will only find common subdomains. This may be sufficient for most tasks. It creates a report located in the Documents/Sublist3r folder. In my test, it found 808 subdomains of cnn.com such as the following.

trends.cnn.com
tours.cnn.com
coupons.cnn.com

Photon: This option does not attempt to find subdomains. Instead, it searches for internal pages within a target website. It creates a report located in the Documents/Photon folder. Examples include the following.

cnn.com/webview
cnn.com/NOKIA
cnn.com/Quickcast

TheHarvester: This program searches a supplied domain with the intent of providing email addresses associated to the target. It creates a report located in the Documents/theHarvester folder. During my search of cnn.com, it located 72 hosts and 16 email addresses. Further investigation into these results may reveal new sources of information. In each of these scenarios, you are greeted with text files containing numerous URLs. If desired, you could submit each to EyeWitness. This would create a new report with screen captures from any valid addresses. This allows for a quick review of the evidence to see if anything warrants further investigation.

Carbon14: This application helped an investigation as I was writing this. I was investigating an anonymous blog, with the intent of learning more about the age of the site. It was a brand-new WordPress installation, and every post had the same recent 2019 date and time. It had obviously been created recently, but then received an import from an older blog. I wanted to know more about the true dates of the original posts. This is the perfect scenario for Carbon14. It searches for any images hosted within the page and analyzes the metadata for creation dates. A query of my own site indicates that the static page was last modified on August 2, 2019, and the images were posted in March 2018, May 2019, and July 2019. If this were my target, I would now have suspicion that the original blog posts were from an earlier date. These dates can be intentionally or unintentionally altered, so this is not a forensically sound method. It is simply an additional piece to the puzzle which may warrant further investigation. I find that Carbon14 works best on static websites and blogs. I have also had surprising success when targeting social network profiles.

We have now made it through the difficult section of this chapter. The remaining programs have standard launch options and do not require custom scripts to make them user friendly. These are all similar to their equivalents within other traditional operating systems. We will replicate all of this within Mac and Windows in Chapter Six.

Metadata-MediaInfo

This is a utility for displaying hidden metadata within a media file. The most common example is the metadata within a video obtained directly from the original source. First, open Terminal and type the following to install the application within the Applications menu of Ubuntu.

```
sudo apt install -y mediainfo-gui
```

You can now click on the Metadata Tool icon in the Applications menu to launch the program. Click on the "File" option in the menu and then open either a file or folder. The default view offers little data, so click on "View" and then "Text" from the menu. This presents all metadata available within the file. In my test, I received

the (partial) output visible in Figure 4.10 from a video sent to me via email directly from a mobile device. It identified the make, model, and operating system version of the device, along with GPS information about the location during capture. Note that this type of data is usually not available from videos downloaded from the internet. The ideal scenario is that you possess a video file sent directly from a device.

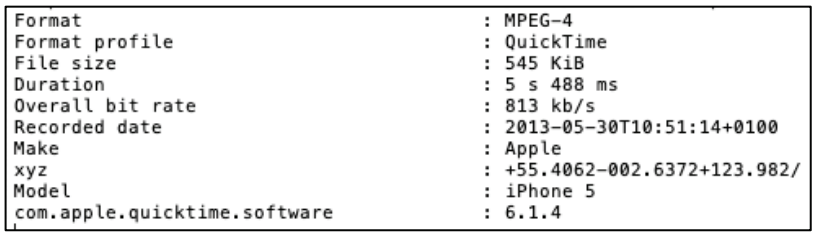


Figure 4.10: A MediaInfo result.

Metadata-Exiftool

Image metadata, also called Exif data, will be explained later in more detail. It can include the make, model, and serial number of a camera or location data of the capture. Third-party applications to view this data are no longer necessary because Ubuntu has this feature embedded in the default image viewer. Simply open any image, click the three horizonal lines in the upper right menu, and select Properties. The Metadata and Details tabs display all available details such as camera information, location, and dates. However, this does not help much in regard to bulk images. We can easily install a Terminal-based utility which allows for automated export called **ExifTool**. The following command in Terminal will complete the process.

```
sudo apt install -y libimage-exiftool-perl
```

You can now export the metadata from multiple images into a spreadsheet. Consider the following example where I possess a folder of images on my Desktop titled "Evidence". I want a spreadsheet of all available Exif data in reference to these images. The following command in Terminal will create a report on my Desktop titled Report.csv.

```
cd ~/Desktop/Evidence && exiftool * -csv > ~/Desktop/Report.csv
```

This command launches ExifTool (exiftool); reads all files in the current folder (*); specifies creation of a CSV file (-csv); and saves the file output to the Desktop titled as Report.csv (> ~/Desktop/Report.csv). I do not use this feature often, but it has saved me a lot of work when needed. I once possessed a folder of 200 images retrieved from a directory on my target's blog. This command created a spreadsheet with all metadata from the images, which identified GPS locations valuable to the investigation. This option is also present within the custom Metadata Tool within your OSINT VM.

Metadata-mat2

While you may want to VIEW metadata within images with the previous command, it is just as likely you will want to REMOVE data from your own media. Maybe you want to upload an image as part of a covert investigation. That image could have metadata such as dates, times, locations, and camera details. This is where MAT2 (Metadata Anonymisation Toolkit 2) can assist. First, we must install it with the following.

```
sudo apt install -y mat2
```

Next, we can issue a command to clean a file. Assume I have an image on my Desktop titled dirty.jpg. The following commands within Terminal would change to the Desktop directory and create a new file called "dirty.cleaned.jpg".

```
cd ~/Desktop
mat2 dirty.jpg
```

If you launch the automated Metadata Tool script included within your OSINT VM, you will be prompted with a selection dialogue box similar to the option presented with the video tools. Choosing "Clean Metadata" presents a file selection tool which launches this mat2 process and generates the clean image within the same folder as the original. This tool will never overwrite your original file.

Metadata-xeuledoc

This application extracts metadata hidden within Google Documents, including the documents owner's name, email address, and Google identifiers. This is a great way to determine who is behind an online Google document which is part of your investigation. The following commands install the program.

```
mkdir ~/Downloads/Programs/xeuledoc
cd ~/Downloads/Programs/xeuledoc
python3 -m venv xeuledocEnvironment
source xeuledocEnvironment/bin/activate
sudo pip install -U xeuledoc
deactivate
```

We can now execute the application manually toward a Google document. Consider the spreadsheet located at https://docs.google.com/spreadsheets/d/1KXksB1vj7fXPNS4OYL0idQne3HXVnamtUP1h0ut3xwk. This page provides a link to a third-party piracy website claiming to offer one of my books for download. There is nothing within this spreadsheet which identifies the creator of this document. However, xeuledoc can assist. The following command queries the tool with the document link.

```
xeuledoc https://docs.google.com/spreadsheets/d/1KXksB1vj7fXPNS4OYL0idQne3HXVnamtUP1h0ut3xwk
```

The result appears below.

```
Document ID : 1KXksB1vj7fXPNS4OYL0idQne3HXVnamtUP1h0ut3xwk
[+] Creation date : 2020/01/08 16:35:56 (UTC)
[+] Last edit date : 2020/01/08 16:35:56 (UTC)
Public permissions :- reader
[+] Owner found !
Name : donlad grifith
Email : donladgrifith@gmail.com
Google ID : 09765685956674862946
```

We now have a name and email address to research. The automated script for this tool is part of "metadata.sh" and is included in your downloads.

Metadata-Sherloq

Sherloq is a standalone application which can help identify modified areas of photographs. The advanced capabilities to alter images within Photoshop can easily fool our eyes. Computers are more difficult to convince. Let's install the application first, then discuss the usage. Conduct the following within Terminal.

```
cd ~/Downloads/Programs
sudo apt install subversion -y
git clone https://github.com/GuidoBartoli/sherloq.git
cd sherloq/gui
python3 -m venv sherloqEnvironment
source sherloqEnvironment/bin/activate
```

```
sudo pip install -r requirements.txt
deactivate
```

You can now launch this application manually with the following command within Terminal.

```
python3 ~/Downloads/Programs/sherloq/gui/sherloq.py
```

Sherloq replicates many of the image metadata and manipulation detection methods which are explained later in the book. The benefit here is the ability to conduct an examination offline. If you have a sensitive photo, you may not want to upload it into one of the online metadata websites which is explained later. Instead, you may want to keep it restricted to your local machine. Click the "Load Image" button and select a photo. There are numerous options present within the left menu, most of which will be explained within the Images chapter (21).

Consider the image visible in Figure 4.11. The upper-right image displays the "Principal Component Analysis" which highlights the lips of the subjects for potential digital manipulation. If you investigate online images often, I encourage you to understand all of the options available within this software application. The author's website at github.com/GuidoBartoli/sherloq and the tutorials provided in Chapter 21 should assist.

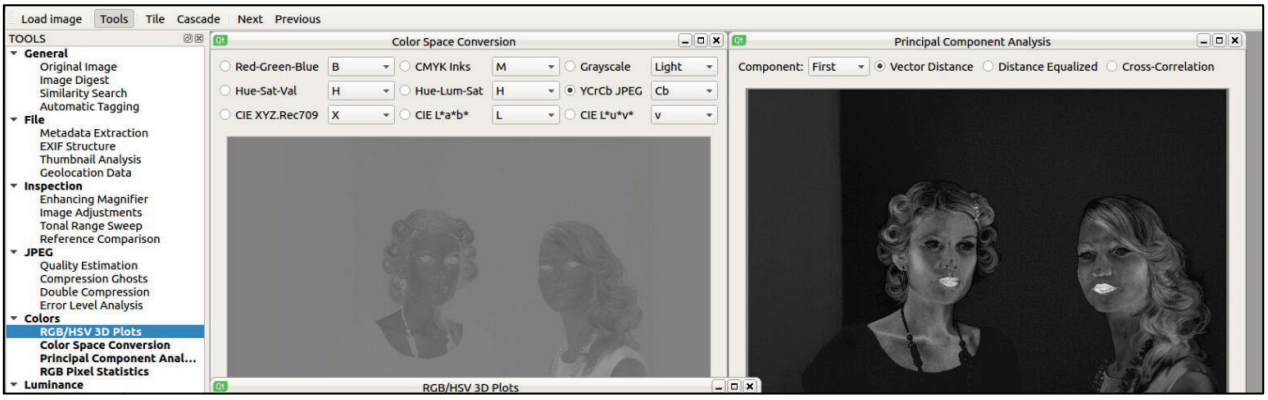


Figure 4.11: A Sherloq image analysis.

The "metadata.sh" script within your downloads prompts you for usage of all of these applications. The dialogue (with included shortcut) replicates each task presented here. Simply launch the "Metadata" icon within your Dock after you have created your final OSINT VM in the next chapter.

HTTrack (httrack.com)

There are several ways to make an exact copy of a static website and this software will walk you through the process. You may want to do this when you locate a target website and are concerned the site could be taken down. Any time that you find any content that will be used in court, you should archive the entire site. This application automates the process. The result is a locally stored copy in which you can navigate as if it were live. This is beneficial in court when internet access is not appropriate, or a website has been taken offline. The following command within a new Terminal session will download the software and configure it within your operating system.

```
sudo apt install -y httrack webhttrack
```

You can now type `webhttrack` at any Terminal prompt and the graphical version of the software will execute within your default browser. Typing `httrack` will load the Terminal version. You should also have the Archives Tool (archives.desktop) available in your Applications menu. When the application loads, you can select either option. The Terminal version presents text prompts to enter your target data. The browser version does the same thing, but it is point-and-click instead of text entry. Note that the browser version will likely require you

to "try again" upon launch and select the option to load a non-secure website. This is not a security risk, but it is an annoyance.

Clicking the "next" button in the browser version will bring you to the project screen. Give your project a title, preferably the website name, and choose a location to save all of the data. Click next and then "add URL". Enter the exact website that you want to archive. Do not enter any login or password. Click "next", then "finished", and you are done. The application will begin extracting all public information from the site. This can take a while, depending on the size of the site. When complete, you will have an exact copy in which you can navigate and search offline. Archiving to an external drive creates a replica that can be held for future analysis. This program only works on smaller static websites, and cannot archive large dynamic sites such as social networks.

HTTrack is a very old program, and was never intended to function within today's latest technology. However, it still works surprisingly well on small business websites. It can also extract a surprising amount of content from large sites such as Wikipedia. I once used this service to extract several gigabytes of scraped LinkedIn data from a questionable website before it disappeared, which included profiles, names, email addresses, user IDs and employment history. This tool should always be ready in your arsenal.

Some tools remaining in this chapter require a small package for sound, which is installed with the following.

```
sudo apt install -y libcanberra-gtk-module
```

Metagoofil

Metagoofil is a Linux option designed to locate and download documents. It does not always work perfectly, as you are at the mercy of search engines to provide the data. When Google decides to block Metagoofil, its usefulness ceases until the software receives an update. Install the software with the following steps.

```
cd ~/Downloads/Programs
git clone https://github.com/opsdisk/metagoofil.git
cd metagoofil
python3 -m venv metagoofilEnvironment
source metagoofilEnvironment/local/bin/activate
sudo pip install -r requirements.txt
deactivate
```

The following Terminal commands in your Linux VM switches to the proper directory (cd); launches Python 3 (python3); loads the script (metagoofil.py); sets the target domain (-d cisco.com); sets the file type as pdf (-t pdf); and saves the output to a new folder called "cisco" on the Desktop (`-o ~/Desktop/cisco/`). The last command asks for docx and xlsx files.

```
cd ~/Downloads/Programs/metagoofil
python3 metagoofil.py -d cisco.com -t pdf -o ~/Desktop/cisco/
python3 metagoofil.py -d cisco.com -t docx,xlsx -o ~/Desktop/cisco/
```

This automatically downloads any found documents. We could now analyze these documents with ExifTool as mentioned previously with the following commands within Terminal. The following commands navigate you to the directory with the files and then creates a report.

```
cd ~/Desktop/cisco
exiftool * -csv > ~/Desktop/cisco/Report.csv
```

As you may suspect, I prefer to download and analyze document metadata within Linux. While the command to launch ExifTool is easy, the commands for Metagoofil can become complicated, especially when we start querying for multiple types of documents. Therefore, we will use the automated script included in the Linux downloads, which are already installed. It executes Metagoofil and tries to download any documents from an

entered domain. The second option will repeat that process and then create a report with all the metadata. This script is titled "documents.sh". In my experience, this utility works fairly well when NOT using a VPN, but Google may block your connection if the VPN IP address appears abusive. While writing this chapter, I could execute a full query one time. All additional queries failed due to Google's blockage. Personally, my protocol in regard to document metadata collection and analysis is as follows.

- If I already possess numerous documents on my computer, I create a metadata CSV spreadsheet using ExifTool. I then analyze this document.
- If my target website possesses few documents, I download them manually through Google or Bing within a web browser.
- If my target website possesses hundreds of documents, I use Metagoofil, but only download one file type at a time. If my target were cisco.com, I would execute the following commands in Terminal.

```
python3 metagoofil.py -d cisco.com -t pdf -o ~/Desktop/cisco/
python3 metagoofil.py -d cisco.com -t doc -o ~/Desktop/cisco/
python3 metagoofil.py -d cisco.com -t xls -o ~/Desktop/cisco/
python3 metagoofil.py -d cisco.com -t ppt -o ~/Desktop/cisco/
python3 metagoofil.py -d cisco.com -t docx -o ~/Desktop/cisco/
python3 metagoofil.py -d cisco.com -t xlsx -o ~/Desktop/cisco/
python3 metagoofil.py -d cisco.com -t pptx -o ~/Desktop/cisco/
```

Reddit Tools

I have found Reddit to become a common source for online evidence within my investigations. I will later discuss manual and semi-automated ways of locating and documenting data available on Reddit. However, automated tools take our efforts to another level and produce impressive results. Let's start with installing the required three Reddit search applications (Bulk Downloader For Reddit, Reddits Finder, and Downloader For Reddit) with the following steps.

```
mkdir ~/Downloads/Programs/bdfr
cd ~/Downloads/Programs/bdfr
python3 -m venv bdfrEnvironment
source bdfrEnvironment/bin/activate
sudo pip install -U bdfr
deactivate
mkdir ~/Downloads/Programs/redditsfinder
cd ~/Downloads/Programs/redditsfinder
python3 -m venv redditsfinderEnvironment
source redditsfinderEnvironment/bin/activate
sudo pip install -U redditsfinder
deactivate
cd ~/Downloads/Programs
git clone https://github.com/MalloyDelacroix/DownloaderForReddit.git
cd DownloaderForReddit
python3 -m venv DownloaderForRedditEnvironment
source DownloaderForRedditEnvironment/bin/activate
sudo pip install -r requirements.txt
deactivate
```

Bulk Downloader For Reddit

The following two commands download data from Reddit. The first retrieves up to 1000 posts within the Subreddit "osint", and saves them to a text file within the Documents folder. The second retrieves all posts by the user "inteltechniques" within the Subreddit "osint" and saves the data.

```
python3 -m bdfr download ~/Documents/Reddit/osint/ --subreddit osint
python3 -m bdfr download ~/Documents/Reddit/inteltechniques/ --user
inteltechniques --subreddit osint
```

Reddits Finder

From any Terminal prompt, I can now issue a command targeting a Reddit user, such as the following.

```
redditsfinder inteltechniques
```

This queries the Pushshift API, which is explained in a later chapter. It presents current and deleted post metadata attached to the target username within the Terminal screen. This is good, but we can do better. The following command requests every post I have made, regardless of deletion, along with any images I uploaded with a documentation text file, and saves the data to a folder titled "inteltechniques" within a folder titled "users" inside the Documents directory of my VM.

```
cd ~/Documents
redditsfinder inteltechniques -pics -d
```

The following is an embarrassing attempt to promote my podcast which I deleted a few hours after posting.

```
"datetime": "Wed Jun 20 2018, 02:32 PM UTC", "created_utc": 1529505149,
"subreddit": "privacy", "score": 7, "link_id": "t3_8scj6u",
"body": New release every Friday: https://inteltechniques.com/podcast.html
```

Downloader For Reddit

Executing this graphical program is easy with the following commands.

```
cd ~/Downloads/Programs/DownloaderForReddit/
python3 main.py
```

Click the "Add User" button in the lower left to add target Reddit user accounts. Click the "Add Subreddit" button to add target Subreddits. Double-click each and change the "Post Limit" to "Max". You can now right-click any entry to download the data. Click "Database" and "Database View" to see your results. You can export any data from this view. This is a very robust program which should be explored before using it for an investigation.

The included script, titled "reddit.sh", automates the processes for all of these applications. Launch "Reddit Tool" from your Dock or Applications menu. It provides the following options.

```
Subreddit Download
User Download
User Archive
Launch Downloader
```

After each script has finished, the folder containing the data is presented on your screen. Hopefully, this simplifies the usage of these great tools.

Google Earth Pro (google.com/earth)

Google Maps is an online website that is discussed later. Google Earth Pro is a standalone application that takes the Google Maps data to another level. With this application, we have access to many mapping tools. These tools can import data from spreadsheets and help you visualize the content. In order to maintain the scope of open source intelligence, I will focus on only a few specific tools. First, we need to enter the following commands within Terminal to install the software.

```
wget http://dl.google.com/dl/earth/client/current/google-earth-
stable_current_amd64.deb
sudo apt install -y ./google-earth-stable_current_amd64.deb
sudo rm google-earth-stable_current_amd64.deb
```

Within the application, the first step is to display your location of interest. This can be accomplished by typing the address or GPS coordinates in the upper left search field. When you see your target location and have set the zoom to an appropriate level, you are ready to start adding layers. By default, you will only see the satellite imagery of the location. The menu on the left possesses options for adding new content to this view. The last box in this menu is titled "Layers". Inside this menu are several data sets that can be enabled and disabled by the checkbox next to each. I recommend disabling all layers and then enabling one at a time to analyze the data that is added to your map view. The following details will explain the layers of interest.

- Photos - Digital images uploaded through social networking sites
- Roads - Text layer of road names
- 3D Building - Alternative 3D view of some locations
- Gallery - User submitted content including YouTube videos

Another Google Earth feature available that is often overlooked is the Historical Imagery option. This can be activated by selecting the "clock" icon in the upper menu bar of the application. This will open a slider menu directly below the icon. This slider can be moved and the result will be various satellite images of the target location taken at different times. Figure 4.12 displays a target area with the Historical Imagery option enabled. The view has been changed to the satellite image obtained on 05/30/2008.

Usually, the quality of the images will decline as you navigate further back in time. This can be useful in identifying changes in the target location such as building modifications, additional vehicles, and land changes. Drug enforcement agents often use this tool to monitor suspected drug growth at a target location. While the Google Earth program is not updated often, the imagery content within is pulled directly from the Google archives. It is vital within our arsenal of tools and I find it more pleasant to navigate than the official Google Maps website. This resource has assisted many of my investigations over the past decade by providing historic satellite imagery which was unavailable from any other online source. Note that Google Earth Pro will not currently install within Apple ARM-based virtual machines or hosts. Hopefully, Google will create an ARM version in the near future.

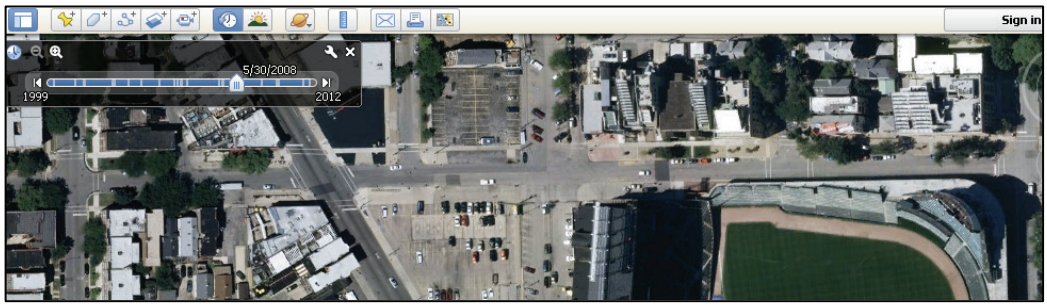


Figure 4.12: A Google Earth view of historic imagery from 2008.

Screen Capture

I previously mentioned multiple unique methods of capturing website evidence within your browser. However, you may need a more robust solution for capturing your entire screen. This can be approached from two specific avenues. First, you could record either a still or video capture of your entire computer from your host. In other words, you could use the default capturing software within your Windows or Mac machine to record video or save a screenshot of your Linux VM. I do not recommend this. Recording from your host displays evidence of your other operating system. While it would likely never be an issue, it exposes you unnecessarily. Consider the following.

You are a detective investigating a homicide. You find evidence online implicating your suspect. You are on a Windows computer, but all of your investigation was conducted within a Linux VM. You launch recording software on the Windows computer and record a video of your work. This video was submitted to the defense. For a brief moment, the video captured a file on your Windows desktop titled Accounts.xlsx. It is a spreadsheet containing all of your covert online accounts, has no connection to the investigation, and was not intended to be exposed. The defense makes a motion to analyze this file, as they believe it could be associated with the investigation. The judge approves, and you must share all of your covert accounts with the other side. Does this sound far-fetched? It happened to a colleague of mine in 2014.

I have become more paranoid of digital mistakes than necessary, but I believe we should never take chances. Therefore, I recommend that all of your screen captures be executed within your VM. Fortunately, Linux has many options. First, let's consider the default Ubuntu screenshot utilities. The following keyboard keys and key combinations create high-resolution exact screenshots of your VM.

- PrtSc: Save a screenshot of the entire screen to the "Pictures" directory.
- Shift + PrtSc: Select and save a screenshot of a specific region to the "Pictures" directory.
- Alt + PrtSc: Save a screenshot of the active window to the "Pictures" directory.
- Ctrl + PrtSc: Copy the screenshot of the entire screen to the clipboard.
- Shift + Ctrl + PrtSc: Select and copy the screenshot of a specific region to the clipboard.
- Ctrl + Alt + PrtSc: Copy the screenshot of the current window to the clipboard.

If you are using an external Mac keyboard, the F13 key replicates "PrtSc". Some Windows keyboards may have a key labeled Print Scrn, Prnt Scrn, Prt Scrn, Prt Scn, Prt Scr, Prt Sc or Pr Sc. If you simply want a screenshot of your Desktop, such as evidence of your output within a Linux application, this is your best option. If you need video of your activities, we will need to install a third-party application.

Kazam is a minimal tool for screen recording. It also includes screenshot support, but I find the native Ubuntu option easier and faster. Kazam is most suitable for getting the task done quickly without providing many options. The following steps will install and execute a Kazam capture.

- In Terminal, enter `sudo apt install -y kazam` and strike enter.
- Launch Kazam from the Applications menu (make a shortcut if desired).
- Click "Capture" within the application.
- After the countdown, your entire screen is being captured.
- When finished, click the icon in the upper right and choose "Finish Recording".
- Choose "Save for later"; click "Continue"; choose a location; and click "Save" in the upper right.

You now possess a high-resolution video of your entire session. This can be very beneficial in many scenarios. I have used this to document a specific portion of my investigation when a simple still capture would not suffice. It could also be used to explain a search technique through video, which can be easily replicated by the person receiving a copy. I appreciate the simplicity of this application, and the lack of any branding or logos within the videos.

BleachBit

I explained BleachBit in Chapter One. Type the following into Terminal to install the application to your VM.

```
sudo apt install bleachbit
```

Clicking the nine dots in the lower left will present two BleachBit applications. The second icon executes the software with administrative privileges and is the option I choose. Upon first launch, click "Close" to accept default configuration. Select every option except the "Free disk space" feature. Click "Preview" to see a report of recommended cleaning. Click "Clean" to execute the process. I run this program at least once a week to remove unwanted files.

Updates

Once you have your new VM created, be sure to apply all updates. If you downloaded the custom VM files mentioned in this chapter (and used within the next), please execute the "Updates" script in the Dock. This makes sure that you have the latest programs and dependencies, but also corrects many version conflicts. **After all updates are applied, you must reboot the VM**.

Program Launching

All of these programs and scripts can be launched from within the Applications menu (nine dots), but I prefer to place them within my Dock. I provide more vital customizations in the next chapter, including a single command which automatically populates your Dock with every OSINT application explained throughout this entire book.

Scripts and Shortcuts

Throughout this chapter, I have explained ways to copy my digital files into your own VM. If you have done that, you are likely in good shape. If you are missing any, you will experience problems. If any scripts are not present or seem damaged, my advice is to download a fresh copy by repeating the steps previously presented. You will then have all of the custom files within the appropriate folders. We will quickly replicate our entire VM in the next chapter. As a reminder, all of the commands required to install and configure all applications from this chapter are online at **https://inteltechniques.com/osintbook10/linux.txt**. This allows for easier copy and paste directly into your own VM. This also presents a way for me to provide updates as needed. I will announce any changes to the Linux steps at **https://inteltechniques.com/osintbook10**. If you find an issue which needs updated, please contact us at **errors@inteltechniques.com**.

Reminder: You will see some Pip errors or warnings displayed while you install these apps. These are often version conflicts, and typically do not impact the functions of the program. While we will take steps to eliminate as many errors as possible, yellow warning and red error text within Terminal is common.

New Programs

We have only scratched the surface of the possibilities within a Linux virtual machine. New OSINT programs and scripts emerge weekly. Hopefully, this chapter has given you the foundation required to understand the installation and configuration of any future applications. The methods here can be replicated to create your own custom Linux build which works best for your investigative needs.

10th Edition Script Modifications

Every script has been modified since the 9th edition. First, I minimized some of the script overhead and attempted to simplify many of the tasks. Next, I tried to include more timestamps within exported files. I wanted

the date and time visible within the file name for easy identification. I also wanted to make sure that additional executions of commands within the tools did not overwrite previously acquired data. This was not appropriate everywhere, but readers of previous editions should see the subtle changes.

License

The custom tools, scripts, tutorials, and files downloaded from my website are released to you for free. You may modify and use the content any way you desire, including unlimited personal and government use, as long as you respect the following restrictions.

- **NonCommercial**: You may not use the material for commercial purposes, including training programs.
- **NonDistribution**: You may not distribute the material, including digital, internet, and physical distribution.

Miscellaneous Repairs

We have installed a lot of software and should clean up a bit. The following attempts to fix any issues we cause within all of our installations.

```
sudo apt update --fix-missing
sudo apt -y upgrade
sudo apt --fix-broken install
```

Troubleshooting

I do not expect everything presented here to work flawlessly for every reader. I wish it were that simple. You may encounter issues. My best advice is to restart the process with which you are having the trouble, and follow all instructions exactly as written. Even then, you may experience frustration. The following are some tips which have helped members of my online video training.

- When typing in Terminal, you can hit the tab key and it will suggest all the possible options that start with the string you have typed so far. It will also autocomplete your command if only one option exists. For example, if you are trying to navigate to your Downloads/Programs folder, you can just type "cd ~/Dow" [tab], then "Pro" [tab] to complete the command. Typing cd ~/D [tab][tab] would list all folders starting with D.
- Use the "up" arrow key to navigate through previous commands.
- "Ctrl" + "C" kills any running process, and typing "ls" displays the directory contents.
- You can copy and paste to/from Terminal, but only with right-click (not "Ctrl" + "V").
- Keyboard arrows will move you through a Terminal command, but mouse clicks do not.
- You can modify the size of your Dock icons in "Settings" > "Dock".

We are not done adding new Linux applications, but I believe we have covered all the basics. Later in the book, I present an advanced Linux applications chapter which will dive deeper into automated OSINT options.

Executable Files

If you downloaded my scripts and copied them to your VM, they should launch without any further steps. However, if you typed your own, they likely do not possess the authority to execute. If you have ".sh" scripts in your "scripts" folder, the following commands in Terminal will apply the appropriate execution rights to each.

```
cd ~/Documents/scripts
chmod +x *.sh
```

Hidden Files

By default, Ubuntu hides system files which should not normally be manipulated. You may need to see these within the Files application. The following steps allow you to see all files on the system, and the change is persistent between reboots.

- Open the "Files" program in either your Dock or Applications menu.
- Click on the three horizontal lines in the upper right area.
- Select "Show Hidden Files".
- Close the "Files" program.

Large File Transfer

We have configured our VM to possess a shared folder on the Desktop. This is an avenue to transfer files from within the VM to the host and vice versa. This can be used to copy evidence from within the VM to your host for preservation to external media. I have experienced limitations with this method. If I have hundreds of large videos, I often receive errors when I try to copy all of them to my host through the shared folder. This is much more prominent in VirtualBox versus the premium option VMWare. The best solution I have found is to transfer evidence directly to a USB drive within the VM. If desired, you could choose this method and eliminate any shared folder on the VM.

When you insert a USB drive into your host computer, it is immediately read by your primary operating system. Your running VM does not see the device. Within the VirtualBox menu, you can choose "Devices", then "USB", and try to click on the USB drive, if found. The issue you may face is that your host has the drive locked and will not allow the VM to take over. You can usually resolve this by ejecting the USB drive within your host. On both Windows and Mac, you should see an eject icon next to the drive or by right-clicking the device. After ejecting, attempt to load the USB device again through the VirtualBox menu. This may present another issue. If your USB device is formatted specifically for Mac or Windows, it may not be readable by Linux. I suggest specifying a USB device solely for use within this VM and formatting it in a way so that it can be read universally by any operating system.

- Insert the USB device into your computer while the VM is running.
- Eject the device from your host OS if necessary.
- In the VirtualBox VM window, click on "Devices", "USB", and the device name.
- In Ubuntu, launch the Applications menu.
- Type "Disks" into the search field.
- Click on the USB drive listed in the left column.
- Click on any boxes within the "Volumes" section.
- Click the minus (-) icon and "Delete" when prompted.
- Repeat until there are no volumes present.
- Click the "+" icon, click "Next", provide a name, choose "Fat", and click "Create".

This will create a USB drive on which files can be read and written within any file system. You cannot possess files larger than 4 GB in size, but I have never had an issue with that during investigations. When you insert the USB device into your computer, you can choose to allow your primary OS see it, or load it into your VM. I have found this method to be much more reliable than transferring large files through the shared folder. You should practice both techniques and pursue the most appropriate option for your investigations.

You should now have an understanding of the benefits of an OSINT virtual machine. Next, let's discuss the maintenance and preservation of all your hard work. I promise we are close to locking our settings in for all future investigations.

CHAPTER FIVE
VM MAINTENANCE & PRESERVATION

You now likely possess a secure computer, a custom Linux VM, several Linux applications, and shortcuts to easily launch each with user prompts for data. You could begin online investigations now, but I have a request. Delete it all. That's right, let's start over, but cheat a bit. Think about the VM you built. Did you have any issues installing software? Did you make any mistakes? Did you test the new applications with your own user data? It is likely that you slightly contaminated your perfect virtual machine while learning the process.

I do not expect you to start over at the beginning of the book, but I do ask that you now repeat the steps with a new machine. Consider the previous chapters an education, and now we will generate a new clean environment.

In the following pages, I display only the abbreviated steps taken throughout the entire previous instruction. There are no explanations or demonstrations. Consider this the cheat sheet to replicate our previous work. `All Terminal commands are in this font`. Afterward, I will explain many considerations for updates, backups, and locking in your "Original VM". I promise this will be much easier now that you have worked through the process. Many of the manual steps previously presented are now automated within many fewer steps here.

This entire list is available on my website at **https://inteltechniques.com/osintbook10/linux.txt**. Enter a username of "**`osint`**" and password of "**`book1443`**" (without quotes) if required for access. It should be easier to copy and paste commands from there versus typing them directly from here. **These steps will change as programs are updated! I will update these steps as necessary on my website. The current "linux.txt" file on my site overrides anything here.** I have split this chapter into small sections, each described below.

VM Creation (VirtualBox): Complete steps to create a standard Ubuntu VM within VirtualBox.
VM Creation (UTM): Complete steps to create a standard Ubuntu VM within UTM for Apple hosts.
VM Configuration: Steps to customize this VM with privacy and OSINT considerations.
VM Tools-Basic: Steps to install and configure all tools in the book up to this point.
VM Tools-Advanced: Steps to install and configure the advanced Linux tools required later.
VM Interface Configuration: Options to customize the appearance of your VM.
VM Complete Configuration Script: A single command to replicate all of the work.
VM Software Updates: Complete details to keep all of your new software updated.
VM Maintenance: Steps to clone and maintain your original VM in order to preserve your work.

This is my favorite chapter, let's get started.

OSINT VM Creation (VirtualBox: Windows and Linux Hosts)

- Open VirtualBox and delete any previous VMs created from this book (if desired).
- Within VirtualBox, click on the button labeled "New".
- Provide a name of "OSINT Original".
- Choose your desired location to save the machine on your host.
- Select "Linux" as the type, "Ubuntu 64-bit" as the version, then click "Continue".
- In the memory size window, move the slider to select 50% of your system memory.
- Click "Continue", then click "Create".
- Leave the hard disk file type as "VDI" and click "Continue".
- Select the default option of "Dynamically allocated" and click "Continue".
- Choose the desired size of your virtual hard drive (40 GB+) then click "Create".
- Click the "Settings" icon, then click the "Storage" icon.
- Click the CD icon which displays "Empty" in the left menu.

- Click the small blue circle to the far right in the "Optical Drive" option.
- Select "Choose a Disk File" and choose the Ubuntu ISO previously downloaded.
- Click "Open" or "Choose" if prompted, then click "OK".
- If prompted, confirm the Ubuntu ISO.
- Click "Start" in the main menu and click "Start" again if prompted.
- Only if needed, increase screen size by clicking "View", "Virtual Screen", and "Scale to 200%" from within the VirtualBox menu bar.
- Select "Install Ubuntu".
- Select your desired language and location, then click "Continue".
- Select "Normal Installation", "Download Updates", and "Install third party".
- Click "Continue".
- Select "Erase disk and install Ubuntu", then "Install Now". Confirm with "Continue".
- Choose your desired time zone and click "Continue".
- Enter a name, username, computer name, and password of "osint" for each.
- Choose "Log in automatically" and click "Continue".
- Allow Ubuntu to complete the installation, click "Restart Now", then press "Enter".
- Upon boot, Click "Skip" then "Next".
- Select "No" and then "Next" when asked to help improve Ubuntu.
- Click "Next" then "Done" to remove the welcome screen.
- If prompted to install updates, click "Remind me later".
- Continue to the section titled "Ubuntu Customization.
- In the VirtualBox Menu, select "Devices" > "Insert Guest Additions CD Image".
- Click "Run" when the dialogue box pops up, provide your password when prompted, then "Authenticate".
- Once the process is complete, press enter, and power off the VM (upper right menu).
- In VirtualBox, select your VM and click "Settings".
- In the "General" icon, click on the "Advanced" tab.
- Change "Shared clipboard" and "Drag n Drop" to "Bidirectional".
- In the "Display" icon, change the Video Memory to the maximum.
- In the "Shared Folders" icon, click the green "+".
- Click the drop-down menu under "Folder Path".
- Select "Other".
- Choose a desired folder on your host to share data and click "Open".
- Select the "Auto-mount" option and then "OK".
- Click "OK" to close the settings window.
- Reboot the VM.
- In Terminal, execute `sudo adduser osint vboxsf.`
- In Terminal, execute `sudo apt install -y build-essential dkms gcc make perl.`
- In Terminal, execute `sudo rcvboxadd setup.`
- Continue to the OSINT VM Configuration section.

OSINT VM Creation (UTM: Apple Hosts)

- Choose the "Virtualize" option and then select "Linux".
- Click the "Browse" button within UTM and select the downloaded ISO file (ARM or x86)
- Choose half of your system's memory and CPU cores.
- Set drive space to 100 MB.
- Enable File Sharing and "Save" or "Continue" until back to the main menu.
- Launch the virtual machine.
- Select "Try or Install Ubuntu".
- Enter the username of "ubuntu" and password of "password" if prompted.
- Double-click "Install Ubuntu 22.04.1 LTS" on the Desktop (ARM version).
- Select "Install Ubuntu", select your desired language and location, then click "Continue".

- Select "Normal Installation", "Download Updates", and "Install third party…".
- Click "Continue".
- Select "Erase disk and install Ubuntu", then "Install Now". Confirm with "Continue".
- Choose your desired time zone and click "Continue".
- Enter a name, username, computer name, and password of "osint" for each.
- Choose "Log in automatically" and click "Continue".
- Allow Ubuntu to complete the installation and choose "Restart Now".
- Click the "CD" icon in the upper-right of the UTM VM window; highlight the CD/DVD option, and click "Eject". THEN press "Enter" to restart the VM or the "left triangle" within UTM.
- If necessary, choose the "power" or "restart" icons in the UTM window.
- If the VM boots into the installation ISO again, shut the machine down and remove the drive.
- Upon boot, Click "Skip" then "Next".
- Select "No" and then "Next" when asked to help improve Ubuntu.
- Click "Next" then "Done" to remove the welcome screen.
- If prompted to install updates, click "Remind me later".
- Continue to the OSINT VM Configuration section.

This would be a great time to either create a snapshot (VirtualBox) or clone (VirtualBox or UTM) of this machine. This was previously explained and is demonstrated at the end of Chapter Two. Regardless of your virtual machine software, open Terminal within Ubuntu and execute the following.

OSINT VM Configuration

- `sudo apt purge -y apport apport-symptoms popularity-contest ubuntu-report whoopsie`
- `sudo apt autoremove -y`
- `sudo apt update`
- Click on "Start" to restart your Ubuntu VM.
- Resize the window if desired.
- Resize the VM if desired ("View" > "Virtual Screen" > "Scale").
- In the left Dock, right-click and eject the CD.
- Click the Applications Menu (9 dots) in the lower left and launch Settings.
- Click "Notifications" and disable both options.
- Click the "Privacy" option, then click "Screen Lock" and disable all options.
- Click "File History & Trash", then disable the option.
- Click "Diagnostics", then change to "Never".
- Click the back arrow and click Power, changing "Blank Screen" to "Never".
- Click "Automatic Suspend" and disable (turn off) the feature.
- Close all Settings windows.
- Click the Applications Menu and launch Software Updater.
- Click "Install Now" to apply all updates.
- If prompted, restart the VM and power off when reboot is complete.

If desired, create another snapshot or clone of your current environment.

OSINT VM Tools-Basic

In the previous chapter, I explained each manual step required to install and configure dozens of applications, services, scripts, and other customizations. The following steps replicate every portion of this instruction and should simplify the creation process. Nothing in here is new, it is simply a summary of all steps. First, restart the "OSINT Original" VM, then launch and close Firefox. Click the Applications Menu, launch Terminal, and execute the following commands. The "2>/dev/null" additions hide errors and warnings.

```
sudo snap remove --purge firefox
sudo add-apt-repository -y ppa:mozillateam/ppa
echo '
Package: *
Pin: release o=LP-PPA-mozillateam
Pin-Priority: 1001
' | sudo tee /etc/apt/preferences.d/mozilla-firefox
sudo apt install -y firefox
sudo apt install -y curl
cd ~/Desktop
curl -O https://inteltechniques.com/data/osintbook10/ff-template.zip
unzip ff-template.zip -d ~/.mozilla/firefox/
cd ~/.mozilla/firefox/ff-template/
cp -R * ~/.mozilla/firefox/*.default-release
cd ~/Desktop && rm ff-template.zip
sudo snap install chromium
sudo apt install torbrowser-launcher -y
sudo apt install -y vlc
sudo apt install -y ffmpeg
sudo apt install -y python3-pip
sudo pip install -U youtube-dl
sudo pip install -U yt-dlp
cd ~/Desktop
curl -O https://inteltechniques.com/data/osintbook10/vm-files.zip
unzip vm-files.zip -d ~/Desktop/
mkdir ~/Documents/scripts && mkdir ~/Documents/icons
cd ~/Desktop/vm-files/scripts && cp * ~/Documents/scripts
cd ~/Desktop/vm-files/icons && cp * ~/Documents/icons
cd ~/Desktop/vm-files/shortcuts && sudo cp * /usr/share/applications/
cd ~/Desktop && rm vm-files.zip && rm -rf vm-files
sudo apt install -y python3-venv
mkdir ~/Downloads/Programs
mkdir ~/Downloads/Programs/Streamlink
cd ~/Downloads/Programs/Streamlink
python3 -m venv streamlinkEnvironment
source streamlinkEnvironment/bin/activate
sudo pip install streamlink 2>/dev/null
deactivate
sudo apt install -y git
mkdir ~/Downloads/Programs/Instalooter
cd ~/Downloads/Programs/Instalooter
python3 -m venv instalooterEnvironment
source instalooterEnvironment/bin/activate
sudo pip install instalooter 2>/dev/null
deactivate
mkdir ~/Downloads/Programs/Instaloader
cd ~/Downloads/Programs/Instaloader
python3 -m venv instaloaderEnvironment
source instaloaderEnvironment/bin/activate
sudo pip install instaloader 2>/dev/null
```

```
deactivate
mkdir ~/Downloads/Programs/Toutatis
cd ~/Downloads/Programs/Toutatis
python3 -m venv toutatisEnvironment
source toutatisEnvironment/bin/activate
sudo pip install toutatis 2>/dev/null
deactivate
cd ~/Downloads/Programs
git clone https://github.com/Datalux/Osintgram.git
cd Osintgram
python3 -m venv OsintgramEnvironment
source OsintgramEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
sudo apt install libncurses5-dev libffi-dev -y
mkdir ~/Downloads/Programs/Gallery-DL
cd ~/Downloads/Programs/Gallery-DL
python3 -m venv gallerydlEnvironment
source gallerydlEnvironment/bin/activate
sudo pip install -U gallery-dl 2>/dev/null
deactivate
cd ~/Downloads
sudo apt install default-jre -y
wget https://github.com/ripmeapp/ripme/releases/latest/download/ripme.jar
chmod +x ripme.jar
cd ~/Downloads/Programs
git clone https://github.com/sherlock-project/sherlock.git
cd sherlock
python3 -m venv SherlockEnvironment
source SherlockEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
mkdir ~/Downloads/Programs/socialscan
cd ~/Downloads/Programs/socialscan
python3 -m venv socialscanEnvironment
source socialscanEnvironment/bin/activate
sudo pip install -U socialscan 2>/dev/null
deactivate
mkdir ~/Downloads/Programs/holehe
cd ~/Downloads/Programs/holehe
python3 -m venv holeheEnvironment
source holeheEnvironment/bin/activate
sudo pip install -U holehe 2>/dev/null
deactivate
cd ~/Downloads/Programs
git clone https://github.com/WebBreacher/WhatsMyName.git
cd WhatsMyName/whatsmyname
python3 -m venv WhatsMyNameEnvironment
source WhatsMyNameEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs
git clone https://github.com/p1ngul1n0/blackbird
cd blackbird
python3 -m venv blackbirdEnvironment
source blackbirdEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
```

```
cd ~/Downloads/Programs
git clone https://github.com/martinvigo/email2phonenumber.git
cd email2phonenumber
python3 -m venv email2phonenumberEnvironment
source email2phonenumberEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
sudo pip install pipx
pipx ensurepath
pipx install ghunt
pipx ensurepath
cd ~/Downloads/Programs
git clone https://github.com/ChrisTruncer/EyeWitness.git
cd EyeWitness/Python/setup && sudo ./setup.sh
wget https://github.com/mozilla/geckodriver/releases/download/
v0.32.0/geckodriver-v0.32.0-linux-aarch64.tar.gz
tar -xvzf geckodriver* && chmod +x geckodriver
sudo mv geckodriver /usr/local/binsudo snap install amass
cd ~/Downloads/Programs
git clone https://github.com/aboul3la/Sublist3r.git
cd Sublist3r
python3 -m venv Sublist3rEnvironment
source Sublist3rEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs
git clone https://github.com/s0md3v/Photon.git
cd Photon
python3 -m venv PhotonEnvironment
source PhotonEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs
git clone https://github.com/laramies/theHarvester.git
cd theHarvester
python3 -m venv theHarvesterEnvironment
source theHarvesterEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
sudo pip install testresources 2>/dev/null
sudo pip install webscreenshot 2>/dev/null
cd ~/Downloads/Programs
git clone https://github.com/Lazza/Carbon14
cd Carbon14
python3 -m venv Carbon14Environment
source Carbon14Environment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
sudo apt install -y mediainfo-gui
sudo apt install -y libimage-exiftool-perl
sudo apt install -y mat2
mkdir ~/Downloads/Programs/xeuledoc
cd ~/Downloads/Programs/xeuledoc
python3 -m venv xeuledocEnvironment
source xeuledocEnvironment/bin/activate
sudo pip install -U xeuledoc 2>/dev/null
deactivate
cd ~/Downloads/Programs
```

```
sudo apt install subversion -y
git clone https://github.com/GuidoBartoli/sherloq.git
cd sherloq/gui
python3 -m venv sherloqEnvironment
source sherloqEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
sudo apt install -y httrack webhttrack
sudo apt install -y libcanberra-gtk-module
cd ~/Downloads/Programs
git clone https://github.com/opsdisk/metagoofil.git
cd metagoofil
python3 -m venv metagoofilEnvironment
source metagoofilEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
mkdir ~/Downloads/Programs/bdfr
cd ~/Downloads/Programs/bdfr
python3 -m venv bdfrEnvironment
source bdfrEnvironment/bin/activate
sudo pip install -U bdfr 2>/dev/null
deactivate
mkdir ~/Downloads/Programs/redditsfinder
cd ~/Downloads/Programs/redditsfinder
python3 -m venv redditsfinderEnvironment
source redditsfinderEnvironment/bin/activate
sudo pip install -U redditsfinder 2>/dev/null
deactivate
cd ~/Downloads/Programs
git clone https://github.com/MalloyDelacroix/DownloaderForReddit.git
cd DownloaderForReddit
python3 -m venv DownloaderForRedditEnvironment
source DownloaderForRedditEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
wget http://dl.google.com/dl/earth/client/current/google-earth-
stable_current_amd64.deb
sudo apt install -y ./google-earth-stable_current_amd64.deb
sudo rm google-earth-stable_current_amd64.deb
sudo apt install -y kazam
sudo apt install bleachbit
sudo apt update --fix-missing
sudo apt -y upgrade
sudo apt --fix-broken install
cd ~/Desktop
curl -O https://inteltechniques.com/data/osintbook10/tools.zip
unzip tools.zip -d ~/Desktop/
rm tools.zip
```

You should now possess a virtual machine which includes every script, icon, shortcut, application, and configuration discussed within this book so far. While you may feel tempted to play around with your new VM, please don't. We want to keep it clean and have more work to finish. If you launch your Applications menu, you should see new programs and scripts as previously explained. Next, let's apply the "advanced" settings which are explained later in the book. We want a complete machine which can be "locked-in" for future use. The following steps prepare your VM for future chapters.

```
mkdir ~/Downloads/Programs/waybackpy
cd ~/Downloads/Programs/waybackpy
python3 -m venv waybackpyEnvironment
source waybackpyEnvironment/bin/activate
sudo pip install -U waybackpy 2>/dev/null
deactivate
mkdir ~/Downloads/Programs/changedetection
cd ~/Downloads/Programs/changedetection
python3 -m venv changedetectionEnvironment
source changedetectionEnvironment/bin/activate
sudo pip install changedetection.io 2>/dev/null
deactivate
mkdir ~/Downloads/Programs/archivebox
cd ~/Downloads/Programs/archivebox
python3 -m venv archiveboxEnvironment
source archiveboxEnvironment/bin/activate
sudo pip install archivebox
deactivate
mkdir ~/Documents/archivebox
cd ~/Documents/archivebox
archivebox init
cd ~/Downloads/Programs
git clone https://github.com/smicallef/spiderfoot.git
cd spiderfoot
python3 -m venv spiderfootEnvironment
source spiderfootEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs
git clone https://github.com/lanmaster53/recon-ng.git
cd recon-ng
python3 -m venv recon-ngEnvironment
source recon-ngEnvironment/bin/activate
sudo pip install -r REQUIREMENTS 2>/dev/null
deactivate
mkdir ~/Downloads/Programs/internetarchive
cd ~/Downloads/Programs/internetarchive
python3 -m venv internetarchiveEnvironment
source internetarchiveEnvironment/bin/activate
sudo pip install -U internetarchive 2>/dev/null
deactivate
sudo apt install -y ripgrep
cd ~/Downloads/Programs
git clone https://github.com/AmIJesse/Elasticsearch-Crawler.git
sudo pip install nested-lookup 2>/dev/null
sudo apt install -y jq
mkdir ~/Downloads/Programs/search-that-hash
cd ~/Downloads/Programs/search-that-hash
python3 -m venv search-that-hashEnvironment
source search-that-hashEnvironment/bin/activate
sudo pip install -U search-that-hash 2>/dev/null
```

```
deactivate
mkdir ~/Downloads/Programs/name-that-hash
cd ~/Downloads/Programs/name-that-hash
python3 -m venv name-that-hashEnvironment
source name-that-hashEnvironment/bin/activate
sudo pip install -U name-that-hash 2>/dev/null
deactivate
mkdir ~/Downloads/Programs/h8mail
cd ~/Downloads/Programs/h8mail
python3 -m venv h8mailEnvironment
source h8mailEnvironment/bin/activate
sudo pip install -U h8mail 2>/dev/null
deactivate
cd ~/Downloads
h8mail -g
sed -i 's/\;leak\-lookup\_pub/leak\-lookup\_pub/g' h8mail_config.ini
```

You should now have an OSINT VM which is ready for the Advanced Linux techniques discussed much later in the book. It will be convenient to maintain a single OSINT VM which is applicable to this entire book instead of "Basic" and "Advanced" versions. We have a lot to get through before we are ready for the advanced section, but we will be able to pick up where we left off without any additional installation or configuration. We can focus only on the usage of these tools without redundant explanation of script creation, icon placement, and desktop shortcuts. Your Linux Ubuntu VM is now completely functional but not very pretty. You might prefer it this way. Many Linux experts enjoy the bland look and requirements to hunt for features as needed. I do not. The next section simplifies several modifications to the graphical interface.

OSINT VM Interface Configuration

Now that your VM contains numerous custom scripts and applications, you may want to customize the appearance. You could open the Applications menu, right-click on each new OSINT shortcut, and add them to your Dock, but that is very time-consuming. The following commands change the background; clear the entire Dock; adjust the Dock position; place all desktop shortcuts within your Dock for easy access; and decrease the icon size. These commands are included in the previously mentioned "linux.txt" file.

```
gsettings set org.gnome.desktop.background picture-uri ''
gsettings set org.gnome.desktop.background primary-color 'rgb(66, 81, 100)'
gsettings set org.gnome.shell favorite-apps []
gsettings set org.gnome.shell.extensions.dash-to-dock dock-position BOTTOM
gsettings set org.gnome.shell favorite-apps "['firefox.desktop',
'chromium_chromium.desktop', 'torbrowser.desktop',
'org.gnome.Nautilus.desktop', 'org.gnome.Terminal.desktop', 'updates.desktop',
'tools.desktop', 'video-download.desktop', 'video-utilities.desktop', 'video-
stream.desktop', 'instagram.desktop', 'galleries.desktop', 'users-
emails.desktop', 'captures.desktop', 'domains.desktop', 'metadata.desktop',
'archives.desktop', 'documents.desktop', 'breaches-leaks.desktop',
'reddit.desktop', 'spiderfoot.desktop', 'recon-ng.desktop', 'api.desktop',
'google-earth-pro.desktop', 'kazam.desktop', 'gnome-control-center.desktop']"
gsettings set org.gnome.shell.extensions.dash-to-dock dash-max-icon-size 32
```

You should now possess a final OSINT VM ready to clone for any investigation. Keep this machine updated and only use it to clone into additional machines for each investigation. Never browse the internet or conduct any investigations from within this clean VM. At the time of this writing, I possessed three VMs within VirtualBox, as follows, and seen in Figures 5.01 and 5.02.

- **OSINT Original:** My final VM which is the source of any clones used for investigations. I keep it updated and apply any changes as needed.
- **Ubuntu Install:** A VM which only contains a basic installation of Ubuntu. I keep a snapshot available which I can revert to if I use this machine to experiment with Ubuntu configurations.
- **Ubuntu Install & Configuration:** A VM of Ubuntu with all custom settings and VirtualBox tools installed. I also keep a snapshot ready here. I can test new software without worry of any conflicts from my final VM.

If I were launching a new online investigation, I would right-click the "OSINT Original" machine (while powered down) and select "Clone". I would then create a "Full Clone" and title it appropriately for my investigation, as explained in just a moment. Once a month, I conduct the following steps to make sure everything is ready for future investigations.

- Open VirtualBox or UTM and launch my OSINT Original Ubuntu VM.
- Apply all updates through the "Updates" script explained in a moment.

- Apply all system updates through the native Ubuntu Software Updater.
- Run BleachBit and eliminate unnecessary files.
- Execute the proper shrinking utility for the VM software.
- Create a clone of the VM.

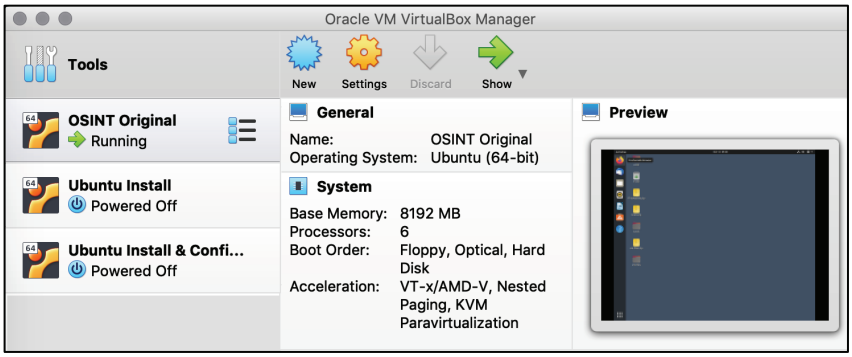


Figure 5.01: A VirtualBox menu of OSINT VMs ready for use.

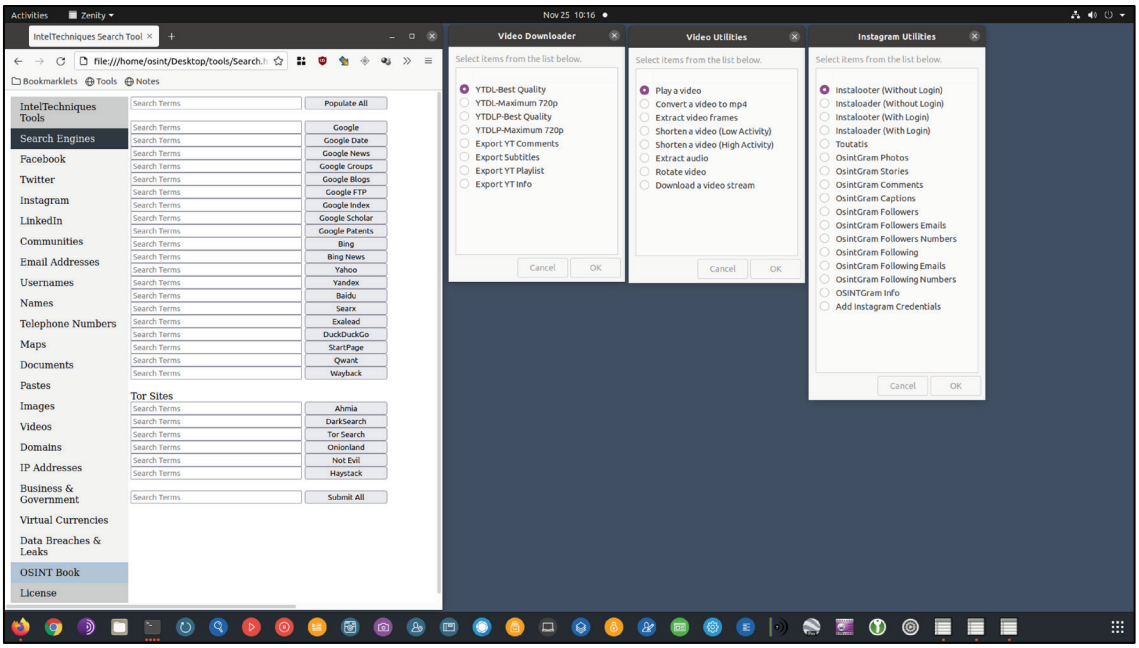


Figure 5.02: The final Linux OSINT VM with customized applications and appearance.

OSINT VM Complete Configuration Script

While this chapter has abbreviated the steps to build your own OSINT VM, you may still feel overwhelmed with the required effort. In late 2019, my colleague Jesse created a single script which replicates every step we have discussed up to this point, including the advanced OSINT Linux tools coming up later. I modified this script to include every Linux configuration, installation, and customization mentioned throughout this entire book. After you build your Ubuntu virtual machine within VirtualBox by conducting the steps previously explained, launch the following command from within Terminal. You will be prompted to enter your password at least once. After the process completes, you possess the same machine which was built during the tutorials throughout this entire book.

```
wget --user osint --password book1443 https://inteltechniques.com/
osintbook10/linux.sh && chmod +x linux.sh && ./linux.sh
```

You may feel frustrated with me. You may wonder why I did not start the book with this script. While I rely on this single script often, it is cheating. Running a single command and achieving a complete VM is convenient and time-saving, but it also eliminates any need to understand the processes. I believe the education received while manually building an OSINT VM is more valuable than the final product itself. However, this automated script simplifies the process when we need to quickly create another OSINT VM. It also allows me to apply updates as needed from my end. You could launch this script a year after reading the book and immediately apply all updates and changes which have occurred since publication.

An ideal scenario would be that you are already familiar with the VM creation and configuration process, but you do not have a configured VM from which to clone. You have an Ubuntu VM, but no OSINT applications. Entering these two commands within the Terminal of any Ubuntu installation should build your OSINT VM in about 10 minutes. This script could also be launched from an Ubuntu host. If you had an old laptop with no other purpose, you could install Ubuntu as the host operating system and run this script.

If you look at the script after download, which should be available within your "Home" folder inside your Ubuntu install, you will see that it appears very similar to the text file with all of the Linux commands (**https://inteltechniques.com/osintbook10/linux.txt**). This new script is simply executing each line as we did manually throughout the book. While it can be a valuable time saver, you also risk missing any errors which occur during execution. I encourage you to ignore this script until you have confidence in your ability to create your virtual machine manually. I find it more satisfying to use a VM which I created myself instead of one generated by an automated script, but I want you to have options. If this script should fail on your VM, revert to the manual methods in order to identify the issue.

OSINT VM Software Updates

Once you have the custom VM installed, immediately apply all updates and reboot. You can do this with either the "Updates" script explained in a moment or the manual commands below. Next, assume that you have not touched your OSINT Original virtual machine in some time, and you are ready to launch a new investigation. You likely have software updates which need to be applied. Instead of cloning and updating every machine, launch your OSINT Original VM and conduct the following within Terminal. These commands will update your operating system, installed applications, and custom programs created during the previous chapter.

```
sudo apt update
sudo apt -y upgrade
sudo snap refresh
sudo apt update --fix-missing
sudo apt --fix-broken install
sudo pip install -U youtube-dl 2>/dev/null
sudo pip install -U yt-dlp 2>/dev/null
cd ~/Downloads/Programs/Streamlink
source streamlinkEnvironment /bin/activate
sudo pip install -U streamlink 2>/dev/null
deactivate
cd ~/Downloads/Programs/Instalooter
source instalooterEnvironment /bin/activate
sudo pip install -U instalooter 2>/dev/null
deactivate
cd ~/Downloads/Programs/Instaloader
source instaloaderEnvironment /bin/activate
sudo pip install -U instaloader 2>/dev/null
deactivate
cd ~/Downloads/Programs/Toutatis
source toutatisEnvironment /bin/activate
sudo pip install -U toutatis 2>/dev/null
```

```
deactivate
cd ~/Downloads/Programs/Osintgram
git pull https://github.com/Datalux/Osintgram.git
source OsintgramEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs/Gallery-DL
source gallerydlEnvironment /bin/activate
sudo pip install -U gallery-dl 2>/dev/null
deactivate
cd ~/Downloads
wget -N https://github.com/ripmeapp/ripme/releases/latest/download/ripme.jar
chmod +x ripme.jar
cd ~/Downloads/Programs/sherlock
git pull https://github.com/sherlock-project/sherlock.git
source SherlockEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs/socialscan
source socialscanEnvironment/bin/activate
sudo pip install -U socialscan 2>/dev/null
deactivate
cd ~/Downloads/Programs/holehe
source holeheEnvironment/bin/activate
sudo pip install -U holehe 2>/dev/null
deactivate
cd ~/Downloads/Programs/WhatsMyName
git pull https://github.com/WebBreacher/WhatsMyName.git
cd whatsmyname
source WhatsMyNameEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs/blackbird
git pull https://github.com/p1ngul1n0/blackbird
source blackbirdEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs/email2phonenumber
git pull https://github.com/martinvigo/email2phonenumber.git
source email2phonenumberEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
sudo pip install -U pipx
pipx upgrade ghunt
cd ~/Downloads/Programs/EyeWitness
git pull https://github.com/ChrisTruncer/EyeWitness.git
cd Python/setup
sudo ./setup.sh
cd ~/Downloads/Programs/Sublist3r
git pull https://github.com/aboul3la/Sublist3r.git
source Sublist3rEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs/Photon
git pull https://github.com/s0md3v/Photon.git
source PhotonEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
```

```
cd ~/Downloads/Programs/theHarvester
git pull https://github.com/laramies/theHarvester.git
source theHarvesterEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
sudo pip install -U testresources 2>/dev/null
sudo pip install -U webscreenshot 2>/dev/null
cd ~/Downloads/Programs/Carbon14
git pull https://github.com/Lazza/Carbon14
source Carbon14Environment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs/xeuledoc
source xeuledocEnvironment/bin/activate
sudo pip install -U xeuledoc 2>/dev/null
deactivate
cd ~/Downloads/Programs/sherloq
git pull https://github.com/GuidoBartoli/sherloq.git
cd gui
source sherloqEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs/metagoofil
git pull https://github.com/opsdisk/metagoofil.git
source metagoofilEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs/bdfr
source bdfrEnvironment/bin/activate
sudo pip install -U bdfr 2>/dev/null
deactivate
cd ~/Downloads/Programs/redditsfinder
source redditsfinderEnvironment/bin/activate
sudo pip install -U redditsfinder 2>/dev/null
deactivate
cd ~/Downloads/Programs/DownloaderForReddit
git pull https://github.com/MalloyDelacroix/DownloaderForReddit.git
source DownloaderForRedditEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
wget http://dl.google.com/dl/earth/client/current/google-earth-
stable_current_amd64.deb
sudo apt install -y ./google-earth-stable_current_amd64.deb
sudo rm google-earth-stable_current_amd64.deb
cd ~/Downloads/Programs/waybackpy
source waybackpyEnvironment/bin/activate
sudo pip install -U waybackpy 2>/dev/null
deactivate
cd ~/Downloads/Programs/changedetection
source changedetectionEnvironment/bin/activate
sudo pip install -U changedetection.io 2>/dev/null
deactivate
cd ~/Downloads/Programs/archivebox
source archiveboxEnvironment/bin/activate
sudo pip install -U archivebox
deactivate
cd ~/Downloads/Programs/spiderfoot
git pull https://github.com/smicallef/spiderfoot.git
```

```
source spiderfootEnvironment/bin/activate
sudo pip install -r requirements.txt 2>/dev/null
deactivate
cd ~/Downloads/Programs/recon-ng
git pull https://github.com/lanmaster53/recon-ng.git
source recon-ngEnvironment/bin/activate
sudo pip install -r REQUIREMENTS 2>/dev/null
deactivate
cd ~/Downloads/Programs/internetarchive
source internetarchiveEnvironment/bin/activate
sudo pip install -U internetarchive 2>/dev/null
deactivate
cd ~/Downloads/Programs/Elasticsearch-Crawler
git pull https://github.com/AmIJesse/Elasticsearch-Crawler.git
sudo pip install -U nested-lookup 2>/dev/null
cd ~/Downloads/Programs/search-that-hash
source search-that-hashEnvironment/bin/activate
sudo pip install -U search-that-hash 2>/dev/null
deactivate
cd ~/Downloads/Programs/name-that-hash
source name-that-hashEnvironment/bin/activate
sudo pip install -U name-that-hash
deactivate
cd ~/Downloads/Programs/h8mail
source h8mailEnvironment/bin/activate
sudo pip install -U h8mail 2>/dev/null
deactivate
```

OSINT VM Software Update Script

I believe these update commands should be executed as often as possible, and always immediately following a VM creation. You always want your OSINT Original VM to have the latest software. Manually entering each of these commands on a daily or weekly basis is exhausting. This is why I have created a script to update everything we have installed in our OSINT Original VM. This file is included within the scripts folder of the Linux "vm-files" archive previously downloaded, and the content can be seen by opening "updates.sh" within your Ubuntu VM. You can launch this file by clicking the "Update Scripts" application in the Applications menu or Dock of your OSINT Original VM. Look for the circular arrow icon in the lower Dock next to the Terminal icon.

This script also includes the basic commands to update your operating system files and stock applications. The first five commands address this issue. Everything else focuses on the customizations made throughout this section of the book. After the script finishes, you should see "Updates Complete!" within Terminal. You can now close the Terminal window knowing your system has been completely updated. While the "Software Updates" icon within Ubuntu's Application menu updates your overall Ubuntu environment, it does not update the OSINT applications. This is a better method.

OSINT VM Maintenance

You should now have a completely functional and updated virtual machine titled "OSINT Original". This is your clean machine with no contamination from any investigation or testing. It has all the software we want, and it is ready to be used. As a final reminder, you must run BleachBit to eliminate excess waste, and then shrink the VM based on the tutorials within Chapter Two. Next, let's consider an "OSINT Test" machine. This is the VM on which you can practice Linux commands, test new programs, or create new scripts. It is a VM which will never be used for any investigations. Its sole purpose is to give you a safe playground to experiment. Complete the following tasks within VirtualBox (revisit the tutorials within Chapter Two for UTM instructions for Apple devices).

- Right-click the VM titled "OSINT Original", click "Clone", and title it "OSINT Test".
- Supply the desired storage location and click "Continue".
- Select "Full Clone" and click the "Clone" button.

You now have a fully functional cloned "Test VM". Any activity within that machine will not change anything in any other VMs. Repeat this cloning process any time you wish to conduct an investigation. In this scenario, assume I created a new clone titled Case #2021-143. I can open this new VM which appears identical to my original. I can conduct my investigation and close the machine. All evidence is stored within the machine and can be extracted to my shared folder or USB drive if desired. All of my VirtualBox VMs are visible in Figure 5.03 (left). Figure 5.03 (right) displays the VirtualBox Snapshots menu option and an example of a Snapshot ready for restoration. Figure 5.04 displays my UTM environment as I was writing this chapter.

Once you have your Original VM created, configured, and updated, it is best to export a copy as a backup. You have put a lot of time into this, and I would hate to see you lose the hard work. If your computer or storage device would crash beyond repair, you would need to start over. If your Original VM would become corrupted, restoring the data would be quite difficult. I keep an exported copy of my Original VM on a USB drive in case of emergency. In VirtualBox, conduct the following steps after shutting down a VM.

- Single-click on the VM within the VirtualBox menu.
- Click on "File" in the menu, "Export Appliance", confirm the selection, and click "Continue".
- Choose the "File" export location on your drive and click "Continue" then "Export".

In UTM, conduct the following.

- Right-click the desired VM and choose the "Show in Finder" option.

- Identify the .utm file which corresponds to your VM.
- Copy and paste the file anywhere desired.

Both options produce a single large file which contains your entire OSINT Original VM. You could import this file into any instance of your VM software at any time. Some strategies follow.

- I launch your OSINT Original VM often and apply all updates.
- Export your OSINT Original VM monthly as a backup.
- Conduct all software auditing and install new apps for testing on the OSINT Test VM.
- Create a new clone of the OSINT Original for every new investigation.
- At the end of the investigation, export all evidence to an external drive.
- If necessary, create an exported copy of investigation VMs for later review.
- Delete investigation VMs when no longer needed.

This plan ensures that every investigation is completed on a clean VM with absolutely no contamination from previous investigations. My exported investigation VMs can be provided to other investigators or as part of discovery in litigation. I am prepared to testify with confidence, if required.

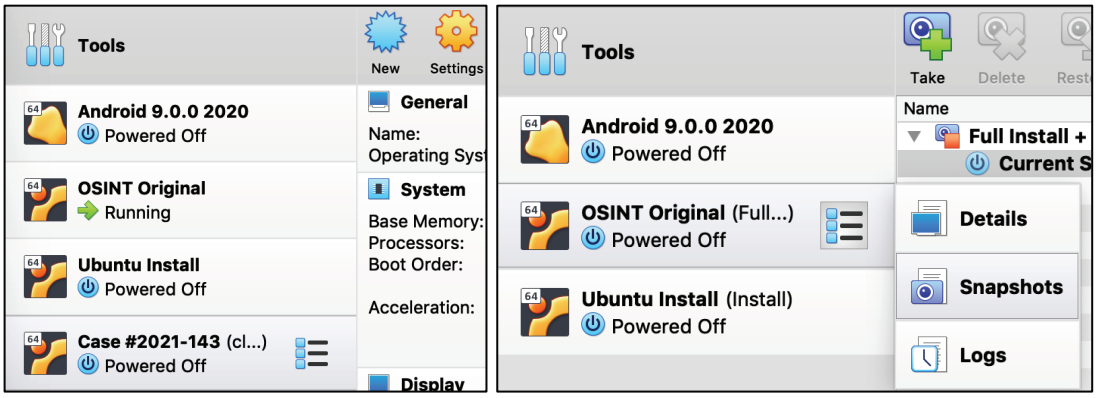


Figure 5.03: Cloned machines and snapshots within VirtualBox.

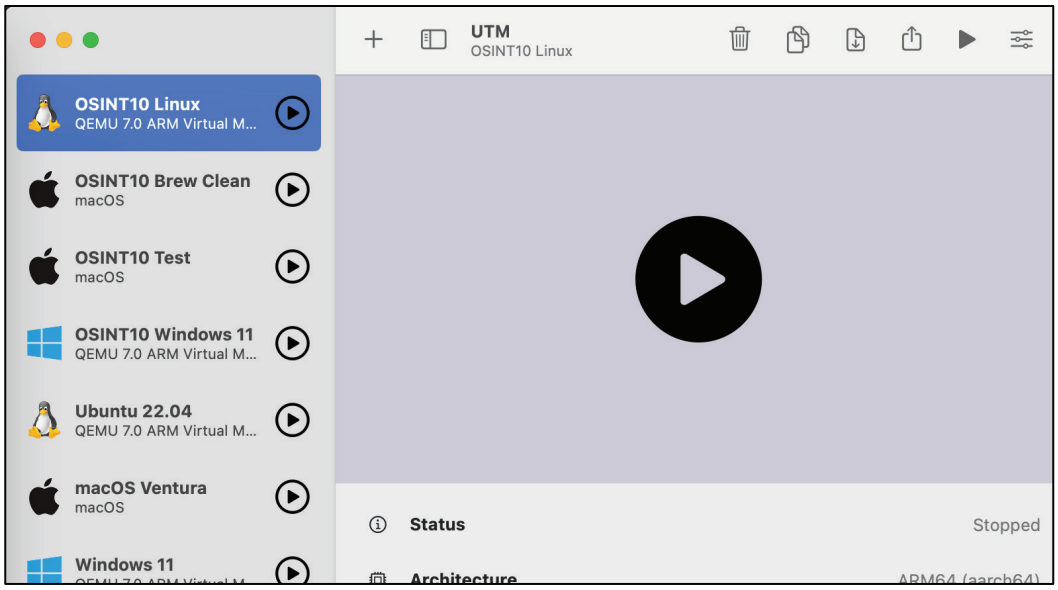


Figure 5.04: The UTM interface.

Windows VM

While I prefer to conduct all investigations solely inside Linux, I respect that there may be a need for a Windows VM. In fact, I am using one now. I write all of my books within an offline copy of Microsoft Word. I create protected press-ready PDFs with Adobe Acrobat Pro. Neither of these applications run reliably on a Linux machine, and my personal laptop possesses Pop!_OS as the host operating system. Therefore, I keep a Windows VM for all writing. Installing Windows inside VirtualBox is not difficult, but licensing may be an issue. Therefore, we will rely on the official Microsoft Windows 10 VM available directly from their website.

- Navigate to https://developer.microsoft.com/en-us/microsoft-edge/tools/vms/.
- Choose "MSEdge on Win10" as the Virtual Machine.
- Choose "VirtualBox" as the Platform, and click "Download Zip".
- Unzip the downloaded file and keep a copy with your other VM backups.
- In VirtualBox, click "File" and "Import Appliance" in the menu.
- Choose the "ovf" file which you extracted from the zip file and click "Continue".
- Make any desired modifications as previously explained and click "Import".
- In the VirtualBox menu, click "Settings" then "Storage".
- Click the first "+" to add an optical drive, click "Leave empty", and click "OK".
- Before launching, create a snapshot of the VM as previously explained.
- Double-click the new Windows 10 machine to launch.
- Enter the password of "Passw0rd!" to enter Windows.
- In the VirtualBox menu, click "Devices" and "Insert Guest Additions CD".
- Through the Files explorer, double-click the mounted CD and choose "Yes".
- Click "Next", "Next", and "Install" to configure the default options.
- Reboot when prompted.

You now have a fully functioning and legal Windows 10 VM at your disposal. You can resize the window as desired and install any Windows applications. You can configure the copy and paste options, shared folder, or any other customizations as previously demonstrated. This is a 90-day trial, at which time the VM will no longer boot. You can revert to the original snapshot you created at any time to restart the 90-day trial. Surprisingly, this is allowed and encouraged from Microsoft. You will notice branding within the desktop advising it is a trial. If that bothers you, you must acquire a license and install from traditional media. This method is the easiest (and cheapest) option to possess a legal copy of Windows within a VM at no cost.

Summary

Hopefully, you now have a Linux virtual machine in pristine condition ready for your investigations. Keep your original clean and only use it for updates. Make clones of it for each investigation, but be sure to shrink it before cloning. Also, make sure you have a safe backup copy stored outside your primary device. Most importantly, understand the steps you have taken. It is very likely that you will need to modify some of these commands as things change from the software developers. If you should receive an error on any specific step in these tutorials, search that exact error online and you should be presented many solutions.

The chances of every application mentioned here installing without any issue is slim. Programs break, updates cause issues, and the countless variables on your system can be problematic. When something fails, keep moving on to the other options. Most of the time, a future update, which you will receive during your update process, will fix things which currently do not work. I send a sincere "Thank you" to David Westcott for opening my eyes to the many ways in which Linux can be customized for our needs as online investigators.

CHAPTER SIX
MAC & WINDOWS HOSTS

I was originally conflicted writing this chapter. I previously explained the benefits of Linux virtual machines for OSINT work and rely on them daily. I encourage everyone to consider them as a way to protect the integrity of online investigations while isolating each case. However, VMs are not for everyone. Since the seventh edition, I have heard from many readers asking to replicate the Linux tools within Mac and Windows. Some readers do not have hardware which supports virtualization. Some do not have the permission from their employers to install virtual machine software due to licensing concerns.

Many simply are not comfortable with Linux and want an easier solution. You may just want the ability to practice within an operating system with which you are familiar. These are all legitimate reasons to avoid Linux and I hold no judgement. I must confess that I often download videos directly from my Mac with my own scripts. Regardless of your reasons, you may want to create an OSINT machine within native Mac or Windows. I believe the ability to replicate some Linux programs within Mac and Windows will encourage non-technical readers to embrace these options. The tutorials within this section assume the following.

- You want to replicate some of the Linux VM applications and settings within your Mac or Windows computer or virtual machine.
- You understand that conducting investigations within a native Mac or Windows host, without the use of snapshots or clones, may contaminate your online evidence.
- You understand that reversing all of these settings and eliminating these applications may require you to reinstall macOS or Windows.

I begin with Mac because it is very similar to Ubuntu. In fact, Mac and Ubuntu are both based on a UNIX file system and most of the commands are cross-compatible. We will need to make many modifications to the operating system and scripts, but the basic usage will seem very familiar. I then explain the limited process for Windows. Every command within this chapter is available in digital form at the following locations. Similar to the previous tutorials, enter a username of "**osint**" and password of "**book1443**" (without quotes), if required, for access to these resources. Always copy text commands from the digital source instead of attempting to manually retype them.

https://inteltechniques.com/osintbook10/mac.txt
https://inteltechniques.com/osintbook10/windows.txt

All tutorials presented within this chapter assume you are working with a new Mac or Windows machine or VM. Pre-existing installations should function fine, but may conflict with some steps. The software and tutorials are provided "as is", without warranty of any kind. In no event shall the author be liable for any claim, damages or other liability, arising from, out of, or in connection with the software or tutorials presented here (this makes my lawyer happy).

Throughout these tutorials, you will encounter many issues and complications while adapting Linux applications into macOS and Windows builds. Please do not expect this chapter to create a perfectly-functioning replica of our Linux VM within macOS or Windows. Instead, please consider this an overall lesson about the possibilities regarding Linux-based applications within alternative operating systems. If you find a Linux program you would like to have available outside of your Ubuntu VM, the tutorials here can help make that happen. However, you will always encounter some instability and non-functioning applications. Remember that Linux programs were intended for Linux, and your Ubuntu OSINT VM will always perform better than anything in this chapter. However, we should also understand that some OSINT programs for Linux will work great within other OS's. Hopefully I have not scared you off, let's get started.

Order 0138471223

Mac OSINT Host

The first step is to install Rosetta on Apple devices which have an M1, M2, or newer processor. This allows some software which does not have an ARM-based version to launch within newer machines.

```
softwareupdate --install-rosetta --agree-to-license
```

The next step is to install Brew. This is a package (software) manager which was mentioned in Chapter One when discussing antivirus options for Mac computers. If you installed Brew then, you do not need to repeat the process. If you did not, the following command within Terminal will install and configure the software. There is no harm executing this command a second time.

```
/bin/bash -c "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/master/install.sh)"
```

This command may take a long time to complete. Be sure to complete all required steps, including the commands presented after the installation! It may ask you for your password a couple of times, so you should monitor occasionally. I had to enter my password twice and confirm installation with "enter" once.

Next, we want to install several of the Linux applications mentioned within the previous chapters onto our native Mac machine using Brew. The first two commands can also take a long time to complete, but the third should run quickly. You may be asked for your password. The following commands configure the majority of our basic programs for immediate use.

```
brew install exiftool
brew install ffmpeg
brew install fileicon
brew install git
brew install httrack
brew install internetarchive
brew install libmagic
brew install mat2
brew install name-that-hash search-that-hash
brew install phantomjs
brew install pipx
brew install ripgrep
brew install streamlink
brew install wget
brew install yt-dlp
brew install youtube-dl
brew install zenity
brew install --cask firefox google-chrome tor-browser
brew install --cask google-earth-pro
brew install --cask mediainfo
brew install --cask vlc
brew install --cask xquartz
brew tap caffix/amass && brew install amass
```

If any of these packages become unavailable, you will receive an error preventing all installations. If this happens, remove the missing package from these commands. After you have installed all of the Brew options, run the following commands to make sure everything is set up appropriately. If you receive any errors, follow the provided guidance within each dialogue.

```
brew autoremove
```

```
brew cleanup -s
rm -rf "$(brew --cache)"
brew doctor
brew missing
```

Next, configure the Firefox browser with all extensions and settings as previously explained. **Open Firefox and close it completely**, then execute the following within Terminal.

```
cd ~/Desktop
curl -O https://inteltechniques.com/data/osintbook10/ff-template.zip
unzip ff-template.zip && cd ff-template
cp -R * ~/Library/Application\ Support/Firefox/Profiles/*.default-release
cd ~/Desktop && rm -rf ff-template __MACOSX
```

Note that pre-existing Firefox installations may conflict with this tutorial. If needed, manually copy my profile to yours using the methods explained in Chapter Three. Now that we have Firefox configured, we need all of the custom scripts, links, icons, and tools which were present within the Linux VM. Conduct the following in Terminal. This will also remove any downloaded files which are no longer needed.

```
cd ~/Desktop
curl -O https://inteltechniques.com/data/osintbook10/tools.zip
unzip tools.zip -d ~/Desktop/
curl -O https://inteltechniques.com/data/osintbook10/mac-files.zip
unzip mac-files.zip -d ~/Desktop/
mkdir ~/Documents/scripts && mkdir ~/Documents/icons
cd ~/Desktop/mac-files/scripts && cp * ~/Documents/scripts
cd ~/Desktop/mac-files/icons && cp * ~/Documents/icons
cd ~/Desktop && rm -rf mac-files.zip tools.zip ff-template.zip mac-files
```

We now need to install all of the remaining programs which are not available within Brew. The following commands within Terminal execute each installation and configuration requirement. In my experience, macOS does not play well with virtual Python environments. Therefore, we should expect to see some dependency errors which rarely impact functionality. **Because of this, a macOS OSINT build will never be as stable as our Linux VM**. I only present this strategy as an option for non-Linux users, and anyone who replicates this setup should expect occasional failures and troubleshooting. You have been warned, let's start the process.

```
sudo python3 -m ensurepip
sudo pip3 install -U pip
sudo pip3 install instaloader
sudo pip3 install instalooter
sudo pip3 install toutatis
mkdir ~/Downloads/Programs && cd ~/Downloads/Programs
git clone https://github.com/Datalux/Osintgram.git
cd Osintgram && sudo pip3 install -r requirements.txt
sudo pip3 install gallery-dl
cd ~/Downloads/Programs
git clone https://github.com/sherlock-project/sherlock.git
cd sherlock && sudo pip3 install -r requirements.txt
sudo pip3 install socialscan
sudo pip3 install holehe
cd ~/Downloads/Programs
git clone https://github.com/WebBreacher/WhatsMyName.git
cd WhatsMyName/whatsmyname
sudo pip3 install -r requirements.txt
cd ~/Downloads/Programs
git clone https://github.com/p1ngul1n0/blackbird
cd blackbird && sudo pip3 install -r requirements.txt
```

```
sudo pip3 install maigret
cd ~/Downloads/Programs
git clone https://github.com/martinvigo/email2phonenumber.git
cd email2phonenumber && sudo pip3 install -r requirements.txt
pipx ensurepath
pipx install ghunt && pipx ensurepath
cd ~/Downloads/Programs
git clone https://github.com/aboul3la/Sublist3r.git
cd Sublist3r && sudo pip3 install -r requirements.txt
cd ~/Downloads/Programs
git clone https://github.com/s0md3v/Photon.git
cd Photon && sudo pip3 install -r requirements.txt
cd ~/Downloads/Programs
git clone https://github.com/laramies/theHarvester.git
cd theHarvester && sudo pip3 install -r requirements.txt
sudo pip3 install testresources
sudo pip3 install webscreenshot
cd ~/Downloads/Programs
git clone https://github.com/Lazza/Carbon14
cd Carbon14 && sudo pip3 install -r requirements.txt
sudo pip3 install xeuledoc
cd ~/Downloads/Programs
git clone https://github.com/GuidoBartoli/sherloq.git
cd sherloq/gui && sudo pip3 install -r requirements.txt
cd ~/Downloads/Programs
git clone https://github.com/opsdisk/metagoofil.git
cd metagoofil && sudo pip3 install -r requirements.txt
sudo pip3 install bdfr
sudo pip3 install redditsfinder
cd ~/Downloads/Programs
git clone https://github.com/MalloyDelacroix/DownloaderForReddit.git
cd DownloaderForReddit && sudo pip3 install -r requirements.txt
sudo pip3 install waybackpy
sudo pip3 install changedetection.io
sudo pip3 install archivebox
mkdir ~/Documents/archivebox && cd ~/Documents/archivebox
archivebox init
cd ~/Downloads/Programs
git clone https://github.com/smicallef/spiderfoot.git
cd spiderfoot && sudo pip3 install -r requirements.txt
cd ~/Downloads/Programs
git clone https://github.com/lanmaster53/recon-ng.git
cd recon-ng && sudo pip3 install -r REQUIREMENTS
cd ~/Downloads/Programs
git clone https://github.com/AmIJesse/Elasticsearch-Crawler.git
sudo pip3 install h8mail
cd ~/Downloads && h8mail -g
sed -i '' 's/\;leak\-lookup\_pub/leak\-lookup\_pub/g' h8mail_config.ini
```

If you are copying these commands from the "mac.txt" file on my website, you should be able to copy an entire section and paste it all into Terminal at once. Striking enter on your keyboard should execute each command individually. This can be very convenient once you are comfortable with this process, but I believe each line should be submitted manually as you become familiar with the actions. To be safe, restart the machine after these steps. Next, I like to make some graphical adjustments to my Mac OSINT build. First, I embed the same icons used for the Linux VM into the Mac Scripts. The following commands replicate my process. However, the "mac.txt" file in your downloads includes a single command which conducts all of these steps at once.

```
cd ~/Documents/scripts
fileicon set API\ Tool ~/Documents/icons/api.png
fileicon set Archives\ Tool ~/Documents/icons/archives.png
fileicon set Breaches-Leaks\ Tool ~/Documents/icons/breaches-leaks.png
fileicon set Captures\ Tool ~/Documents/icons/captures.png
fileicon set Documents\ Tool ~/Documents/icons/documents.png
fileicon set Domains\ Tool ~/Documents/icons/domains.png
fileicon set Galleries\ Tool ~/Documents/icons/galleries.png
fileicon set Instagram\ Tool ~/Documents/icons/instagram.png
fileicon set Metadata\ Tool ~/Documents/icons/metadata.png
fileicon set Recon-NG ~/Documents/icons/recon-ng.png
fileicon set Reddit\ Tool ~/Documents/icons/reddit.png
fileicon set Spiderfoot ~/Documents/icons/spiderfoot.png
fileicon set OSINT\ Tools ~/Documents/icons/tools.png
fileicon set Updates ~/Documents/icons/updates.png
fileicon set Users-Emails\ Tool ~/Documents/icons/users-emails.png
fileicon set Video\ Download\ Tool ~/Documents/icons/video-download.png
fileicon set Video\ Stream\ Tool ~/Documents/icons/video-stream.png
fileicon set Video\ Utilities ~/Documents/icons/video-utilities.png
```

You should now see your Mac scripts with custom names and icons. However, you may not be able to launch any of them by simply double-clicking the files. This is due to Apples Gatekeeper security rules within previous version of macOS. Since my simple scripts are not registered with Apple and none of them possess security certificates, they may be blocked by default. There are two ways to bypass this. You could double-click a script; acknowledge the block; open your security settings; enter your password; choose the option to allow the script to run; and repeat for each file. This is quite time consuming. I prefer to right-click on each file; select "Open"; then confirm the security exception. You only need to do this once per file. This is a one-time nuisance, but the protection is beneficial for digital security. Ventura users may be able to avoid all of this.

Next, I want to add my custom scripts to the Dock at the bottom of my screen and within my Applications folder. I also want to decrease the size of the icons; remove all standard Apple applications from the Dock; and place my programs in the exact same order as the Linux VM. The command to add the scripts to your Applications folder is as follows.

```
ln -s ~/Documents/scripts/ /Applications/
```

The command to place the scripts within the Dock is too long to place here. It would have taken three pages to display in the book. The following would copy only the Domains Tool script.

```
defaults write com.apple.dock persistent-apps -array-add '<dict>< key>tile-
data</key><dict><key>file-data</key><dict><key>_CFURLString
</key><string>/Applications/scripts/Domains Tool</string> <key>_
CFURLStringType</key><integer>0</integer></dict></dict></dict>'
```

The long commands included within your "mac.txt" file places shortcuts to every program and script present within the Linux VM into your Dock. Look for the section titled "Add Programs To Dock And Modify Size". Warning: It overrides all settings of your current Dock. Finally, execute the "Updates" script in your Dock to finish the installation. This should update the various dependencies required for all applications.

Single Install Command (Mac)

You should now possess a Mac machine which replicates most aspects of the Ubuntu OSINT VM. Figure 6.01 displays the final result with all applications visible. You may be expecting a simple command which will replicate all of these steps as was presented with the Linux VM chapter. If so, you will be disappointed. I exclude a single install command for a few reasons. First, the Linux section assumes that you have created a virtual machine which can be easily deleted and rebuilt. Making these changes to your Mac computer should be very intentional and deliberate. I don't want to offer a script which could make drastic changes to your host computer which could impact other installed applications. Next, replicating the Linux steps is not the same as on a Mac due to security restrictions within the operating system. There are a few steps which require manual execution which cannot be properly replicated within Terminal. As an example, I can easily launch and close Firefox within Terminal on Linux, but it can be difficult to do on a Mac. You must also manually launch the custom programs during first use on some hosts. For these reasons, you should apply these steps manually.

macOS Issues

If you applied the Mac build steps presented here, you may see many errors while executing the steps. The following presents information which you may find beneficial.

- Some installation steps may present an error about existing software. This is mostly due to the presence of software already on your system. There is no concern. Simply go through these steps, ignore any errors, and be sure to run the update script when complete. If your Terminal session asks you to override any files, confirm this option. You might be presented with an option to enter "A" in order to allow overwriting of all files. Either way, allow your system to replace any previous software and you should be caught up to this version of all software, scripts, and tools.
- If you find many of the Terminal applications installed within Brew to be missing, you may have an issue with your "PATH". Execute "`brew doctor`" within Terminal and follow the advice presented on the screen. As an example, immediately after installing Brew, I was prompted to copy and paste three commands into Terminal, executing each. If I had skipped this, many applications would not be seen by the operating system and would not launch. Executing "brew doctor" at any time will remind you of pending tasks which need completed.
- If you find some applications will not open or provide results, it is possible that your macOS operating system's Gatekeeper utility has blocked them. This is usually due to unsigned software from developers who do not want to pay fees for Apple to vouch for their work. I encourage you to visit "System Settings" > "Privacy & Security" > "Security", and click "Allow Anyway" for any trusted applications. As I was writing this, macOS was blocking PhantonJS, which prevented webscreenshot from functioning. After allowing the software, the application performed as desired. This is a constant battle with macOS, and another reason I prefer Linux OSINT virtual machines.
- Newer Mac machines with the Apple M1 processor may have many issues, especially with virtualization. This is due to the compatibility with this chip. As an example, Google Earth Pro currently does not offer an ARM-based macOS version. However, installing Rosetta should resolve most issues.
- Previous editions of this book encouraged readers to install a newer version of Python from Brew. I no longer recommend this. It tends to cause conflicts with the existing default Python application. Since the latest macOS operating system now offers a minimum of Python 3.10, we can rely on it. You can always check the version of your Python with `python3 --version` within Terminal.
- Missing dependencies, outdated programs, file structure issues, authorization permissions, and countless other issues might prevent your desired OSINT application from performing properly. For these and other reasons, I always prefer a Linux virtual machine for my investigations. However, don't

let the lack of Linux within your arsenal prohibit you from attempting these Linux features. Create the OSINT environment best for your needs. I keep the API and Video Downloads scripts within my macOS host for easy access when needed.

Updates

Finally, I want the ability to update all of these programs whenever desired. The following can be entered manually, or you can simply click on the custom "Updates" icon now in your Dock which executes each line presented here. These commands are also available at the bottom of the "mac.txt" file on my website. If (when) things change, I will update the information there, and you can apply it to your own Updates script if desired.

```
brew update && brew upgrade && brew upgrade --greedy
brew autoremove && brew cleanup -s && rm -rf "$(brew --cache)"
brew doctor && brew missing
sudo pip3 install -U pip 2>/dev/null
sudo pip3 install -U changedetection.io 2>/dev/null
sudo pip3 install -U instaloader 2>/dev/null
sudo pip3 install -U toutatis 2>/dev/null
cd ~/Downloads/Programs/Osintgram
git pull https://github.com/Datalux/Osintgram.git
sudo pip3 install -r requirements.txt 2>/dev/null
sudo pip3 install -U gallery-dl 2>/dev/null
cd ~/Downloads/Programs/sherlock
git pull https://github.com/sherlock-project/sherlock.git
sudo pip3 install -r requirements.txt 2>/dev/null
sudo pip3 install -U socialscan 2>/dev/null
sudo pip3 install -U holehe 2>/dev/null
cd ~/Downloads/Programs/WhatsMyName
git pull https://github.com/WebBreacher/WhatsMyName.git
cd whatsmyname && sudo pip3 install -r requirements.txt 2>/dev/null
cd ~/Downloads/Programs/blackbird
git pull https://github.com/p1ngul1n0/blackbird
sudo pip3 install -r requirements.txt 2>/dev/null
cd ~/Downloads/Programs/email2phonenumber
git pull https://github.com/martinvigo/email2phonenumber.git
sudo pip3 install -r requirements.txt 2>/dev/null
pipx upgrade ghunt
cd ~/Downloads/Programs/Sublist3r
git pull https://github.com/aboul3la/Sublist3r.git
sudo pip3 install -r requirements.txt 2>/dev/null
cd ~/Downloads/Programs/Photon
git pull https://github.com/s0md3v/Photon.git
sudo pip3 install -r requirements.txt 2>/dev/null
cd ~/Downloads/Programs/theHarvester
git pull https://github.com/laramies/theHarvester.git
sudo pip3 install -r requirements.txt 2>/dev/null
cd ~/Downloads/Programs/Carbon14
git pull https://github.com/Lazza/Carbon14
sudo pip3 install -r requirements.txt 2>/dev/null
sudo pip3 install -U xeuledoc 2>/dev/null
cd ~/Downloads/Programs/sherloq
git pull https://github.com/GuidoBartoli/sherloq.git
cd gui && sudo pip3 install -r requirements.txt 2>/dev/null
cd ~/Downloads/Programs/metagoofil
git pull https://github.com/opsdisk/metagoofil.git
sudo pip3 install -r requirements.txt 2>/dev/null
sudo pip3 install -U bdfr 2>/dev/null
```

```
sudo pip3 install -U redditsfinder 2>/dev/null
cd ~/Downloads/Programs/DownloaderForReddit
git pull https://github.com/MalloyDelacroix/DownloaderForReddit.git
sudo pip3 install -r requirements.txt 2>/dev/null
sudo pip3 install -U waybackpy 2>/dev/null
sudo pip3 install -U changedetection.io 2>/dev/null
sudo pip3 install -U archivebox
cd ~/Downloads/Programs/spiderfoot
git pull https://github.com/smicallef/spiderfoot.git
sudo pip3 install -r requirements.txt 2>/dev/null
cd ~/Downloads/Programs/recon-ng
git pull https://github.com/lanmaster53/recon-ng.git
sudo pip3 install -r REQUIREMENTS 2>/dev/null
cd ~/Downloads/Programs/Elasticsearch-Crawler
git pull https://github.com/AmIJesse/Elasticsearch-Crawler.git
sudo pip3 install -U nested-lookup 2>/dev/null
sudo pip3 install -U h8mail 2>/dev/null
```

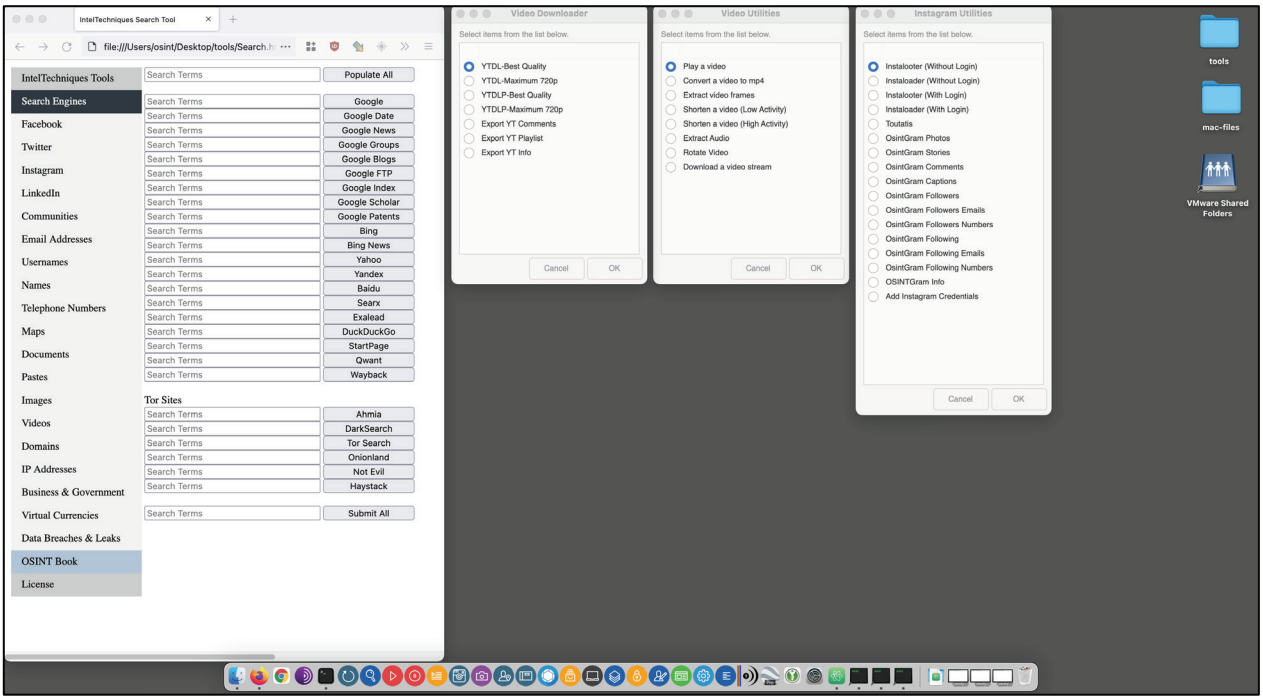


Figure 6.01: A final macOS OSINT machine.

Windows OSINT Host

When I first started thinking about creating a Windows OSINT machine, I considered using the Windows Subsystem for Linux (WSL). This feature allows native use of Linux commands and applications within a virtual space directly in Windows. However, this presented many issues. Some versions of Windows rely on WSL 1 while others support WSL 2. The file storage within WSL and execution of programs can present complications, and I eventually abandoned the idea of using WSL for our needs. I then focused on Power Shell. This command line utility is very similar to the native Windows Command Prompt but with added features. I found it to be overkill for our needs. Therefore, I settled on traditional batch files executed through Command Prompt for the previous edition.

I have been creating batch files since the 90's. These small text files are very similar to the Bash scripts which we created previously within Linux. I believe Windows batch files are actually simpler to create and more straight-forward in design. The menus are not as pretty as what we saw with Linux and Mac. However, the function will be identical. Let's take a look at the difference. Figure 6.02 (left) displays the Video Download Tool within the Linux VM. Figure 6.02 (right) displays the Video Download Tool available within the Windows OSINT machine. The left is a pop-up menu while the right is a script within a traditional Command Prompt display. Some may prefer the simplicity of the Windows option.

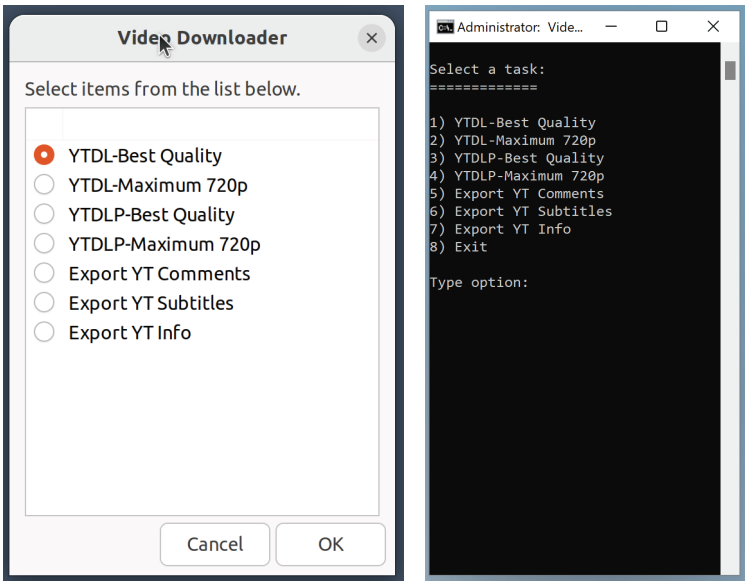


Figure 6.02: A comparison of Linux and Windows script execution.

Next, we should compare the scripts. The previous Linux ".sh" files were explained in Chapter Four and were used during the Mac option presented within this chapter. The ".bat" file included within the "scripts" folder at **https://inteltechniques.com/data/osintbook10/windows-files.zip** is an example of a pre-configured script to replicate the Video Download Tool within Windows. I include this script as a lesson, which could be applied toward other Terminal-based applications which you desire on your machine. However, we have a problem. If you thought making Python applications work within macOS was a struggle, Windows is much worse. While you **can** install Python and use Pip to configure many of the programs previously discussed, it will result in many errors and conflicts. The Windows OSINT machine I presented within the previous edition was constantly breaking and did not perform well, even when things "worked". I feel it is irresponsible to continue forcing all Linux-based OSINT apps to work within Windows when we have the better option of virtual machines. However, that does not mean that Windows cannot have a strong presence within our OSINT strategies. The following pages present many options, but know that we can never truly replicate our Linux build within Windows.

Let's take a look at the batch file previously mentioned. The following displays the Video Download Tool titled "youtubedl.bat" available within your downloaded files.

```
@echo off
title Video Download Tool
:home
cls
echo.
echo Select a task:
echo =============
echo.
echo 1) YTDL-Best Quality
echo 2) YTDL-Maximum 720p
echo 3) YTDLP-Best Quality
echo 4) YTDLP-Maximum 720p
echo 5) Export YT Comments
echo 6) Export YT Subtitles
echo 7) Export YT Info
echo 8) Exit
echo.
set /p web=Type option:
if "%web%"=="1" goto 1
if "%web%"=="2" goto 2
if "%web%"=="3" goto 3
if "%web%"=="4" goto 4
if "%web%"=="5" goto 5
if "%web%"=="6" goto 6
if "%web%"=="7" goto 7
if "%web%"=="8" exit

:1
set /p url=Target URL:
mkdir %userprofile%\Documents\Videos
cd %userprofile%\Documents\Videos
youtube-dl %url% -o "%%(title)s.%%(ext)s" -i --all-subs
start "" %userprofile%\Documents\Videos
goto home

:2
set /p url=Target URL:
mkdir %userprofile%\Documents\Videos
cd %userprofile%\Documents\Videos
youtube-dl %url% -o "720-%%(title)s.%%(ext)s" -i -f
"bestvideo[height<=720]+bestaudio" --all-subs
start "" %userprofile%\Documents\Videos
goto home

:3
set /p url=Target URL:
mkdir %userprofile%\Documents\Videos
cd %userprofile%\Documents\Videos
yt-dlp %url% -o "%%(title)s.%%(ext)s" -i --all-subs
start "" %userprofile%\Documents\Videos
goto home

:4
set /p url=Target URL:
mkdir %userprofile%\Documents\Videos
```

```
cd %userprofile%\Documents\Videos
yt-dlp %url% -o "720-%%(title)s.%%(ext)s" -i -f
"bestvideo[height<=720]+bestaudio" --all-subs
start "" %userprofile%\Documents\Videos
goto home

:5
set /p url=Target URL:
mkdir %userprofile%\Documents\Videos
cd %userprofile%\Documents\Videos
yt-dlp %url% --write-comments --skip-download
start "" %userprofile%\Documents\Videos
goto home

:6
set /p url=Target URL:
mkdir %userprofile%\Documents\Videos
cd %userprofile%\Documents\Videos
youtube-dl --all-subs --skip-download %url%
start "" %userprofile%\Documents\Videos
goto home

:7
set /p url=Target URL:
mkdir %userprofile%\Documents\Videos
cd %userprofile%\Documents\Videos
yt-dlp %url% --write-info-json --skip-download
start "" %userprofile%\Documents\Videos
goto home
```

Next, let's break down these new commands.

- `@echo off:` This disables commands from being displayed within the Command Prompt menu.
- `title Video Download Tool:` This displays a title of the menu prompt.
- `:home:` This identifies the following text as the "home" screen.
- `cls:` This clears all text from the Command Prompt window.
- `echo Select a task:` This displays any text after "echo", such as "Select a task".
- `echo 1):` This displays the menu visible within the Command Prompt window.
- `set /p web=Type option:` This provides an option to accept user input. The menu displays "Type Option" and waits for a response, such as "1".
- `if "%web%"=="1" goto 1:` This option collects the input from the previous command and navigates the user accordingly. If you had entered "1", the menu would take you to the commands under ":1".
- `:1:` This identifies a set of commands based on the previous selection.
- `set /p url=Target:` This provides an option to accept user input.
- `cd %userprofile%\Downloads:` This changes the directory in order to launch a specific software tool or store data to a folder.
- `youtube-dl:` This executes the command.
- `start "" %userprofile%\:` This opens a folder.
- `goto home:` This instructs the menu to return to the home screen in order to allow an additional query.

Now, let's start building a Windows OSINT host. Ideally, you will be creating this machine from a new Windows installation, but it should also work with existing builds. You may want to practice with a Windows VM first in order to ensure you want these changes applied to your current machine. The following steps generate numerous modifications to your Windows operating system and should not be executed without serious consideration.

During the Mac setup, we used Brew as a package manager for the basics. For Windows, we will use Chocolatey. It can be installed with the following steps. Every command in this section, including any updates, can be found at **https://inteltechniques.com/osintbook10/windows.txt**.

- Click the Windows menu button (lower-left) and type "cmd".
- Right-click on Command Prompt and select "Run as Administrator".
- Enter the following command into Command Prompt.
- `@"%SystemRoot%\System32\WindowsPowerShell\v1.0\powershell.exe" -NoProfile -InputFormat None -ExecutionPolicy Bypass -Command " [System.Net.ServicePointManager]::SecurityProtocol = 3072; iex ((New-Object System.Net.WebClient).DownloadString ('https://chocolatey.org/install.ps1'))" && SET "PATH=%PATH%;%ALLUSERSPROFILE%\chocolatey\bin"`

Next, we can install our basic applications, such as Firefox, Chrome, and others with the following commands. You could eliminate any apps which are undesired, but this demonstration will take advantage of several programs. Many of these should appear familiar by now.

```
choco install -y curl
choco install -y ffmpeg
choco install -y firefox
choco install -y git
choco install -y googlechrome
choco install -y googleearthpro
choco install -y httrack
choco install -y mediainfo
choco install -y python3
choco install -y ripgrep
choco install -y sed
choco install -y streamlink
choco install -y tor-browser
choco install -y unzip
choco install -y vlc
choco install -y wget
choco install -y youtube-dl
choco install -y yt-dlp
```

During installation, you might be prompted to accept all defaults. Type "A" and strike enter if this occurs. **You must reboot after the previous step is complete!** Next, we can configure Firefox as described in Chapter Three with the following steps. Open Firefox, close it, and enter the following in Command Prompt.

```
cd %userprofile%\Downloads\
curl -O https://inteltechniques.com/data/osintbook10/ff-template.zip
unzip ff-template.zip
cd %USERPROFILE%\AppData\Roaming\Mozilla\Firefox\Profiles\
*.default-release
xcopy /Y /E %userprofile%\Downloads\ff-template\*
```

Next, we can download and configure custom Windows scripts, shortcuts, icons, and tools with the following commands within Command Prompt.

```
cd %userprofile%\Desktop
curl -O https://inteltechniques.com/data/osintbook10/windows-files.zip
curl -O https://inteltechniques.com/data/osintbook10/tools.zip
unzip tools.zip -d %userprofile%\Desktop
unzip windows-files.zip -d %userprofile%\Documents
del windows-files.zip tools.zip
```

You can now drag and drop the files within your Documents\windows-files\shortcuts folder anywhere you like, including another folder or the Desktop. You can drag and drop each for desired arrangement. These are shortcuts which launch the batch file included within your download. I used shortcuts because I can customize them with specific icons and folder paths to match our needs. For the same reasons cited within the Mac section, I do not offer a single installation command for Windows. Please conduct all steps manually.

Windows Updates

You will need to keep all of these programs updated often. You can either launch the shortcut titled "Updates" or enter the following within Command Prompt as Administrator. The script launches with administrative privileges by default. This command is also available at the bottom of the "windows.txt" file on my website. If (when) things change, I will update the information there, and you can apply it to your script.

```
choco upgrade all -y
```

Windows Issues

You will likely encounter undesired issues within your own Windows OSINT build which are not present within the Linux VM. Most of these will be related to security or missing dependencies. You may receive a warning stating the operating system "prevented an unrecognized app from running" while executing the batch files. This is because these simple scripts are not registered with Microsoft. Clicking "Run Anyway" the first time each script is executed should resolve the issue.

Summary

macOS: I want to stress again that a macOS OSINT build will never be the same as a Linux machine. While macOS plays well with most Linux applications, it is simply not Linux. If there are specific applications explained within the Linux chapter which you would like to replicate on macOS, this build is a great start. It may allow you to focus on making those applications work consistently for you. You should not expect the entire macOS OSINT build to function reliably. Some things will break, and the developers of these programs will likely have no interest in making them work within macOS. However, these steps allow us to experiment with the possibilities. The macOS build should offer approximately 75% of the overall functionality of the Linux machine.

Windows: The Windows build is designed for convenience and experimentation. While the macOS setup is fairly similar to our OSINT Linux machine, the Windows version is quite limited. However, it should reliably offer a custom OSINT Firefox option, additional browsers, and video download utilities. Expect occasional frustration, troubleshooting, and instability with both macOS and Windows, especially in regard to Python programs.

CHAPTER SEVEN
ANDROID VIRTUALIZATION

For several years, online researchers have been navigating through various social networking websites for information about individuals. Whether it was older sites such as Friendster and Myspace, or current networks such as Twitter and Facebook, we have always flocked to our web browsers to begin extracting data. Times have changed. Today, an entire generation of social network users rarely touch a traditional computer. They operate completely from a cellular telephone or tablet. Many of the networks through which individuals engage will only operate on a mobile device. Services such as Snapchat, Tinder, and TikTok do not allow a user to access content from a traditional web browser. As this shift occurs, investigators must transition with it. Our preparation is not complete until we have disposable Android environments in place.

This chapter will focus on the huge amount of information available through mobile platforms that is not accessible through a web browser. I will explain a method of emulating a portable device within a traditional computer. Before we dive into the nuts and bolts of making things work, we should discuss why this is the way to go. In my investigations, documentation is my primary reason for launching a simulated mobile device within my computer operating system. If I conducted my investigation on an actual smartphone, documenting my findings can be difficult. Mobile screen captures only cover a small amount of visible content. Extracting any captured images can be a hassle. Referencing my findings within a final report can become very tedious. When using Android emulation within my traditional computer, I can easily create numerous screen captures, record a video of my entire investigation, and paste my results directly into the report.

Privacy and security are also important reasons to consider emulation versus directly investigating from a portable device. I have seen many law enforcement investigators conduct a search or use an app directly from their personal or work phones. This opens that device to scrutiny and discovery. An attorney could rightfully request a copy of the investigator's phone in order to conduct an independent forensic analysis. That would make most people nervous. Additionally, if I encounter malicious software or a virus from my portable device, it could affect all future investigations using that hardware. Emulation will remedy both of these situations.

The idea of Android emulation (or virtualization) is to recreate the mobile operating experience within an application on your computer. This application will execute in the same manner that your web browser, word processor, or email client would open. It will have the exact same appearance as if you were staring at a telephone or tablet. Any actions that you take within this emulated device will not affect anything else on your computer. Think of it as an encapsulated box, and nothing comes in or gets out, very similar to our Linux VM previously explained. A great feature of emulation is that you can create unlimited virtual devices. You could have one for every investigation in order to prevent any contamination.

Some readers will question why I chose to explain Android emulation instead of iPhone. The most obvious reason is the number of options. I will explain software solutions for recreating the Android environment on your computer. An iPhone simulator will only function on Apple computers and has very limited features. The Android techniques will work on any major operating system. Additionally, we can create Android virtual machines that possess all original functionality. An iPhone simulator will not connect to most applications and features, and provides almost no value to the OSINT investigator.

The following pages differ greatly from previous editions of this book. Even if you have previous Android machines functioning for your investigations, please read through this chapter. I believe you will find alternative ways of Android virtualization which could benefit your investigations.

The previous editions of this book focused heavily on Genymotion and VirtualBox for Android virtualization. These cross-platform applications allowed us to easily create virtual Android environments which appeared as magical mobile devices on our screens. **I no longer recommend Genymotion or VirtualBox as our best options.** I have several reasons, as outlined in the following.

- Genymotion requires a user account in order to access your Android virtual machine. This requires an email address and includes tracking of your activity.
- Genymotion has limited their free option for personal use. Relying on their software during professional investigations may violate their terms of service.
- Genymotion continues to make the free version difficult to locate and download.
- Genymotion relies on VirtualBox for all virtualization. This introduces additional layers of complexity, troubleshooting, and limitations.
- VirtualBox does not currently support Apple M1 or later processors, and runs poorly on Apple Intel machines. This prevents Genymotion from running on newer Apple hardware.
- Both Genymotion and VirtualBox only support x86 virtual machines, which prevents native ARM virtualization for those with machines which support ARM builds.
- Both options are inferior to native Android virtualization.
- Many mobile applications specifically block Genymotion, VirtualBox, or other x86 builds.

Instead, we will build our Android devices without the need for an account through Genymotion or Google. We can stay fairly anonymous and rely on cross-platform software provided by the creator of Android.

Android Studio (developer.android.com/studio)

This software was created and is maintained by Google, the maker of Android. It is designed for Android developers who need to create and test mobile applications within a desktop environment before applying them to actual mobile devices. While this software has way more features than we need as online investigators, I believe it has the best Android virtualization option. It can be installed with the following commands (assuming macOS users have Brew installed and Windows users configured Chocolatey, as previously explained).

- Linux: `sudo snap install android-studio`
- macOS: `brew install --cask android-studio`
- Windows: `choco install androidstudio`

Once you have it installed, you must launch and configure the applications. This will appear slightly different on each operating system and within each major software revision. However, the following steps which I took on my macOS OSINT machine should assist.

- Launch Android Studio.
- Do not import any settings but allow any default downloads.
- Choose "Don't Send" for analytics, then click "Next" in the Welcome Wizard.
- Choose the "Standard" option and click "Next".
- Select your desired appearance, click "Next", accept both agreements, and click "Finish".

Android Studio will download numerous items, which may take some time. If you receive any download errors, choose the "Retry" option. Once everything has been successfully downloaded, click "Finish". This should launch the new application window, and the following should be performed.

- Click "New Project", then "Next" to confirm "Empty Activity".
- Provide a name for your application (I chose "OSINT"), and click "Finish".
- Allow all additional default downloads and click "Finish".

The application will navigate to the main developer panel, and you should see several downloads continue in the lower right. Allow everything to complete before proceeding. You should then see a vertical bar on the right titled "Device Manager". Clicking this should present a virtual device, which may be titled similar to "Pixel_3a". This will likely contain the latest version of Android, but we should also explore other options. Click the "Create device" button, as seen in Figure 7.01. This allows you to select from many virtual Android options. I chose to add a Pixel 6 Pro to my available devices, and selected the most recent API level, which can also be seen in Figure 7.01.

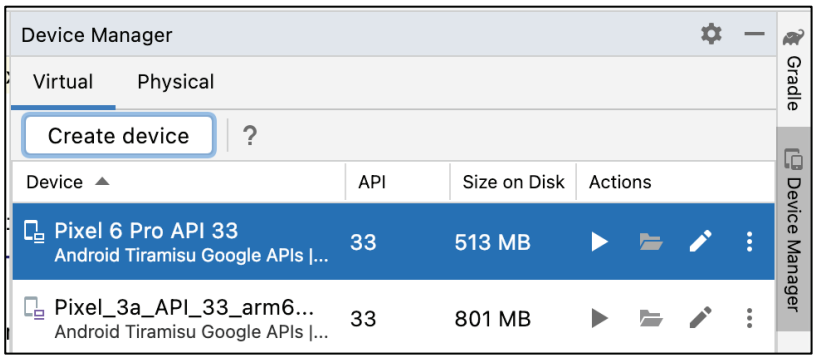


Figure 7.01: The Android Studio Device Manager.

Before we launch our first device, we should consider an important adjustment. Since Android Studio is targeted toward developers, virtual Android devices launch within the developer portal. I prefer my devices to launch within their own window, so I make the following configuration change in the "Preferences" or "Settings" menu within Android Studio.

- Click on "Tools" then "Emulator".
- Disable "Launch in a tool window" and click "OK".

Now choose your desired device and click the arrow button to launch. You should see a device similar to that in Figure 7.02 (left). You are now running a virtual version of Android within your desktop environment. Next, let's configure the device for future OSINT usage.

- Remove any desired applications from the home screen by clicking, holding, and dragging up.
- Launch the Chrome browser and deny any features.
- Search for F-Droid and visit the F-Droid site.
- Click "Download F-Droid" and confirm request.
- When finished, click the "Open" option and then "Settings" when prompted.
- Enable "Allow from this source" and click "Install" then "Open". Allow notifications.
- Within F-Droid, search for "Aurora Store" and select it.
- Click "Install" then "Settings"; enable "Allow from this source"; and click "Install".
- Click the circle "Home" button within the floating sidebar.

You should now see two additional applications within your home screen. Let's open Aurora Store and make the following configuration changes.

- When prompted, select the Terms of Service box and click "Accept".
- Click "Next" four times to navigate through the settings.
- Click "Grant" next to each option and "Allow" when prompted.
- Enable any access menu and use the back button until all options show "Granted".
- Click "Finish" then "Anonymous".

Aurora Store should now connect you to their proxy for the Google Play Store. You should be able to download any applications as you would on any other device. If you are on a VPN, you may be blocked from this connection. During my own configuration, I had to connect to a dedicated VPN IP address to complete the connection. Once you are within Aurora, install an application. I searched "Facebook"; chose the first option; clicked "Install"; then confirmed my choice. I can now open the Facebook mobile application within a secure virtual environment from my desktop computer.

Let's digest these steps. You installed F-Droid which is an open-source package installer for Android. It allowed us to install Aurora Store which is an anonymous replacement for the Google Play Store. Through Aurora Store, you installed Facebook to ensure the ability to add apps. You authorized all applications to install additional apps on your device, which should only be a one-time requirement. You can now launch Aurora Store and install practically any app desired. During this writing, I installed the following apps and moved them to my home screen, as visible in Figure 7.02 (right).

Facebook	Messenger	WhatsApp	Instagram	Twitter	Snapchat	Kik
TikTok	TextNow	Truecaller	Bumble	Telegram	YouTube	Tinder

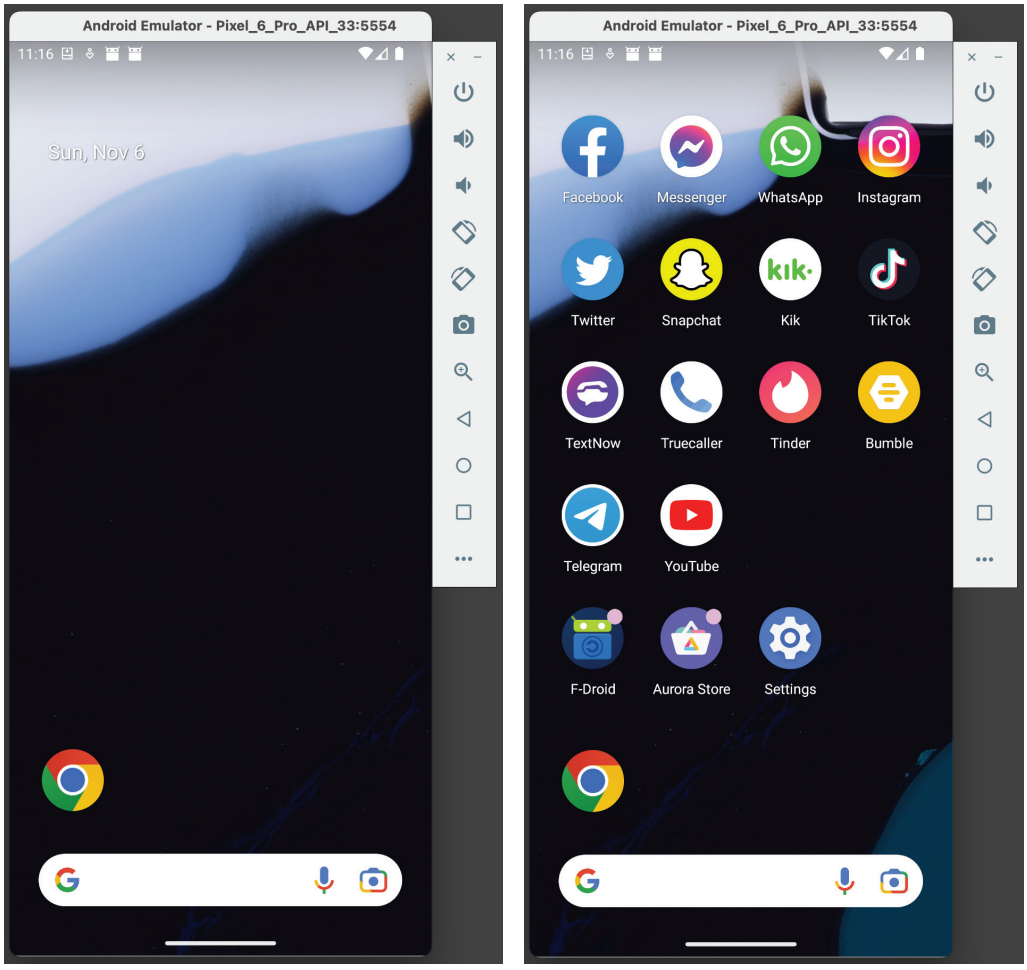


Figure 7.02: Virtual device screens within Android Studio.

Navigating these screens is identical to a physical device. However, controlling them from a mouse and keyboard may take some practice. You can click the circle on the floating bar to return to your home screen at any time. The square button allows you to browse through open apps and swipe them up to close them. The left arrow

takes you back one step within an app. Always click and hold an app before moving it, and dragging the screen right or left navigates between home spaces.

Since we are working with a virtual device on a computer screen, there are a few nuances which should be discussed. By default, internet access is gained through your host computer. If you ever find that applications seem to stop communicating, check and be sure that "Wi-Fi" is enabled. I have experienced unexplained internet outages which were corrected by re-enabling Wi-Fi under "Settings". The easiest way to turn the device off is to click the power button on the top of the floating bar. Go ahead and turn the device off so we can explore cloning options. Once the screen is black, click the "X" in the floating bar and return to the Android Studio application.

Click the three dots next to your device, as seen in Figure 7.01. Click "Duplicate" and call the device "OSINT Primary". Click "Finish" and launch the device. You can now make any changes to this virtual Android device without impacting your original OSINT copy. You can create a new Android device for every investigation, as explained within Chapter Two about virtual machines. Once you have a device with all desired apps, you can clone it without the need to replicate your work. You can also delete devices from within this menu. Clicking "Show on disk" allows you to locate the machine on your drive and archive the folder for later use.

Relaunch a device and click the three horizontal dots at the bottom of the floating bar. This opens the extended controls for your Android device. The Location tab allows you set a location which will be announced to any application which wants to know it. This allows you to spoof your location to apps which only provide nearby content, such as dating apps. Figure 7.03 (left) displays this menu while Figure 7.03 (right) shows the Maps app as it confirmed my spoofed location.

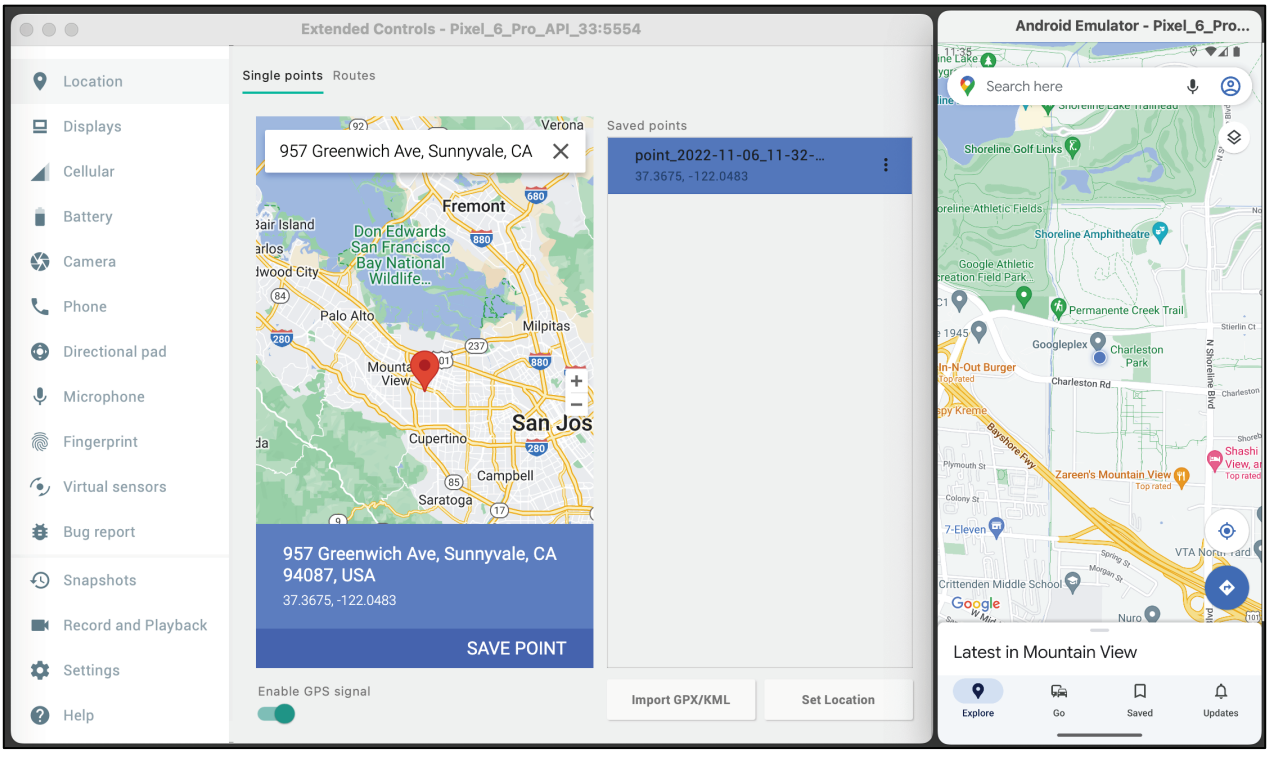


Figure 7.03 Location settings and confirmation.

Once you have your Android device configured exactly as desired, you can make a clone via the Android Studio application. While powered down, select your device from the Device Manager; click the three dots to expand the menu, and choose "Duplicate". You can also delete any unneeded devices on this screen.

Finally, once you have your desired Android virtual machine configured, you may want to launch it without the need to open Android Studio; select the device; execute the environment, power-on the operating system, and close the main application. Fortunately, we can do all of this within Terminal, and launch ONLY the Android devices without any other clutter. The following commands are based on my macOS M1 OSINT host, but it should be easy to modify the paths and executable names using the overall lessons from Chapter Six. The following switch to the Android Studio directory and list all configured Android machines.

```
cd ~/Library/Android/sdk/emulator/
./emulator -list-avds
```

The result appears similar to the following.

```
Pixel_6_Pro_API_33
```

We can now launch that machine with the following quick-boot command, which will also save any changes.

```
./emulator -avd Pixel_6_Pro_API_33
```

The following will cold-boot the machine as if it were fully rebooted from an off state, and save changes.

```
./emulator -avd Pixel_6_Pro_API_33 -no-snapshot-load
```

We can modify this command to launch the machine, but not save any changes, with the following command.

```
./emulator -avd Pixel_6_Pro_API_33 -no-snapshot-save
```

Finally, the following command will launch the device as it was on first boot, erasing all custom data.

```
./emulator -avd Pixel_6_Pro_API_33 -wipe-data
```

What do I do? I created a script with these desired options and devices. However, I mostly execute machines with "`./emulator -avd Pixel_6_Pro_API_33`". This preserves my current state and loads the quickest. When I want to erase a machine and start over, I can do so without the GUI interface, or researching any commands. The text from my script is located at the following URL on my website.

https://inteltechniques.com/data/osintbook10/android.txt

Use the lessons from Chapters Four and Six to make this script executable within Linux or macOS. Use the batch file demonstration from Chapter Six to make this data work for Windows. Launching this script presents the following, and will only work if you also created a Pixel 6 Pro with API 33.

```
1) List Devices
2) Launch Pixel 6 (Quick-Changes Saved)
3) Launch Pixel 6 (Quick-Changes Not Saved)
4) Launch Pixel 6 (Cold-Changes Saved)
5) Launch and Wipe Pixel 6
Selection:
```

After you have your desired location configured and you have confirmed accuracy, you can start to put this feature to work. The following tutorials explain how I use various mobile applications within my OSINT investigations, especially location-aware applications. This could never be a complete documentation. Any time you encounter a target using a service which possesses a mobile application, you should consider installing that app to see what further details you can obtain about the target's profile.

Facebook/Messenger/Instagram: The Facebook app on Android will appear similar to a compressed view of a standard profile page. The benefit of the mobile app is the ability to check into places. After launching the app and logging in for the first time, allow Facebook to access your location (which is spoofed). When you click the "Check In" option, Facebook will present businesses near your current spoofed location. If you choose a location, and create a post on your timeline, Facebook will verify that you were there. I have used this when I needed to portray that I was somewhere I was not. This method can help you establish credibility within your pseudo profile. I also once helped a victim confuse her ex-husband with this technique. I posted from her Facebook account leaving my spoofed location enabled. After wasting his time going to random places trying to find her, he began doubting the information that he uncovered about her whereabouts.

WhatsApp: WhatsApp Messenger is an instant messaging app for smartphones that operates under a subscription business model. In addition to text messaging, WhatsApp can be used to send images, videos, and audio media messages to other users. Locations can also be shared through the use of integrated mapping features. You will need to create an account and provide a telephone number for verification. This number can be a cellular, landline, or VOIP number. After you have an account, you can communicate directly with any target using the service. I have found that several of my targets refuse to converse over traditional text messaging, but freely text over WhatsApp. If you conduct any covert operations, you should have this set up ahead of time.

Twitter: The first time that you use Twitter within your Android environment, you might be asked if you want to share your location. While I usually discourage this type of activity, sharing your spoofed location can have many benefits. Similar to Facebook, you can make yourself appear to be somewhere which you are not. You may want to confuse your target. If you know that he or she will be monitoring your social networks using the techniques in later chapters, this method should throw them off and be misleading.

Snapchat: Much of the content within Snapchat is only visible within a mobile environment. Having access to the mobile app is vital. If you simply want to search public posts, we will tackle that via a traditional browser later in the book.

TikTok: While I will explain investigation techniques for this network much later in the book, having the mobile app ready is important. The TikTok website does not currently allow native keyword search, but the mobile app does. Preparation now will provide great benefit later.

TextNow: If you conduct online investigations and communicate with a suspect, it is very possible that you may be asked to send or receive a standard SMS text message. Since your virtual device does not possess a cellular connection, and it is not assigned a telephone number, there are no native opportunities for this activity. However, you can install TextNow, which allows you to send and receive SMS text messages. With this setup, you can conduct all of your communications through the virtual device, and preserve the evidence within a single archive.

Truecaller: A later chapter explains reverse caller ID services and how they can identify subscriber information associated with telephone numbers. There are several additional services that only support mobile use. Truecaller is a powerful service which allows search of unlimited cellular and landline numbers in order to identify the owners. Other options include Mr. Number and Showcaller.

Tinder: This dating app relies on your location in order to recommend people in your area that want to "hook up". It can use your Facebook account associated with your device or a VOIP telephone number for the login credentials. The preferences menu will allow you to specify the gender, age range, and distance of the targeted individuals. Most people use this to identify members of their sexual preference within one mile of their current location. The users can then chat within the app. I have used this to identify whether a target was at home or another location. During one investigation, I set my GPS in my Android emulator to my target's residence. I then searched for men his age within one mile and confirmed he was at home. If I did not get his profile as a result, I could change my GPS to his work address or favorite bar. When I received his profile in the results, I knew that he was near the spoofed location. I could do all of this from anywhere in the world.

Contact Exploitation: Many applications allow (and encourage) you to locate your friends' profiles within the service by their email address or telephone number. This is one of the most reliable ways which apps can keep you within their ecosystem. As investigators, we can use this to our advantage. Apps such as CashApp allow you to enter your friends' telephone numbers and it will display their complete profile. You could replicate this with any online target. Sometimes, the only way I can identify a telephone number's owner is to search it via mobile applications. I have found that adding my unknown target's cellular telephone number to the Android phone's address book will often obtain the following information relative to the target.

- Associated Facebook accounts (name) from the "Find Friends" feature.
- Google Play purchases and reviews (interests) from the Google Play Store.
- Associated Twitter accounts (name) from the "Find Friends" feature.
- WhatsApp usernames and numbers (contact) registered to the cell number.

Basically, entering a target's phone numbers and email addresses into your address book on an Android emulator forces many apps to believe that you are friends with the person. It overrides many authority protocols that would otherwise block you from seeing the connection from the real details to the connected profiles. Figure 7.04 displays a redacted result of one attempt. I launched "Contacts" from within the Android applications and added a cellular number of a target with any name desired. I then launched Facebook and clicked the "Find Friends" option. Facebook immediately identified an account associated with the number entered.

Figure 7.04: A Facebook "friend" disclosure after adding a cellular number to Contacts.

Let's consider another example using the popular secure messaging program Signal. When I downloaded the Signal app, it wanted me to register a telephone number. I chose a Google Voice number and configured the app. I then added my target's cellular number into my Android contact list and asked Signal to search for friends. Signal immediately confirmed that my target was active on Signal. This alone is valuable in regard to behavior, but not very helpful to establish identity. If I launch a new window to send a message to the number, even if I do not send the data, I may see a name associated with the account. This would need to be a deliberate act by the target, but this behavior is common.

Secure Communications Apps: If you plan to communicate directly with targets of your investigation, you should be familiar with the popular secure communication preferences. Asking a suspect of a sex trafficking investigation to text you via cellular telephone number will not be well received. If you possess a secure Proton Mail email address or Wire encrypted communications username, your request may be honored. Possessing these apps within your Android environment allows you to contain the evidence within a VM and protect your host machine. You could also possess multiple accounts through these providers and log in only after cloning your machine, as explained later.

There are many other beneficial apps. Now that you have an idea of how to integrate mobile applications into your investigations, you can apply the same techniques to the next future wave of popular apps. Many social network apps have no association with location. This content can still have value to an investigation. Some apps, such as Kik, only function within a portable device. You cannot load a web browser on a traditional computer and participate with these networks. However, you can access them from within your Android virtual machine. The goal within this chapter is simply preparation. While we have not yet discussed specific investigation methods within these services, having a virtual Android device ready now will ease the explanations later.

Genymotion (genymotion.com)

I previously mentioned that Android devices created directly within Android Studio are preferred over those provided through third parties. I stand by those statements, but I also respect readers who may prefer other options. Genymotion may have undesired issues in regard to privacy and licensing, but the product can also be beneficial to those who have issues with the previous example. Many readers report that Genymotion Android VMs load faster, feel smoother, and seem more intuitive on Windows machines, but I have yet to experience that. This application-based Android solution is extremely easy to use. It works with Windows, macOS, and Linux operating systems, but not devices with newer Apple processors.

First, you must create a free account online at genymotion.com. This can be all alias information, and the login will be required in order to fully use the application. After you have created the account and successfully logged in to the site, navigate to genymotion.com/download and click on the "Download Genymotion Personal Edition" link. This presents the standard download page for Windows, Mac, and Linux. If prompted, choose the version without VirtualBox, as you should already have that program installed. Executing the download and accepting all default installation options will install all of the required files. When the setup process has completed, you will have a new icon on your desktop titled Genymotion. This entire process should occur on your HOST operating system, and not within a virtual machine.

Execute this application and note that an Android virtual machine may already be pre-installed and ready for launch. Instead of accepting this default option, consider creating your own machine in order to learn the process for future investigations. I recommend deleting this machine by clicking the menu icon to the right of the device and choosing "Delete". Perform the following instructions in order to create your first custom Android devices.

- In the left menu, expand the "Android API" menu and select the highest number. My option was 10.0 at the time of this writing. On the right, choose the device. I chose "Google Pixel XL" since I have a high-resolution screen, and then clicked "Add custom device". You may want to choose a device with a smaller screen for your hardware.
- Rename this device similar to Android 10.0 Original. Change the "Android Version" to the highest option and click "Install". This will download and configure the device for immediate use, and can take several minutes.
- Launch the new device by double-clicking the new machine present in the Genymotion software. The machine will load in a new window which should appear similar to the screen of an Android telephone. Click "OK" to any feature notifications. Figure 7.05 (left) displays the default view of my home screen.
- Navigate within the Android emulator by single-clicking on icons and using the "Back" icon in the lower left that appears similar to a left facing arrow.
- Consider the following customizations to improve the look and feel of the device. Figure 7.05 (right) displays the view of the home screen after these configurations.
- Drag any app icons up and drop them in the "Remove" option.
- Click and hold the bottom of the screen and drag up to view installed applications.
- Drag the Settings icon to your home screen and open the app.
- Choose "Display", then "Sleep", and select "30 Minutes".
- Choose "Security", then "Screen Lock", and choose "None".
- Press and hold the main window, select "Wallpaper", and change if desired.
- Shut down the device and open VirtualBox.
- Similar to the VM settings, change the Video Memory to the maximum.
- Change the Memory size to half of the system resources.
- Relaunch your device from within the Genymotion application.

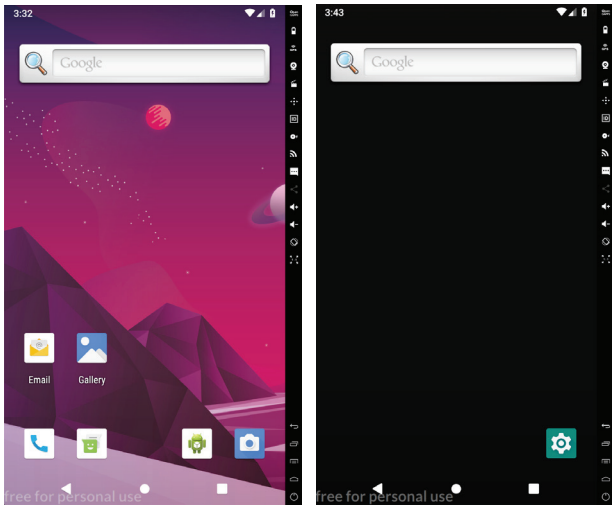


Figure 7.05: A default Android screen (left) and the custom version free of clutter (right).

You should now have a functioning replica of a standard Android device. However, you are missing several features. The biggest void is the absence of key applications such as Google Play and Gmail. Without core Google services, you cannot download apps to your device as part of your investigation tools. This has been the biggest hurdle with emulation. Consequently, there is finally an official fix, and an alternative option for advanced users. First, let's try the easy way by using the Genymotion built-in Google features.

- While inside our virtual Android device, click the "Open GAPPS" icon in the upper right corner. Accept the agreement and allow Google Apps to install. Select the option to restart the devices.
- Your browser should open to https://opengapps.org/?source=genymotion. Select "ARM64", the version of the device that you created (10.0.0), and "Stock". Click the red download option in the lower right and save the large file to your Desktop. Do NOT open the downloaded zip file.
- Drag and drop the downloaded zip file into your running Android device. Accept any warnings. You may receive errors. When complete, close and restart the device.

You should now have the Google Play Store in your applications menu. Launching it should prompt you to connect to an existing or new Google account. Consider using an anonymous account that is not used for anything else. I do not recommend creating a new account from within this virtual machine because Google will likely demand a cellular telephone number for verification. I prefer to create Google accounts from a traditional computer before connecting to the virtual Android device. After syncing with an active Google account on your new device, you should now be able to enter the Google Play Store. You should also now see all core Google services in your applications menu.

You can now install any apps within the Play Store. If any apps refuse to install because of an incompatible device, you could replicate the F-Droid and Aurora Store technique explained in the previous tutorial. The addition of Google Play will allow you to natively install Android applications as if you were holding a real telephone or tablet. Launch Google Play and you will be able to search, install, and execute most apps to your new virtual device. After you install a new program, click on the applications menu. Click and hold the new app and you will be able to drag it to your home screen. Figure 7.06 (left) displays the screen of my default investigation emulator. Next, you should understand the features embedded into the Genymotion software.

When you launch an Android virtual machine, you will see a column on the right side of the window and a row of icons horizontally on the bottom. The bottom icons are part of the emulated Android system. Clicking the first icon will navigate you backward one screen from your current location. If you are within an app, this would take you back one step each time that you press it. The second icon represents the "Home" option and will always return you to the home screen. The third button is the "Recent Apps" option and it will load a view of

recently opened applications. The icons on the right of the emulator are features of Genymotion and allow you to control aspects of the Android machine from outside of the emulator. These replicate the features discussed with Android Studio. Note that many options are not available in the free version, but I have never found that to be a hindrance to my investigations. Genymotion is quite clear that if you plan on making money by designing an app through their product, you should pay for a license. Non-commercial usage allows unlimited use of the free personal version.

The GPS option within Genymotion is the most beneficial feature of their toolset. Clicking this icon and clicking the Off/On switch will execute the location spoofing service. You can either supply the exact coordinates directly or click on the "Map" button to select a location via an interactive Google map. Figure 7.06 (middle) displays the default GPS menu in the disabled state. Figure 7.06 (right) displays coordinates entered. I recommend changing the altitude, accuracy, and bearing settings to "0". Close this window and you will see a green check mark in the GPS button to confirm that your location settings are enabled.

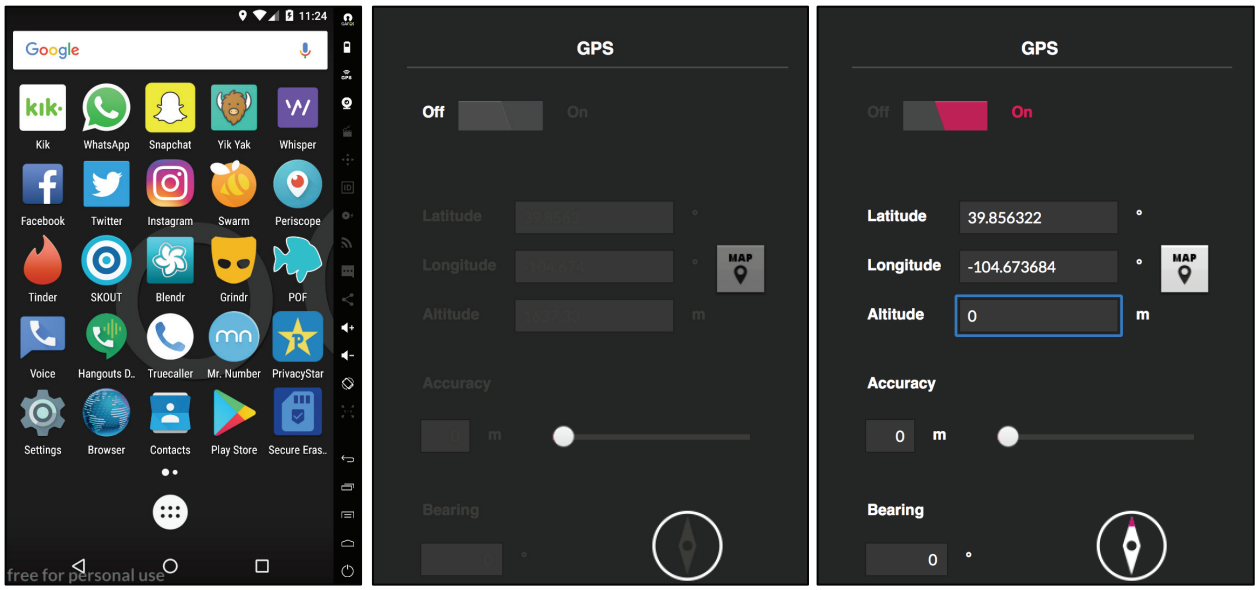


Figure 7.06: An Android screen with apps (left), disabled GPS menu (middle) and spoofed GPS (right).

Virtual Device Cloning

Similar to the tutorials for cloning Linux virtual machines, we can apply the same process toward our new Genymotion Android VM. You should consider a new Android virtual device every time to research a target. The following instructions will clone the exact state of any virtual Android device within VirtualBox, including devices created within Genymotion.

- Create and customize an Android virtual device as desired. Configure all apps that you want present in all cloned copies. Optionally, execute the app "Secure Eraser" to eliminate unnecessary hard drive space. Shut down the machine completely.
- Open VirtualBox from your Applications folder (Mac) or Start menu (Windows). Right-click the machine that you want to duplicate and select "Clone". Figure 7.07 displays this program with a right-click menu option from an active machine.
- Provide a name for your new machine. This could be "Investigation Original Copy" or "2021-1234". Choose the options of "Full Clone" and "Current Machine State" and click the "Clone" button. VirtualBox will create an exact duplicate of the chosen machine in the default folder for VirtualBox machines. You can identify this folder by right-clicking your new machine and choosing "Show in Finder" (Mac) or "Show on disk" (Windows).

You can now use this cloned device to conduct your investigation. Any changes made within it will have no impact on the original device. In fact, I titled my original investigation device "Android Original 9.0", as seen in Figure 7.07. This way, I know to only open it to apply updates, and never for active investigations. Every time I need to use a device to research a target, I quickly clone the original and keep all of my cases isolated.

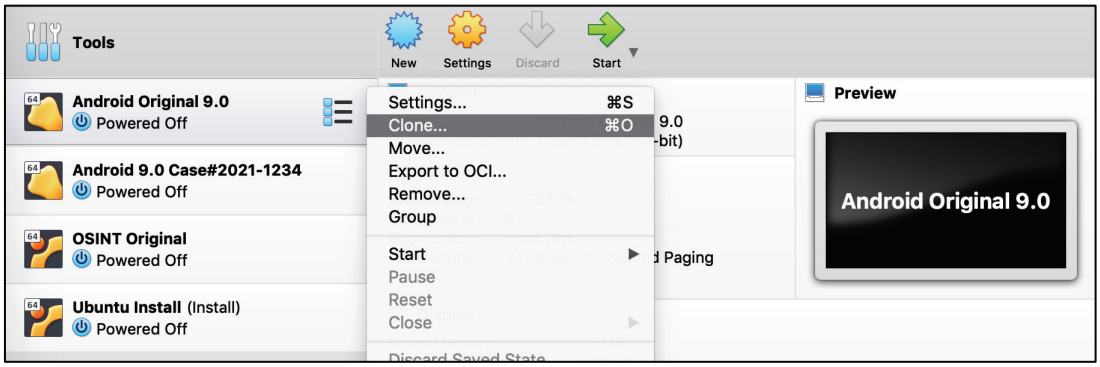


Figure 7.07: A VirtualBox menu with a clone option in the menu.

Virtual Device Export

You may be asked to provide all digital evidence from your investigation as a matter of discovery. This could happen to a forensic examiner hired in a civil case or law enforcement prosecuting a criminal case. This is the precise reason that I create a new virtual device for all my investigations. Not only is it a clean and fair environment, it is easy to archive and distribute when complete. The following instructions will generate a large single file that contains the entire virtual operating system and apps from your investigation.

- Open VirtualBox in the same manner as previously mentioned.
- Select the target virtual device, click on "File" in the menu bar, and select "Export Appliance". Select the device again and provide the save location and name of the file.
- Click "Export" and allow the process to complete. The final result will consist of a single file that can be archived to DVD or flash media.
- This file can be imported into VirtualBox by choosing the "Import Appliance" option in the File menu. This would allow another investigator to view the exact investigation environment as you.

Android within Android Studio and Genymotion are not your only options. Third-party applications such as **BlueStacks** (bluestacks.com), **Andy** (andyroid.net), and **NoxPlayer** (bignox.com) all offer the same basic functionality with added overhead and requirements. After installation, most of these programs work the same way as VirtualBox. In fact, most of them rely on VirtualBox on the back end**. I choose Android Studio over these because it is the most native way to present Android.** It is also quite fast.

I encourage you to experiment with all of the options, and choose any that work best for you. I always keep a version of Android Studio and Genymotion available and updated at all times. At the time of this writing, my Android Studio machine seemed more responsive and functional, but my Genymotion devices were more isolated. Only you can decide the best path for your investigations, but I encourage you to explore both options.

Overall, I believe the future of OSINT collection will become more focused on mobile apps that have no website search option. In order to conduct thorough online investigations, mobile environment emulation is required. I highly recommend practicing these techniques with non-essential apps and data. This will better prepare you for an actual investigation with proper evidence control. Expect frustration as apps block access from within virtual devices due to fraud. However, the occasional investigation success through virtual Android environments justifies all of the headaches encountered along the way.

CHAPTER EIGHT
CUSTOM SEARCH TOOLS

From 2010 through 2019, I offered a set of public free interactive online investigations tools. In June of 2019, I was forced to remove these tools due to abuse and legal demands. They returned to my website in 2022, exactly three years later. In the previous edition of this book, I offered an offline version of these tools which could be self-hosted and immune from vague takedown requests. This chapter revisits these tools and offers several enhancements. The goal of this chapter is to help you create and maintain your own custom search tools which can automate queries for any investigation. First, let's talk about why this is so important.

I assumed my search tools would be around as long as I maintained my site. I learned the hard way that nothing lasts forever. We can no longer rely on third-party tools, a theme which I have overly-emphasized throughout this entire book. Any online search tool outside of your control could disappear any day. While mine have returned, they could go away again at any time.

That is not the worst possible outcome. We never truly know what information various online search tools are storing about our searches and activity. Many aggregated search sites possess numerous tracking cookies and practically all "link collections" force embedded analytics capturing data about every visitor. Creating and hosting your own tools eliminate these issues. We still must query sensitive data to a final repository of information, but let's eliminate the middle-man. All of the tools presented in this chapter, and referenced throughout the book, do not need to be placed online within a website. You can store them on your computer and launch them without fear of questionable connections. Let's get started.

First, download a copy of all search tool templates used within the entire book. This can be found at **https://inteltechniques.com/data/osintbook10/tools.zip**. Enter a username of "**osint**" and password of "**book1443**" (without quotes) if required. Unzip this archive to a destination of your choice. If using the Linux, Mac, or Windows OSINT builds which were previously explained, you should already have the necessary files on your Desktop. I always suggest saving them to the Desktop for easy access. However, anywhere should suffice. Be sure to extract all of the files within the zip file.

This collection reveals a folder titled "Tools" consisting of multiple files within it. Technically, you have everything you need to replicate my public search tools locally on your own machine. However, it is up to you to modify these as needed. You will eventually want to remove dead sources, add new features, and modify the structures due to changes at third-party websites. I will use my Email Tool as a demonstration. Figure 8.01 displays the current view of the Email tool. As you can see, there are several individual search options and a "Submit All" feature at the bottom. Inserting an email address into any of these fields will query that address through the designated option, or the final field executes a search through all of them.

Let's pick apart one of these queries from within the code. By default, double-clicking on any of the files within the search tool folder opens the selected option within your default web browser. This is required for any of them to function. In order to edit the files, we must open them within an HTML editing tool or any text processing application. If you are on a Mac, that could be TextEdit, Windows users have Notepad, and Linux users have Text Edit. All work fine for our needs. Lately, I prefer **VSCodium** (vscodium.com), which is a cross-platform free text editor.

If you open the file titled email.search.html within a text editor ("File" > "Open"), you will see the code which makes this document function within a web browser. The following explains each section. Complete understanding of each term is not required to use and modify your own tools, but these pages may serve as a reference if you ever want to make substantial changes.

```
<!DOCTYPE html><html>
```

This informs a web browser that this is a web page, even if offline, and begins the page.

```
<style>
ul {list-style-type: none;margin: 0;padding: 0;width: 200px;background-color: #f1f1f1;}
li a {display: block;color: #000;padding: 8px 16px;text-decoration: none;}
li a:hover {background-color: #555;color: white;}
li a.active {background-color: #303942;color: white;}
li a.grey {background-color: #cdcdcd;color: black;}
li a.blue {background-color: #b4c8da;color: black;}
table td, table td * {vertical-align: top;}</style>
```

This sets the style requirements such as colors and sizes of the content within the page. You can experiment with these settings without risking the function of the tools.

```
<head>
```

This informs your browser that the "head" or "header" portion of the page begins now.

```
<title>Email Search Tool</title>
```

This represents the title of the page, visible in the browser tab.

```
</head>
```

This discloses the end of the "head" or "header" section.

```
<body>
```

This informs your browser that the "body" portion of the page begins now.

```
<table width="1000" border="0"><td width="200"><td width="800">
```

This creates a table within our content and sets the overall width with no border. It then specifies the width of the columns. The data in between identifies the menu items visible on the left of the page, which are considered the first column within the table.

```
<script type="text/javascript">
```

This identifies the following text as a JavaScript command.

```
function doPopAll(PopAll)…
```

This provides instruction to the browser which allows our tools to populate given data to the remaining fields. It is required for the next option.

```
<form onsubmit="doPopAll…
```

This section creates the "Populate All" button which populates the given data throughout the remaining tools.

```
Function doSearch01(Search01)
```

This tells the page we want it to "do" something, and the task is called Search01.

```
{window.open('https://haveibeenpwned.com/unifiedsearch/' + Search01,'Search01window');}
```

This instructs the page to build a URL, add a piece of data, and open the result in a tab.

```
</script>
```

This identifies the end of each script.

```
<form onsubmit="doSearch01(this.Search01.value); return false;">
```

This creates a form to generate the URL, looking for a specific value.

```
<input type="text" name="Search01" "id="Search01" size="30" placeholder="Email Address"/>
```

This creates a form input identified as Search01 with "Email Address" populated in the field.

```
<input type="submit" style="width:120px" value="HIBP Breaches" /><br /></form>
```

This creates the Submit button with specific text inside, inserts a new line, and closes the form.

```
</table></body></html>
```

This identifies the end of the "table", "body", and "HTML" sections, and closes the page.

This only represents the first search option within this tool, but it is quite powerful. This collects a target email address and queries the website Have I Been Pwned to identify known data breaches containing this account. This technique will be explained in more detail later in the Email chapter. This also demonstrates the need for a search tool versus simply visiting the search site. If you go to haveibeenpwned.com, you can enter an email address to conduct your search. The new page presented does not include a static URL for that search. The page with your results still possesses a simplified address of haveibeenpwned.com, and not something static and similar to haveibeenpwned.com/test@email.com. Bookmarking this page would not present these search results. This is why I use the address https://haveibeenpwned.com/unifiedsearch/test@email.com. It presents the same content, but is a text-only format. Below is another example to explain this.

Conducting the search on the official site presents a graphical output similar to that seen in Figure 8.02. However, the static address I just mentioned presents a view from the Have I Been Pwned application programming interface (API), as seen in Figure 8.03. The same data can be found in each offering, but the text view can be copied and pasted more easily. It also possesses a static URL which can be referenced in your report and recreated later. You may be wondering where this URL came from. It is not advertised on the site, and is not an official option within the API (which is now a paid service, but this URL is free). That is our next tutorial. **Please note there will be occasional references to "APIs" throughout the book, but Chapter Twenty-Eight is devoted entirely to API usage and scripting.**

Navigate to haveibeenpwned.com within Firefox and allow the page to load. Conduct the following steps to identify the exact URL which submits a query to return data about your target.

- Right-click on the page and choose "Inspect Element".
- Click the "Network" tab in the new window at the bottom of the page.
- Type an email address into the website and execute the search.
- Scroll through the new text in the Inspector window at the bottom of the page.
- Click on the result displaying the target email address with "xhr" in the "Cause" column.
- Copy the URL in the window to the right under "Headers" as seen in Figure 8.04.

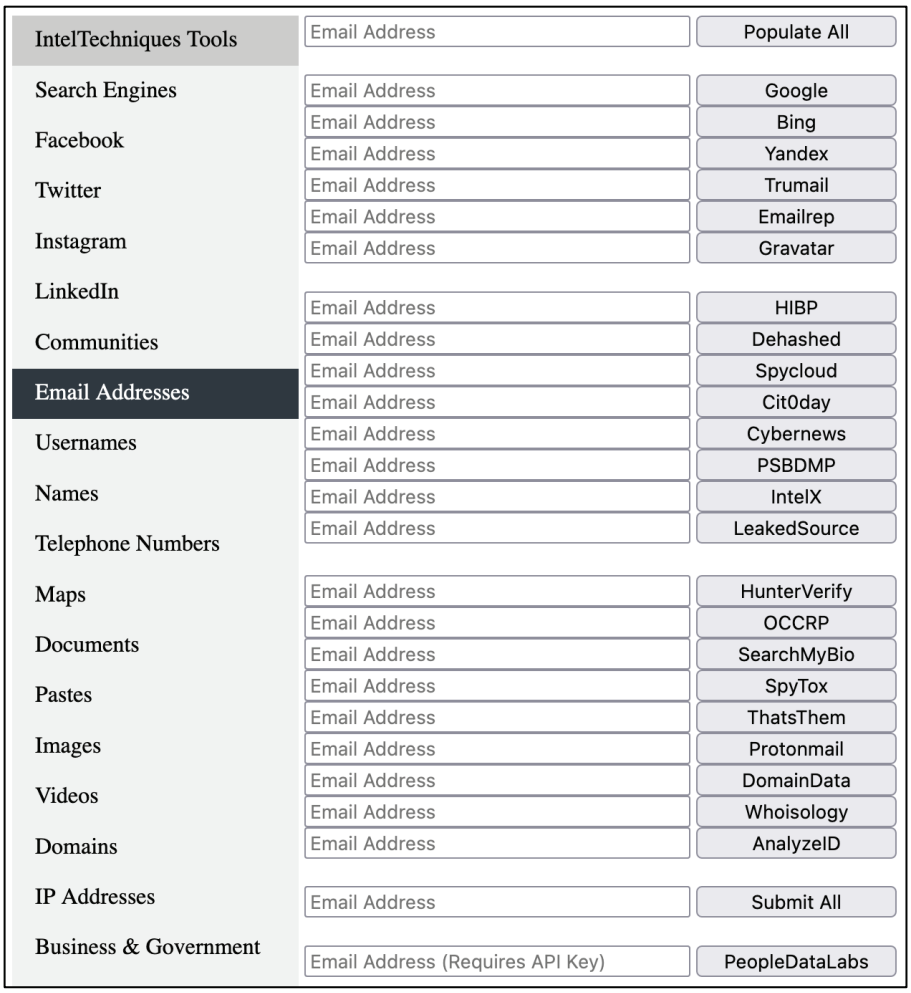


Figure 8.01: The Email Addresses Tool.

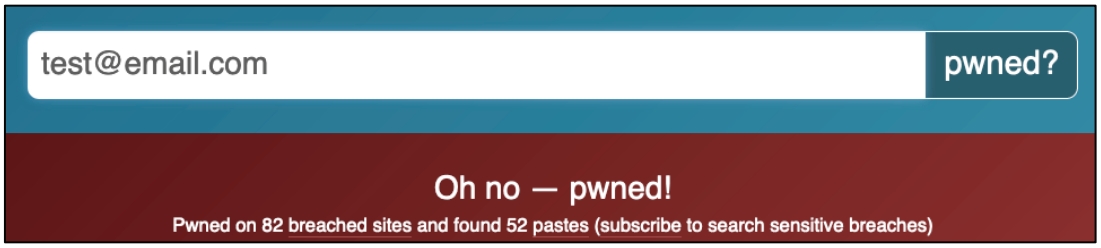


Figure 8.02: A result from the Have I Been Pwned website.

Figure 8.03: A result from the Have I Been Pwned API.

Figure 8.04: The static URL from a query as seen in Inspector.

With this method, we can identify the URL structures for our tool. In the example displayed previously, our tool presented a line of code which included the URL required for the search.

```
{window.open('https://haveibeenpwned.com/unifiedsearch/' + Search01,'Search01window');}
```

This line instructs the tool to open a new browser window, navigate to the website https://haveibeenpwned.com/unifiedsearch/, followed by whatever text was entered into the search tool, and define that new window (or tab) with a unique name in order to prevent another search within our tool from overwriting the page. This results in our tool opening a new tab with our desired results (https://haveibeenpwned.com/unifiedsearch/test@email.com). Let's look at another search option within this same tool with a slightly different structure.

The Dehashed search option is unique in that it requires quotation marks surrounding the input. In other words, you must enter "test@email.com" and not simply test@email.com. This requires us to add an additional character after the target input has been provided. Below is the example for this query. Note that double quotes (") are inside single quotes ('), which appears quite messy in print. Always rely on the digital files to play with the actual content.

```
{window.open('https://dehashed.com/search?query="' + Search05 + '"', 'Search05window');}
```

This line instructs the tool to open a new browser window, navigate to the website https://dehashed.com/search?query=", followed by whatever text was entered into the search tool, plus another quotation mark (a single quote, double quote, and another single quote), and define that new window (or tab) with a unique name in order to prevent another search within our tool from overwriting the page. The lesson here is that you can add as many parameters as necessary by using the plus (+) character. You will see many examples of this within the files that you have downloaded. Remember, every search tool presented in this book is included in digital format. You only need to modify the options as things change over time.

Let's assume Dehashed made a change to their search, which appears in the tool as follows.

```
{window.open('https://dehashed.com/search?query="' + Search05 + '"', 'Search05window');}
```

This is because the URL structure of the search is as follows:

https://dehashed.com/search?query="test@email.com"

Assume that Dehashed changed the search structure on their site to the following:

https://dehashed.com/?query="test@email.com"&trial

Your new line within the tool would need to be manipulated as follows:

```
{window.open('https://dehashed.com/?query="' + Search05 + '"&trial', 'Search05window');}
```

Next, let's assume that you found a brand-new search service which was not included in the downloadable search tools. You will need to modify the tools to include this new option. Again, we will use the email tool as an example. Open the "Email.html" file within a text editor. Look through the text and notice that each search

script possesses an identifier similar to "Search01", "Search02", "Search03", etc. These must each be unique in order to function. You will notice that the final option (after the Submit All feature) is "Search25". We now know that our next option should be "Search26". Assume that you found a website at emailleaks.com and you want to add it to the tools. A query of an email address presents a URL as follows.

https://emailleaks.com/ajax.php?query=test@email.com

You would next copy the "Search24" script and paste it at the end of the tool (before the Submit All feature). You can then edit the script, which should look like the following, using "Search26" and our new URL.

```
<script type="text/javascript">
function doSearch26(Search26)
{window.open('https://emailleaks.com/ajax.php?query=' + Search26, 'Search26window');}
</script>
<form onsubmit="doSearch26(this.Search26.value); return false;">
<input type="text" name="Search26" "id="Search26" size="30" placeholder="Email Address"/>
<input type="submit" style="width:120px" value="Email Leaks" /><br /></form>
```

All we changed within this copy and paste job was the target URL, the Search26 identifiers, and the descriptor. You can place this section anywhere within the tools, as it does not need to be at the end. Note it is titled Search26, so any new options added would need to start with Search27. These numbers do not need to be sequential throughout the tool, but they must be unique.

Submit All

Many of the online search tools offer a "Submit All" button at the bottom of the options. This executes each of the queries referenced above the button and can be a huge time saver. If you open one of the search tools with this option in a text editor, you will see the code for this at the bottom. It appears very similar to the other search options, but there are multiple "window.open" elements such as those listed below.

```
Window.open('https://haveibeenpwned.com/unifiedsearch/' + all, 'Search01window');
window.open('https://dehashed.com/search?query=' + all, 'Search05window');
```

In most of the tools, I have simply replicated the individual search options within one single "Submit All" feature. If you modify a search tool within the code next to the manual search, you should also update it under the final option to execute all queries. If you feel overwhelmed with all of this, do not panic. None of this is required at this point. **Your own custom offline search tools are already configured and functioning**. If a specific desired tool stops functioning, you can use this chapter to change your scripts.

You may have noticed that there are several files within the Tools folder. Launching any of these opens that specific tool, such as "Email.html", but a menu exists within each of the pages in order to navigate within the tool to the desired page. The file titled "index.html" is the "Main menu", and might be appropriate to set as your browser's home page. Clicking on the desired search option within the left side of the menu opens that specific tool. As an example, clicking on "Twitter" presents numerous Twitter search options. These will each be explained at the end of each corresponding chapter.

Simplified Modification

I am sure some readers are frustrated at the technology presented here. Some may look at this code and cite numerous ways it could be made better. I agree this is amateur hour, as I am not a strong HTML coder. Other readers may be confused at all of this. For those, there are two options which simplify things. First, ignore this entire chapter and simply use the free tools without any modification. Some options will break eventually as sites come and go, but that should not impact the other fields. Second, don't worry too much about adding new

features. Instead, simply replace any searches that stop functioning. If Dehashed shuts down tomorrow, simply wait for a replacement. When that happens, modify only the URL and name, leaving the structure as-is.

You have a strong start with the current tools template. Very minimal modifications as things break will keep you in good shape. Any major updates which I perform to my own set of tools will be offered on my site for download. Check the "Updates" section at the following page.

https://inteltechniques.com/osintbook10

Populate All

You may have noticed that most of the tools have an option to populate all of the fields from a single entry. This is beneficial as it prevents us from copying and pasting target data within multiple fields. This code, which was presented earlier, tells your browser to populate anything you place into the first field within every field on that page which has an ID of "Search" plus any numbers. In other words, it would populate both examples on the previous page because they have id="Search25" and id="Search26". Test this within the Email search tool. Make sure each "id" field is unique, as no two can be the same on one page.

When I need to search a specific target, I do not copy the data into each search field and press the corresponding button for each service. I place the input directly into the "Populate All" option and then execute any individual searches desired. Alternatively, I place my target data into the "Submit All" option and let it go. If using Firefox, this will fail on the first attempt. This is because you have pop-ups blocked by default, and Firefox is trying to protect you from multiple new pages loading automatically. The following steps will prevent this.

- Open the Email.html search tool included in your downloaded offline search tools.
- Place any email address in the last option and press the Submit All button.
- A new tab will open, but close it.
- Back in the Email search tool, you should see a yellow banner at the top.
- Click the Preferences button and click the first option to "Allow pop-ups for file".

This will prevent your pop-up blocker from blocking that specific page. You would need to repeat the process for each of the other tools, such as Twitter, Facebook, etc., which can be quite a burden. If desired, you can disable the pop-up blocker completely, but that carries risks. You may visit a malicious website which begins loading new tabs. I do not see this as much as in years past, but the threat does still exist. If conducting your research within a VM, I do not see a huge risk in disabling this blocker. If you do, all of the tools will function without specific modifications to the blocker. Make this decision carefully.

- Click the Firefox menu in the upper right and choose Preferences or Options.
- Click on Privacy & Security and scroll to Permissions.
- If desired, uncheck the "Block pop-up windows" option.

While I would never do this on my primary browser used for personal activity on my main computer, I have disabled the pop-up blocker within my OSINT Original VM (and therefore all clones). It simply saves me headaches when trying to use automated tools. If only using the single queries within the tool, your pop-up blocker will not interfere. I highly recommend that you become familiar with these search tools before you rely on them. Experience how the URLs are formed, and understand how to modify them if needed. Each of these tools will be explained in the following chapters as we learn all of the functions.

License & Warranty

These tools are released to you for free. Full details of allowances and restrictions can be found in the "License.txt" file and "License" link within the tools download. The tools are provided "as is", without warranty

of any kind. Please follow my blog or Twitter account for any updates. Ultimately, it is your responsibility to update your tools as desired as things change after publication. The torch has been passed.

Easy Access

Regardless of where you save your set of tools, I highly recommend that you create a bookmark within your browser for easy access. I prefer them to be within my bookmarks toolbar so that they are always one click away. Navigate to your search tools. If using the Linux, Mac, or Windows OSINT machines, they are in the Tools folder on your desktop. Double-click the file titled "Search.html" and it should open within your default browser, preferably Firefox. If the page opens in Chrome or another browser, open Firefox and use the file menu to select "Open File" and browse to the "Search.html" file. After the page loads, create a bookmark. In Linux and Windows, press "Ctrl" + "D" ("command" + "D" on Mac). When prompted, provide a name of "Tools" and save the page in the folder titled "Bookmarks Toolbar". You should now see a new bookmark in your browser's toolbar titled "Tools". If your Bookmarks Toolbar is not visible, click on "View", then "Toolbars", then "View Bookmarks Toolbar". You can now click this new button within your toolbar at any time and immediately load the Search Engines tool. Clicking through the other options in the left menu of that page should present all other search tool pages. I use this shortcut to launch my tools daily.

Online Version

As previously explained, I offer an updated live copy of all tools at **https://inteltechniques.com/tools/**. These tools are visually similar to my overall website theme, but the code powering them is identical to the strategy you have seen here and the offline files previously downloaded. As resources change, I will apply all modifications to my copy online, but I cannot modify your current offline set.

Some may still not be convinced they should maintain their own search tools and should just rely on my version on the website. The live tools could disappear at any time due to another takedown demand. Also, you cannot modify my live online version as you can your own copy. Finally, you must rely on my site being available during your investigations. The offline version is available on your desktop at any time.

Please use these responsibly. I am optimistic that we can keep this valuable resource for our daily investigations for years to come. Figure 8.05 displays an example of the updated online tools.

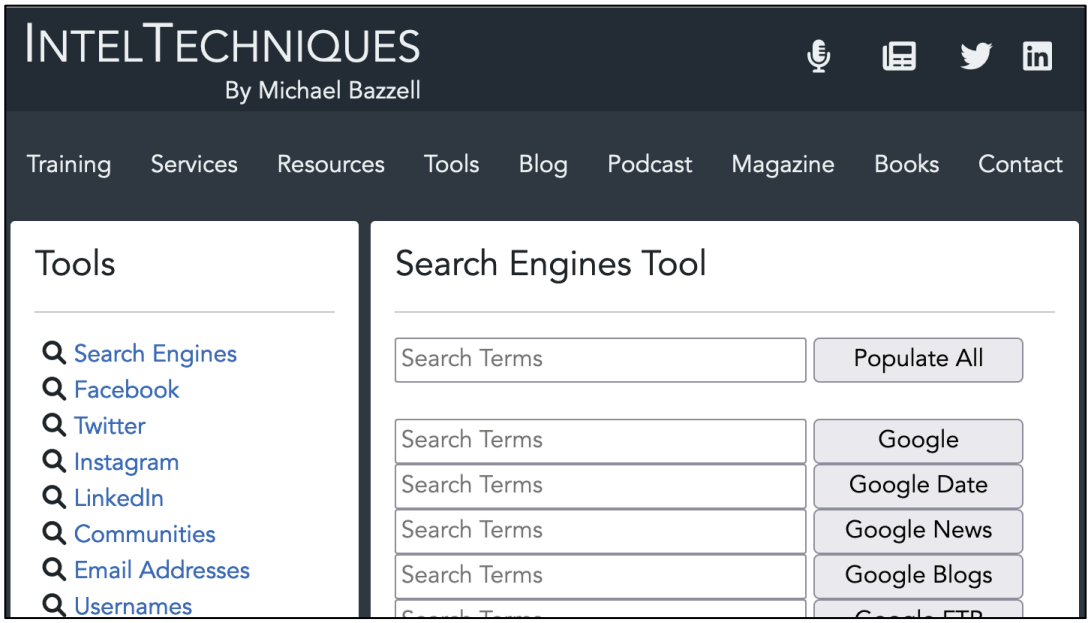


Figure 8.05: The online IntelTechniques search tools.

SECTION II
OSINT RESOURCES & TECHNIQUES

Some may consider this section to be the "guts" of the book. It contains the OSINT tips, tricks, and techniques which I have taught over the past twenty years. Each chapter was rewritten and confirmed accurate in December 2022. All outdated content was removed, many techniques were updated, and numerous new resources were added. The first four editions of this book only consisted of this section. Only recently have I adopted the preceding preparation section and the methodology topics toward the end. OSINT seems to have become a much more complex industry over the years. It is exciting to watch the community grow and I am honored to play an extremely small role.

This section is split into several chapters, and each explains a common type of target investigation. I have isolated specific topics such as email addresses, usernames, social networks, and telephone numbers. Each chapter provides every valuable resource and technique which I have found beneficial toward my own investigations. No book could ever include every possible resource, as many tools become redundant after a superior version has been identified. I do my best to limit the "noise" and simply present the most robust options for each scenario. This section should serve as a reference when you encounter a specific need within your own investigations.

Covert Accounts

Before proceeding with any of the investigation methods here, it is important to discuss covert accounts, also referred to by some as "Sock Puppets". Covert accounts are online profiles which are not associated with your true identity. Many social networks, such as Facebook and Instagram, now require you to be logged in to an account before any queries can be conducted. Using your true personal account could reveal your identity as an investigator to the target. Covert accounts on all of the social networks mentioned here are free and can be completed using fictitious information. However, some networks will make this task more difficult than others. Google, Facebook, Twitter, Instagram, and Yahoo are known to make you jump through hoops before you are granted access. We begin this chapter discussing ways around this.

Email: It is vital that you possess a "clean" email address for your covert accounts. Every social network requires an email address as a part of account registration, and you should never use an already established personal address. Later chapters explain methods for researching the owners behind email addresses, and those techniques can be applied to you and your own accounts. Therefore, consider starting fresh with a brand-new email account dedicated toward use for covert profiles.

The choice of email provider is key here. I do not recommend GMX, Proton Mail, Yahoo, Gmail, MSN, or any other extremely popular providers. These are heavily used by spammers and scammers, and are therefore more scrutinized than smaller providers. My preference is to create a free email account at **Fastmail** (https://ref.fm/u14547153). This established mail provider is unique in two ways. First, they are one of the only remaining providers which do not require a pre-existing email address in order to obtain a new address. This means that there will be no connection from your new covert account to any personal accounts. Second, they are fairly "off-radar" from big services such as Facebook, and are not scrutinized for malicious activity.

Fastmail will provide anyone unlimited free accounts on a 30-day trial. I suggest choosing an email address that ends in fastmail.us instead of fastmail.com, as that domain is less used than their official address. This is a choice during account creation. Once you have your new email address activated, you are ready to create covert profiles. Note that the free trial terminates your access to this email account in 30 days, so this may not be best for long-term investigations. Personally, I possess a paid account which allows me 250 permanent alias email addresses.

- **Facebook**: This is by far the most difficult in terms of new account creation. For most new users, Facebook will require you to provide a cellular telephone number where a verification text can be sent and confirmed. Providing VOIP numbers such as a Google Voice account will not work anymore. I have found only one solution. Turn off any VPN, Tor Browser, or other IP address masking service and connect from a residential or business internet connection. Make sure you have cleared out all of your internet cache and logged out of any accounts. Instead of creating a new account on facebook.com, navigate directly to m.facebook.com. This is the mobile version of their site which is more forgiving on new accounts. During account creation, provide the Fastmail email address that you created previously. In most situations, you should bypass the requirement to provide a cellular number. If this method failed, there is something about your computer or connection that is making Facebook unhappy. Persistence will always equal success eventually. I find public library Wi-Fi our best internet option during account creation. **Instagram** is similar to (and owned by) Facebook. Expect the same scrutiny.
- **Twitter**: Many of the Twitter techniques presented later will not require an account. However, the third-party solutions will mandate that you be logged in to Twitter when using them. I highly recommend possessing a covert account before proceeding. As long as you provide a legitimate email address from a residential or business internet connection, you should have no issues. You may get away with using a VPN to create an account, but not always.
- **Google/Gmail/Voice**: While Google has become more aggressive at refusing suspicious account registrations, they are still very achievable. As with the previous methods, Google will likely block any new accounts that are created over Tor or a VPN. Providing your Fastmail address as an alternative form of contact during the account creation process usually satisfies their need to validate your request. I have also found that they seem more accommodating during account creation if you are connected through a Chrome browser versus a privacy-customized Firefox browser (Google owns Chrome).
- **Network:** I always prefer to conduct online investigations behind a VPN, but this can be tricky. Creating accounts through a VPN often alerts the service of your suspicious behavior. Creating accounts from public Wi-Fi, such as a local library or coffee shop, are typically less scrutinized. A day after creation from open Wi-Fi, I attempt to access while behind a VPN. I then consistently select the same VPN company and general location upon every usage of the profile. This builds a pattern of my network and location, which helps maintain access to the account.
- **Phone Number:** The moment any service finds your new account to be suspicious, it will prompt you for a valid telephone number. Landlines and VOIP numbers are blocked, and they will demand a true cellular number. Today, I keep a supply of Mint Mobile SIM cards, which can be purchased for $0.99 from Amazon (https://amzn.to/2MRbGTI). Each card includes a telephone number with a one-week free trial. I activate the SIM card through an old Android phone, select a phone number, and use that number to open accounts across all of the major networks. As soon as the account is active, I change the telephone number to a VOIP option and secure the account with two-factor authentication (2FA).
- **2FA:** Once I have an account created, I immediately activate any two-factor authentication options. These are secondary security settings which require a text message or software token (Authy) in order to access the account. Typically, this behavior tells the service that you are a real person behind the account, and not an automated bot using the profile for malicious reasons.
- **Activity:** After the account is created and secured, it is important to remain active. If you create a new account and allow it to sit dormant for months, it is likely to be suspended the moment you log back in to the account. If you access the account weekly, it is less likely to be blocked.

You may assume that you can use your personal social network accounts to search for information. While this is possible, it is risky. Some services may never indicate to the target that your specific profile was used for searching. Others, such as Facebook, will eventually notify the target that you have an interest in him, usually in the form of friend recommendations. On any service, you are always one accidental click away from sending a friend request from your real account to the suspect. For these reasons, I never use a personal social network profile during any investigation. I maintain multiple covert accounts. The topic of undercover operations quickly exceeds the scope of this book. For our purposes, we simply need to be logged in to valid accounts in order to pacify the social networks. I will assume that you now have some accounts created. Let's dig in to online search.

Profile Content

Possession of an empty profile on a social network may suffice for your investigations. However, lack of personal details might appear suspicious to both the provider and your target. Facebook is well known for suspending accounts which do not contain personal information, and your targets may conduct their own OSINT research into your publicly available details after you begin the hunt. For most scenarios, I believe you should populate a minimum amount of fake details into your covert profiles. You should never provide anything which may be associated with your true identity, such as interests, occupation, or location. Because of this, I rely heavily on randomly-generated and AI-produced content. The resources below have helped me within my own profile creation. Consider your own needs and employer policies before proceeding with your accounts.

Images: You may want a headshot within your profile which adds a layer of authenticity to your new covert account. This can also eliminate scrutiny from Facebook and Twitter when their algorithms suspect your profile to be fraudulent. I recommend **This Person Does Not Exist** (thispersondoesnotexist.com). This site generates a very realistic image of a "person", which is entirely generated by computers. The image you see is not a real person and should not be visible anywhere else online. Refreshing the page generates a new image. If you find this beneficial, I encourage you to generate numerous images for future use in the event the site should disappear.

Name and Background: It may be easy to create your own alias name, but could you quickly generate a maiden name, birthday, birthplace, zodiac sign, username, password, religion, and political view? This is where services such as **ElfQrin** (elfqrin.com/fakeid.php) and **Fake Name Generator** (fakenamegenerator.com) can be beneficial. The example below was created instantly with these services.

Resume: If you want to add another layer of realism to your new online identity, you might consider posting a resume online. If your target begins investigating your profile and finds the resume, you may appear to be a real person.

Physical Space: Finally, you might consider **This Rental Does Not Exist** (thisrentaldoesnotexist.com). It uses the same artificial intelligence technology as This Person Does Not Exist to generate fake interior views of a home. These artificial images are intended to emulate a rental home or Air BNB profile, but they could be used if you ever need to post pictures of your "home".

First name	Ella
Middle name	Theresa
Last name	Bowman
Mother's Maiden name	Sutton
Birthday	November, 07 1998 (Age: 23 years)
Birthplace	Walnut Creek, CA, USA
Zodiacal sign	Scorpio
User name	boe431
Password	f2b4qawha
Address	101 Maryland Ave Ne, Washington, DC 20002
Car	Wrangler

Hair color	Black (BLK)	Political side	Independent
Eyes color	Brown (BRO)	Favorite Color	Purple
Height	166 cm / 5 ft 5 in	Favorite Comfort Food	Chocolate
Weight	56 Kg / 123 pounds	Favorite Cereal	Raisin Bran
Shoe Size	7.5	Favorite Season	Spring
Blood Type	B+	Favorite Animal	Elephant
Religion	Jehovah's Witnesses	Lucky Number	2

CHAPTER NINE
SEARCH ENGINES

The first stop for many researchers will be a popular search engine. The two big players in the United States are Google and Bing. This chapter will go into great detail about the advanced ways to use both and others. Most of these techniques can apply to any search engine, but many examples will be specific for these two. Much of this chapter is unchanged from the 9th edition.

Google (google.com)

There are entire books dedicated to Google searching and Google hacking. Most of these focus on penetration testing and securing computer networks. These are full of great information, but are often overkill for the investigator looking for quick personal information. A few simple rules can help locate more accurate data. No book in existence will replace practicing these techniques in a live web browser. When searching, you cannot break anything. Play around and get familiar with the advanced options.

Quotation Marks

Placing a target name inside of quotation marks will make a huge difference in a quick first look for information. If I conducted a search for my name without quotes, the result is 147,000 pages that include the words "Michael" and "Bazzell". These pages do not necessarily have these words right next to each other. The word "Michael" could be next to another person's name, while "Bazzell" could be next to yet another person's name. These results can provide inaccurate information. They may include a reference to "Michael Santo" and "Barry Bazzell", but not my name. Since technically the words "Michael" and "Bazzell" appear on the page, you are stuck with the result in your list. In order to prevent this, you should always use quotes around the name of your target. Searching for the term "Michael Bazzell", including the quotes, reduces the search results to 31,800.

Each of these pages will contain the words "Michael" and "Bazzell" right next to each other. While Google and other search engines have technology in place to search related names, this is not always perfect, and does not apply to searches with quotes. For example, the search for "Michael Bazzell", without quotes, located pages that reference Mike Bazzell (instead of Michael). This same search with quotes did not locate these results. Placing quotes around any search terms tells Google to search exactly what you tell it to search. If your target's name is "Michael", you may want to consider an additional search for "Mike". If a quoted search returns nothing, or few results, you should remove the quotes and search again.

When your quoted search, such as "Michael Bazzell", returns too many results, you should add to your search. When I add the term "FBI" after my name, the results reduce from 31,800 to 12,000. These results all contain pages that have the words "Michael" and "Bazzell" next to each other, and include the term "FBI" somewhere on the page. While all of these results may not be about me, the majority will be and can be easily digested. Adding the occupation, residence city, general interest, or college of the target may help eliminate unrelated results. This search technique can be vital when searching email addresses or usernames. When searching the email address of "michael@inteltechniques.com", without quotes, I receive 14,200 results. When I search "michael@inteltechniques.com" with quotes, I receive only 7 results that actually contain that email address (which does not reach my inbox).

Search Operators

Most search engines allow the use of commands within the search field. These commands are not actually part of the search terms and are referred to as operators. There are two parts to most operator searches, and each are separated by a colon. To the left of the colon is the type of operator, such as "site" (website) or "ext" (file

extension). To the right is the rule for the operator, such as the target domain or file type. The following will explain each operator and the most appropriate uses.

Site Operator

Google, and other search engines, allow the use of operators within the search string. An operator is text that is added to the search, which performs a function. My favorite operator is the "site:" function. This operator provides two benefits to the search results. First, it will only provide results of pages located on a specific domain. Second, it will provide all of the results containing the search terms on that domain. I will use my name again for a demonstration. I conducted a search of "Michael Bazzell" on Google. One of the results is a link to the website forbes.com. This search result is one of multiple pages on that domain that includes a reference to me. However, this search only displayed one of the many pages on that domain that possessed my name within them. If you want to view every page on a specific domain that includes your target of interest, the site operator is required. Next, I conducted the following exact search.

site:forbes.com "Michael Bazzell"

The result was all eight pages on forbes.com that include my name within the content. This technique can be applied to any domain. This includes social networks, blogs, and any other website that is indexed by search engines.

Another simple way to use this technique is to locate every page that is part of a specific domain. A search query of site:inteltechniques.com displays all 628 pages that are publicly available on my personal website. This can be a great way to review all the content of a target's personal website without attempting to navigate the actual site. It is very easy to miss content by clicking around within a website. With this technique, you should see all of the pages in a format that is easy to digest. Also, some of the pages on a website that the author may consider "private" may actually be public if he or she ever linked to them from a public page. Once Google has indexed the page, we can view the content using the "site" operator.

Real World Application: While conducting private background checks, I consistently use the site operator. A search such as "site:amazon.com" and the target name can reveal interesting information. A previous background check of an applicant that signed an affidavit declaring no previous drug or alcohol dependencies produced some damaging results. The search provided user submitted reviews that he had left on Amazon in reference to books that he had purchased that assisted him with his continual addiction to controlled substances. Again, this result may have appeared somewhere in the numerous general search results of the target; however, the site operator directed me exactly where I needed to look.

File Type Operator

Another operator that works with both Google and Bing is the file type filter. It allows you to filter any search results by a single file type extension. While Google allows this operator to be shortened to "ext", Bing does not. Therefore, I will use the original "filetype" operator in my search examples. Consider the following search attempting to locate PowerPoint presentation files associated with the company Cisco.

"Cisco" "PowerPoint"

The result is over 10,000,000 websites that include the words Cisco and PowerPoint in the content. However, these are not all actual PowerPoint documents. The following search refines our example for accuracy.

"Cisco" filetype:ppt

The result is 15,200 Microsoft PowerPoint presentations that contain Cisco within the content. This search only located the older PowerPoint format of PPT, but not newer files that may have the PPTX extension. Therefore, the following two searches would be more thorough.

"Cisco" filetype:ppt
"Cisco" filetype:pptx

The second search provided an additional 12,700 files. This brings our total to over 27,000 PowerPoint files, which is overwhelming. I will begin to further filter my results in order to focus on the most relevant content for my research. The following search will display only newer PowerPoint files that contain the exact phrase Cisco Confidential within the content of the slides.

"Cisco Confidential" filetype:pptx

The result is exactly 1,080 PowerPoint files of interest. There are many uses for this technique. A search of filetype:doc "resume" "target name" often provides resumes created by the target which can include cellular telephone numbers, personal addresses, work history, education information, references, and other personal information that would never be intentionally posted to the internet. The "filetype" operator can identify any file by the file type within any website. This can be combined with the "site" operator to find all files of any type on a single domain. By conducting the following searches, I was able to find several documents stored on the website irongeek.com.

site:irongeek.com filetype:pdf
site:irongeek.com filetype:ppt
site:irongeek.com filetype:pptx

Previously, Google and Bing indexed media files by type, such as MP3, MP4, AVI, and others. Due to abuse of pirated content, this no longer works well. I have found the following extensions to be indexed and provide valuable results.

7Z: Compressed File
BMP: Bitmap Image
DOC: Microsoft Word
DOCX: Microsoft Word
DWF: Autodesk
GIF: Animated Image
HTM: Web Page
HTML: Web Page
JPG: Image
JPEG: Image
KML: Google Earth
KMZ: Google Earth
ODP: OpenOffice Present
ODS: OpenOffice Spreadsheet
ODT: OpenOffice Text
PDF: Adobe Acrobat
PNG: Image
PPT: Microsoft PowerPoint
PPTX: Microsoft PowerPoint
RAR: Compressed File
RTF: Rich Text Format
TXT: Text File
XLS: Microsoft Excel
XLSX: Microsoft Excel
ZIP: Compressed File

Hyphen (-)

The search operators mentioned previously are filters to include specific data. Instead, you may want to exclude some content from appearing within results. The hyphen (-) tells most search engines and social networks to exclude the text immediately following from any results. It is important to never include a space between the hyphen and filtered text. The following searches were conducted on my own name with the addition of excluded text. Following each search is the number of results returned by Google.

"Michael Bazzell" 31,800
"Michael Bazzell" -police 28,000
"Michael Bazzell" -police -FBI 22,100
"Michael Bazzell" -police -FBI -osint 6,010
"Michael Bazzell" -police -FBI -osint -books 4,320

"Michael Bazzell" -police -FBI -osint -books -open -source 604
"Michael Bazzell" -police -FBI -osint -books -open -source -"mr. robot" 92

The final search eliminated results which included any of the restricted words. The pages that were remaining referenced other people with my name. My goal in search filters is to dwindle the total results to a manageable amount. When you are overwhelmed with search results, slowly add exclusions to make an impact on the amount of data to analyze.

InURL Operator

We can also specify operators that will focus only on the data within the URL or address of the website. Previously, the operators discussed applied to the content within the web page. My favorite search using this technique is to find File Transfer Protocol (FTP) servers that allow anonymous connections. The following search would identify any FTP servers that possess PDF files that contain the term OSINT within the file.

inurl:ftp -inurl(http|https) filetype:pdf "osint"

The following will dissect how and why this search worked.

inurl:ftp - Instructs Google to only display addresses that contain "ftp" in the URL.

-inurl(http|https) - Instructs Google to ignore any addresses that contain either http or https in the URL. The separator is the pipe symbol (|) located above the backslash key. It tells Google "OR". This would make sure that we excluded any standard web pages.

filetype:pdf - Instructs Google to only display PDF documents.

"osint" - Instructs Google to mandate that the exact term osint is within the content of results.

Obviously, this operator could also be used to locate standard web pages, documents, and files. The following search displays only blog posts from inteltechniques.com that exist within a folder titled "blog" (WordPress).

inurl:/blog/ site:inteltechniques.com

InTitle Operator

Similar to InURL, the "InTitle" operator will filter web pages by details other than the actual content of the page. This filter will only present web pages that have specific content within the title of the page. Practically every web page on the internet has an official title for the page. This is often included within the source code of the page and may not appear anywhere within the content. Most webmasters carefully create a title that will be best indexed by search engines. If you conduct a search for "osint video training" on Google, you will receive 2,760 results. However, the following search will filter those to 5. These only include web pages that had the search terms within the limited space of a page title.

intitle:"osint video training"

Note that the use of quotation marks prevents the query from searching "video training" within websites titled "osint". The quotes force the search of pages specifically titled "osint video training". You can add "all" to this search to force all listed words to appear in any order. The following would find any sites that have the words osint, video, and training within the title, regardless of the order.

allintitle:training osint video

An interesting way to use this search technique is while searching for online folders. We often focus on finding websites or files of interest, but we tend to ignore the presence of online folders full of content related to our search. As an example, I conducted the following search on Google.

intitle:index.of OSINT

The results contain online folders that usually do not have typical website files within the folders. The first three results of this search identified the following publicly available online data folders. Each possess dozens of documents and other files related to our search term of OSINT. One provides a folder structure that allows access to an entire web server of content. Notice that none of these results points to a specific page, but all open a folder view of the data present.

http://cyberwar.nl/d/
http://bitsavers.trailing-edge.com/pdf/
http://conference.hitb.org/hitbsecconf2013kul/materials/

OR Operator

You may have search terms that are not definitive. You may have a target that has a unique last name that is often misspelled. The "OR" (uppercase) operator returns pages that have just A, just B, or both A and B. Consider the following examples which include the number of results each.

"Michael Bazzell" OSINT 61,200
"Mike Bazzell" OSINT 1,390
"Michael Bazzell" OR "Mike Bazzell" OSINT 18,600
"Michael Bazell" OR "Mike Bazell" OSINT 1,160
"Michael Bazzel" OR "Mike Bazzel" OSINT 582

Asterisk Operator (*)

The asterisk (*) represents one or more words to Google and is considered a wild card. Google treats the * as a placeholder for a word or words within a search string. For example, "osint * training" tells Google to find pages containing a phrase that starts with "osint" followed by one or more words, followed by "training". Phrases that fit this search include: "osint video training" and "osint live classroom training".

Range Operator (..)

The "Range Operator" tells Google to search between two identifiers. These could be sequential numbers or years. As an example, OSINT Training 2015..2018 would result in pages that include the terms OSINT and training, and also include any number between 2015 and 2018. I have used this to filter results for online news articles that include a commenting system where readers can express their views. The following search identifies websites that contain information about Bonnie Woodward, a missing person, and between 1 and 999 comments within the page.

"bonnie woodward" "1..999 comments"

Related Operator

This option has been proven to be very useful over the past year. It collects a domain, and attempts to provide online content related to that address. As an example, I conducted a search on Google with the following syntax.

related:inteltechniques.com

The results included no references to that domain, but did associate it with my other websites, my Twitter page, my Black Hat courses, and my book on Amazon. In my investigations, this has translated a person's personal website into several social networks and friends' websites.

Google Search Tools

There is a text bar at the top of every Google search result page. This allows for searching the current search terms within other Google services such as Images, Maps, Shopping, Videos, and others. The last option on this bar is the "Tools" link. Clicking this link will present a new row of options directly below. This provides new filters to help you focus only on the desired results. The filters will vary for each type of Google search. Figure 9.01 displays the standard search tools with the time menu expanded.

The "Any time" drop-down menu will allow you to choose the time range of visible search results. The default is set to "Any time" which will not filter any results. Selecting "Past hour" will only display results that have been indexed within the hour. The other options for day, week, month, and year work the same way. The last option is "Custom range". This will present a pop-up window that will allow you to specify the exact range of dates that you want searched. This can be helpful when you want to analyze online content posted within a known time.

Real World Application: Whenever I was assigned a missing person case, I immediately searched the internet. By the time the case is assigned, many media websites had reported on the incident and social networks were full of sympathetic comments toward the family. In order to avoid this traffic, I set the search tools to only show results up to the date of disappearance. I could then focus on the online content posted about the victim before the disappearance was public. This often led to more relevant suspect leads.

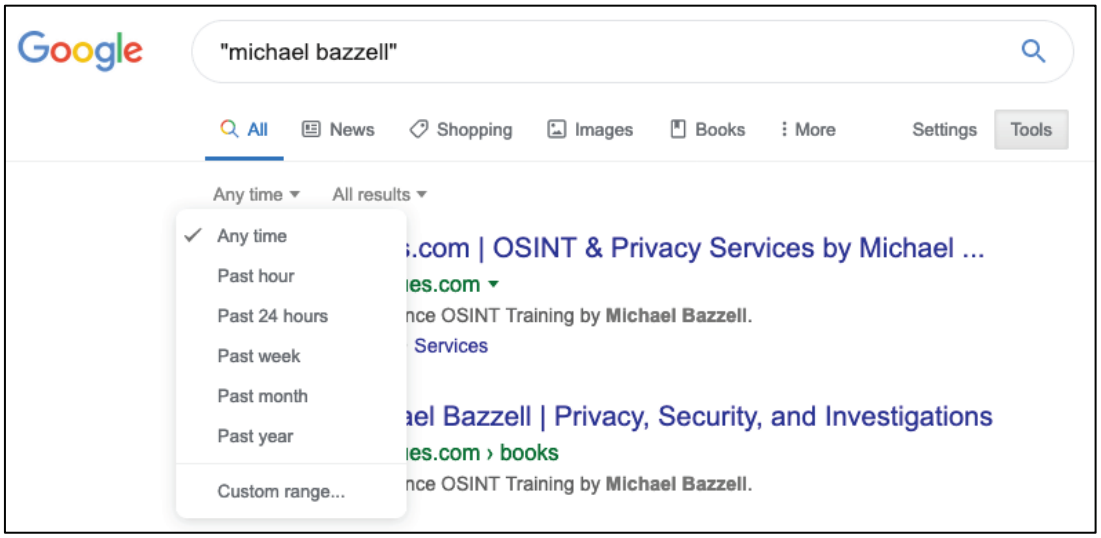


Figure 9.01: A Google Search Tools menu.

Dated Results

Google can be very sporadic when it comes to supplying date information within search results. Sometimes you will see the date that a search result was added to the Google index and sometimes you will not. This can be frustrating when you desire this information in order to identify relevant results. There is a fairly unknown technique that will force Google to always show you the date of each search result.

When you modify the "Any Time" option under the Search Tools menu, you will always see a date next to each result. If you are only searching for recent information, this solves the issue. However, if you are conducting a standard search without a specific date reference, the dates next to each result are missing. To remedy this, you

can conduct a specific search that includes any results indexed between January 1, 1 BC and "today". The appropriate way to do this is to add "&tbs=cdr:1,cd_min:1/1/0" at the end of any standard Google search. Figure 9.02 displays the results of a search for the term "Michael Bazzell". The exact URL of the search was as follows.

https://www.google.com/search?q="michael+bazzell"

Notice that the result does not include a date next to the item. Figure 9.03 displays the results of this same search with the specific data added at the end. The exact URL of this search was the following address.

https://www.google.com/search?q="michael+bazzell"&tbs=cdr:1,cd_min:1/1/0

Notice that the result now has the date when the content was first indexed by Google. You can also now sort these results by date in order to locate the most recent information. The search tools menu also offers an "All results" menu that will allow you to choose to see "all results" or "Verbatim". The All Results will conduct a standard Google search. The Verbatim option searches exactly what you typed. One benefit of the Verbatim option is that Google will often present more results than the standard search. It digs a little deeper and gives additional results based on the exact terms you provided.

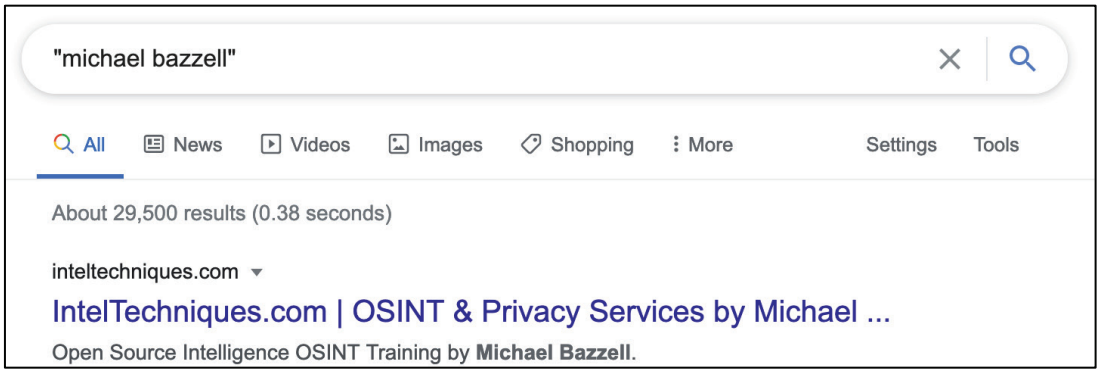


Figure 9.02: Google results without date injection.

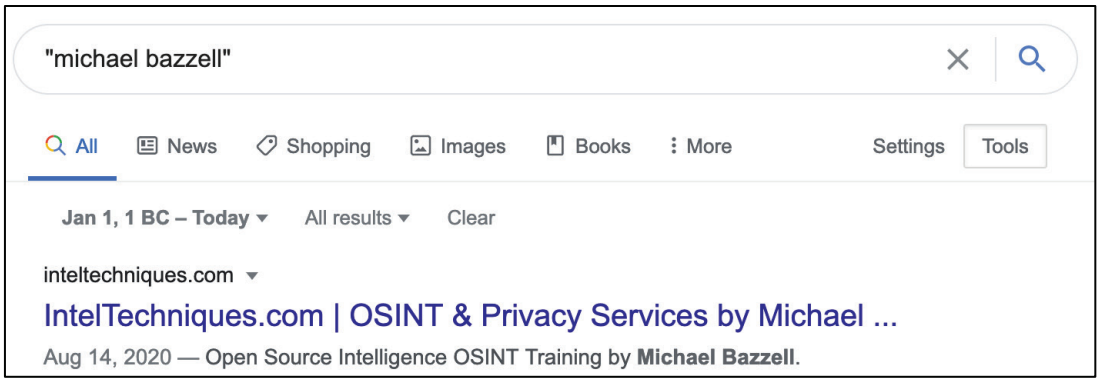


Figure 9.03: Results with date injection.

Google Programmable Search Engines (programmablesearchengine.google.com)

Now that you are ready to unleash the power of Google, you may want to consider creating your own custom search engines, which Google has rebranded to Programmable Search Engines. Google allows you to specify the exact type of searches that you want to conduct, and then create an individual search engine just for your needs. Many specialty websites that claim to search only social network content are simply using a custom engine

from Google. For our first example, we will create a basic custom search engine that only searches two specific websites.

After you log in to a Google account, navigate to the website listed above. If you have never created an engine, you will be prompted to create your first. Enter the first website that you want to search. In my example, I will search inteltechniques.com. As you enter any website to search, Google will automatically create another field to enter an additional website. The second website that I will search is inteltechniques.net. Provide a name for your custom engine and select "Create". You now have a custom search engine. You can either embed this search engine into a website or view the public URL to access it from within any web browser.

This basic functionality can be quite powerful. It is the method behind my custom Pastebin search engine discussed in a later chapter. In that example, I created a custom search engine that scoured dozens of specific websites in order to retrieve complete information about specific topics. This is only the first layer of a Google custom search engine. Google offers an additional element to its custom engines. This new layer, labeled Refinements, allows you to specify multiple actions within one custom search engine. The best way to explain this is to offer two unique examples.

For the first example, I wanted to create a custom search engine that allowed us to search several social networks. Additionally, we will isolate the results from each network across several tabs at the top of our search results. The first step will be to create a new custom search engine by clicking "New search engine" in the left menu. Instead of specifying the two websites mentioned earlier, we will identify the websites to be searched as the following.

Facebook.com	Twitter.com	Instagram.com
LinkedIn.com	YouTube.com	Tumblr.com

While this is not a complete list of active social networks, it represents the most popular social networks at the time of this writing. At this point, our custom search engine would search only these websites and provide all results integrated into one search result page. We now want to add refinements that will allow us to isolate the results from each social network.

After you have added these websites, provided a name, and created your engine, navigate to the control panel option in order to view the configuration of this custom search engine. On the left menu, you should see an option called "Edit search engine". Expanding this should present a list of your engines. Select your test engine and click "Search features". This will present a new option at the top of the page labeled "Refinements". Click the "add" button to add a new refinement for each of the websites in this example. You should create these in the same order that you want them to appear within the search results. For this demonstration, I created the following refinements in order, accepting the default options.

Facebook	Twitter	Instagram
LinkedIn	YouTube	Tumblr

When each refinement is created, you will have two options of how the search will be refined. The option of "Give priority to the sites with this label" will place emphasis on matching rules, but will also reach outside of the rule if minimal results are present. The second option of "Search only the sites with this label" will force Google to remain within the search request and not disclose other sites. I recommend using the second option for each refinement.

Now that you have the refinements made, you must assign them each to a website. Back on the "Setup" menu option, select each social network website to open the configuration menu. Select the drop-down menu titled "Label" and select the appropriate refinement. Repeat this process for each website and save your progress. You should now have a custom search engine that will not only search several specific social network websites, but it should also allow you to isolate the results for each network. Navigate back to "Setup" in the left menu and

select the Public URL link to see the exact address of your new engine. Go to that address and you should see a very plain search engine. You can now search any term or terms that you want and receive results for only the social networks that you specified. Additionally, you can choose to view all of the results combined or only the results of a specific network. Figure 9.04 displays the results when I searched the term osint. In this example, I have selected the Twitter refinement in order to only display results from twitter.com.

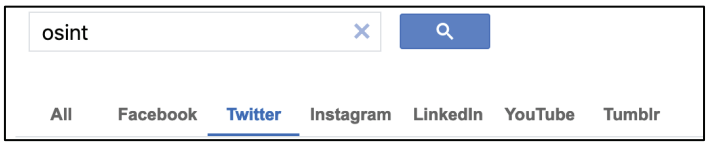


Figure 9.04: A Twitter refinement in a Google Custom Search.

You can now bookmark this new search engine that you created and visit it whenever you have a target to search. You can take your custom search engines to another level by adding refinements that are not website specific. In the next example, we will make a search engine that will search the entire internet and allow us to filter by file type.

Create a new custom search engine and title it "Documents". Add only "google.com" as the website to be searched. We do not actually want to search google.com, but a website is required to get to the edit panel. Save your engine, click "Edit search engine", and then click "Setup". In the "Sites to search" portion, enable the "Search the entire web" toggle. Delete google.com from the sites to be searched. You now basically have a custom search engine that will search everything. It will essentially do the same thing as Google's home page. You can now add refinements to filter your search results. Navigate to the search features menu and add a new refinement. Title the new refinement "PDF"; change the default setting to "Give priority to the sites with this label"; and enter the following in the "Optional word(s)" field.

ext:pdf

This will create a refinement that will allow you to isolate only PDF documents within any search that you conduct. Save this setting and create a new refinement. Title it DOC; change the default search setting; and place the following in the "Optional word(s)" field.

ext:doc OR ext:docx

This will create a new tab during your search results that will allow you to isolate Microsoft Word documents. By entering both the doc and docx formats, you will be sure to get older and newer documents. The word "OR" tells Google to search either format. Repeat this process for each of the following document types with the following language for each type.

XLS (Excel Spreadsheets) - ext:xls OR ext:xlsx OR ext:csv
PPT (PowerPoint Files) - ext:ppt OR ext:pptx
TXT (Text Docs) - ext:txt OR ext:rtf
WPD (Word Perfect Docs) - ext:wpd
ODT (OpenOffice Docs) - ext:odt OR ext:ods OR ext:odp
ZIP (Compressed Files) - ext:zip OR ext:rar OR ext:7z

Figure 9.05 displays the results of a search for the term osint within this new engine. The All tab is selected which reveals 717,000 results. Clicking the PowerPoint presentations option (PPT) reveals 45 files which contain the term. There are endless possibilities with this technique. You could make an engine that isolated images with extensions such as jpg, jpeg, png, bmp, gif, etc. You could also replicate all of this into a custom engine that only searched a specific website. If you were monitoring threats against your company, you could isolate only these files that appear on one or more of your company's domains.

Figure 9.05: A Documents File Type Google Custom Search.

One negative aspect to custom Google search engines is that they only display the most relevant 100 results. These are presented in ten pages of ten results per page. If you are searching very specific terms, this may not be an issue. However, standard searches can be limiting.

Google Alerts (google.com/alerts)

When you have exhausted the search options on search engines looking for a target, you will want to know if new content is posted. Checking Google results every week on the same target to see if anything new is out there will get mundane. Utilizing Google Alerts will put Google to work on locating new information. While logged in to any Google service, such as Gmail, create a new Google Alert and specify the search term, delivery options, and email address to which to send the alert. In one of my alerts, Google will send an email daily as it finds new websites that mention "Open Source Intelligence Techniques" anywhere in the site. Another one of my alerts is for my personal website. I now receive an email when another site mentions or links to my website. Parents can use this to be notified if their child is mentioned in a website or blog. Investigators that are continuously seeking information about a target will find this beneficial.

Real World Application: A police detective was assigned a runaway case where a 15-year-old had decided to leave home and stay with friends at an unknown location. After several extensive internet searches, a Google Alert was set up using the runaway's name and city of residence. Within three days, one of the alerts was for a blog identifying the runaway and where she was currently staying. Within 30 minutes, the unhappy child was back home.

Bing (bing.com)

Google is not the only great search engine. While Google is the overwhelming choice of search engines used today, other sites should not be ignored, especially when having trouble locating any information on a subject. Bing is Microsoft's competition to Google and provides a great search experience. In 2009, Yahoo search (yahoo.com) began using the Bing search engine to produce search results. This makes a Yahoo search redundant if a Bing search has already been conducted. The same tactics described previously, and in the numerous Google books, can be applied to any search engine. The site operator and the use of quotes both work with Bing exactly as they do with Google. Bing also introduced time filtered searching that will allow you to only show results from the last 24 hours, week, or month. There are a couple of additional operators that are important that only apply to Bing. Bing offers an option that will list every website to which a target website links, and is the only search engine that offers this service.

Bing LinkFromDomain

I conducted a search on Bing of "LinkFromDomain:inteltechniques.com". Note that there are no spaces in the entire search string and you should omit the quotation marks. This operator creates a result that includes every website to which I have a link, located on any of the pages within my website. This can be useful to an investigator. When a target's website is discovered, this site can be large and contain hundreds of pages, blog entries, etc. While clicking through all of these is possible, sometimes links are hidden and cannot be seen by visually looking at the pages. This operator allows Bing to quickly pull links out of the actual code of the website.

Bing Contains

Earlier, I discussed searching for files with specific file extensions on Google. The "filetype" and "ext" operators that were explained both work on Bing the same way. However, Bing offers one more option to the mix. The "contains" operator allows you to expand the parameters of the file type search. As an example, a Bing search of "filetype:ppt site:cisco.com" returns 13,200 results. These include PowerPoint files stored on the domain of cisco.com. However, these results do not necessarily include links on the cisco.com website to PowerPoint files stored on other websites. A search on Bing for "contains:ppt site:cisco.com" returns 36,200 results. These include PowerPoint files that are linked from pages on the domain of cisco.com, even if they are stored on other domains. This could include a page on cisco.com that links to a PowerPoint file on hp.com. In most cases, this search eliminates the need to conduct a filetype search, but both should be attempted.

Google Images (images.google.com)

Google Images scours the web for graphical images based on a search term. Google obtains these images based on keywords for the image. These keywords are taken from the filename of the image, the link text pointing to the image, and text adjacent to the image. This is never a complete listing of all images associated with a subject, and will almost always find images completely unrelated to a target. In the case of common names, one should enclose the name in quotes and follow it with the city where the subject resides, place of employment, home town, or personal interests. This will help filter the results to those more likely to be related to the subject. When results are displayed, clicking the "Tools" button will present five new filter menus. This menu will allow you to filter results to only include images of a specific size, color, time range, image type, or license type. The most beneficial feature of Google Images is the reverse image search option. This will be explained in great detail later in the book.

Bing Images (bing.com/images)

Similar to Google, Bing offers an excellent image search. Both sites autoload more images as you get toward the end of the current results. This eliminates the need to continue to load an additional page, and leads to faster browsing. Bing also offers the advanced options available on Google, and adds the ability to filter only files with a specified layout such as square or wide. Bing provides a "filter" option in the far right of results that provides extended functionality. The People tab offers restriction for images of "Just faces" and "Head & shoulders". It also provides suggested filters with every image search. Clicking image search links may provide additional photographs of the specific target based on the listed criteria. This intelligence can lead to additional searches of previously unknown affiliations.

International Search Engines

Search engines based in the U.S. are not the primary search sites for all countries. Visiting search sites outside of the U.S. can provide results that will not appear on Google or Bing. In Russia, Yandex is the chosen search engine. Yandex offers an English version at yandex.com. These results are often similar to Google's; however, they are usually prioritized differently. In the past, I have found unique intelligence from this site when Google let me down. In China, most people use Baidu. It does not offer an English version; however, the site is still usable. Striking the "enter" key on the keyboard after typing a search will conduct the search without the ability to understand the Chinese text. New results not visible on Google or Bing may be rare, but an occasional look on these sites is warranted.

Yandex (yandex.com)

In a previous edition of this book, I only made a brief reference to Yandex and quickly moved on. In the past few years, I have discovered many advanced features of Yandex which justify an expanded section. Visually, the Yandex home page and search results pages do not possess additional search operators. These options are only available by issuing a direct command within your search. While this can be more cumbersome than a Google

search, the results can include much new data. Some of these searches can be overkill for daily use, but those who conduct brand reputation monitoring or extensive background checks may take advantage of this.

Exact terms: Similar to Google and Bing, quotation marks will search for exact terms. Searching "Michael Bazzell" inside of quotes would search those terms, and would avoid "Mike" or "Bazel".

Missing word: You can search an exact phrase without knowing every word of the phrase. A search for "Open Source * Techniques" inside of quotation marks will identify any results that include that phrase with any word where the asterisk (*) is located. This identified not only results with the title of this book, but also results for "Open Source Development Techniques" and "Open Source Responsive Techniques". This search can be very useful for identifying a person's middle name. "Michael * Bazzell" produced some interesting results.

Words within the same sentence: The ampersand (&) is used in this query to indicate that you want to search for multiple terms. "Hedgehog & Flamingo", without the quotation marks, would identify any websites that contained both of those words within one sentence. If you want the results to only include sentences that have the two words near each other, you can search "Hedgehog /2 Flamingo". This will identify websites that have a sentence that includes the words Hedgehog and Flamingo within two words of each other.

Words within the same website: Similar to the previous method, this search identifies the searched terms within an entire website. "Hedgehog && Flamingo", without quotation marks, would identify pages that have both those words within the same page, but not necessarily the same sentence. You can also control the search to only include results that have those two words within a set number of sentences from each other. A search of "Hedgehog && /3 Flamingo", without the quotation marks, would identify websites that have those two words within three sentences of each other.

Include a specific word: In Google and Bing, you would place quotation marks around a word to identify pages that contain that word in them. In Yandex, this is gained with a plus sign (+). Michael +Bazzell would mandate that the page has the word Bazzell, but not necessarily Michael.

Search any word: In Google and Bing, you can use "OR" within a search to obtain results on any of the terms searched. In Yandex, this is achieved with the pipe symbol (|). This is found above the backslash (\) on your keyboard. A search of "+Bazzell Michael|Mike|M", without quotation marks, would return results for Michael Bazzell, Mike Bazzell, and M Bazzell.

Exclude a word: Google and Bing allow you to use a hyphen (-) to exclude a word in a search. Yandex does not technically support this, but it seems to work fine. The official Yandex operator is the tilde (~). A typical search would look like "Michael Bazzell ~ Mike", without the quotation marks. This would identify websites that contained Michael Bazzell, but not Mike Bazzell. I prefer to stick with the hyphen (-) until it no longer works.

Multiple identical words: This is a technique that I have needed several times in the past before I learned of Yandex's options. You may want to search for websites that contain a specific word more than once. An example might be if you are searching for someone that has two identical words in his or her full name. "Carina Abad Abad" would fit in this scenario. You could use quotation marks to identify the majority of the results, but you would filter out anything that was not exact such as Abad,Abad, Abad-Abad, or AbadAbad. This is where the exclamation point (!) comes in. A search of "!Carina !Abad !Abad", without quotation marks, would identify any results that included those three words regardless of spacing or punctuation.

Date specific searches: While Google provides a menu to filter your searches by date, Yandex makes you work harder for it. You must specify the date range within the search. The following queries should explain the options.

date:20111201..20111231 OSINT - Websites mentioning OSINT between December 1-31, 2011
date:2011* OSINT - Websites mentioning OSINT in the year 2011
date:201112* OSINT - Websites mentioning OSINT in December of 2011
date:>20111201 OSINT - Websites mentioning OSINT after December 1, 2011

Standard operators: Most of the operators explained earlier for Google and Bing should also work in Yandex. The commands for Site, Domain, Inurl, and Intitle should work the same way. Yandex maintains a list of operators at https://yandex.com/support/search/how-to-search/search-operators.html. All Yandex operators work together and multiple operators can be used to form very specific searches. Figure 9.06 displays the results for a search of any websites from 2013 with the phrase Michael Bazzell and the word OSINT while excluding the word Mike.

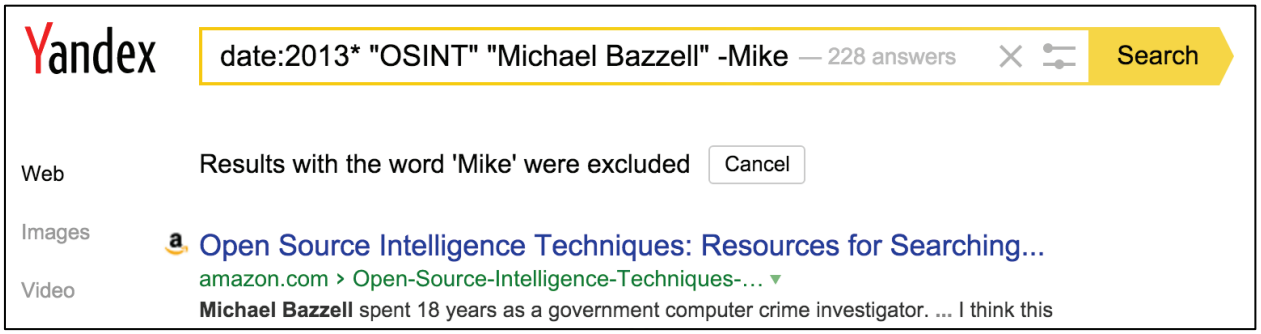


Figure 9.06: A custom Yandex search.

There are hundreds of additional international search engines. Of those, most are extremely specialized and do not offer great general search. The following have been most beneficial to my international investigations, in order of usefulness. I have included a direct search URL, which could be useful for your custom search tools.

Baidu http://www.baidu.com/s?wd=osint
Sogou https://www.sogou.com/web?query=osint
So https://www.so.com/s?q=osint
Mail.ru https://go.mail.ru/search?q=osint
Goo https://search.goo.ne.jp/web.jsp?MT=osint
Daum https://search.daum.net/search?w=tot&q=osint
Parseek http://parseek.com/Search/?q=osint
Parsijoo http://parsijoo.ir/web?q=osint
Naver https://search.naver.com/search.naver?query=osint
Coccoc https://coccoc.com/search?query=osint
Pipilika https://www.pipilika.com/search?q=osint
Seznam https://search.seznam.cz/?q=osint

I Search From (isearchfrom.com)

If you want to search Google within a version specified for another country, this site simplifies the process. Choose the country and language, and the tool will do the rest. While testing this service, I entered Japan as my country, English as my language, an iPad as my device, and OSINT as my search term. I was presented a google.co.jp search page in tablet view. Many results were similar to the U.S. version, but all were in a unique order. I find this useful when searching for international targets when I do not want bias toward a U.S. user. The "News" tab of foreign searches is often catered toward that geographical audience. This can display emphasis on news articles which would otherwise be buried in a typical Google result page.

Web Archives

Occasionally, you will try to access a site and the information you are looking for is no longer there. Maybe something was removed, amended, or maybe the whole page was permanently removed. Web archives, or "caches" can remedy this. I believe that these historical copies of websites are one of the most vital resources when conducting any type of online research. This section will explain the current options in order from most effective to least.

Google Cache (google.com)

When conducting a Google search, notice the result address directly below the link to the website. You will see a green down arrow that will present a menu when clicked. This menu will include a link titled "Cached". Clicking it will load a version of the page of interest from a previous date. Figure 9.07 (first image) displays a search for phonelosers.org which returns a result that includes a cached version of the page. This version was taken four days prior to the current date, and displays information different from the current version. The second option visible within this menu, titled "Similar", identifies web pages that contain content similar to the listed result.

If you have a specific page within a website that you want to view as a cached version, type the exact website into Google to link to the cached page. For example, if I wanted to see a previous view of the podcast for The Phone Show, an audio archive about telephone pranks, I would conduct a Google search for the site "www.phonelosers.org/snowplowshow". This will return the main landing page as well as sub-pages that will each have a cached view. If any of these pages were to go offline completely, Google would hold the last obtained version for viewing. I could have also typed the following directly into any Google search page to be navigated directly to the cached page.

cache:www.phonelosers.org/snowplowshow

Bing Cache (bing.com)

Similar to Google, Bing offers a cached view of many websites. Searching for a domain name, such as phonelosers.org, will present many results. The first result should link to the actual website. Directly next to the website name is a small green down arrow. Clicking it will present the option of "Cached page". Clicking this link will display a previous version of the target website as collected by Bing. Figure 9.07 (second image) displays their menu option.

Yandex Cache (yandex.com)

The Russian search engine Yandex will be explained in great detail later, but it is important to note now that it also possesses a cache option. Very similar to Google and Bing, Yandex presents a green drop-down menu directly under the title of the search result. Figure 9.07 (third image) displays their cache menu option. Selecting the Cached page option opens a new tab displaying the most recent Yandex archive of the page. The top banner displays the date and time of capture, the original website address, and a search option to highlight selected keywords within the result. The biggest strength of the Yandex cache is the lack of updates. While this may sound counterintuitive, an older cache can be very helpful in an investigation. Assume that the Phone Losers website was your target. At the time of this demonstration, September 7, 2019, the Google, Bing, and Yandex caches of this page were dated as follows.

Google: September 6, 2019
Bing: September 7, 2019
Yandex: September 1, 2019

Google and Bing tend to have very recent results which often appear identical to the live view. However, the Yandex option from a week prior is more likely to contain modified content. You can often locate a cached version of a page that is older than the Yandex version on Baidu.

Baidu Cache (baidu.com)

This Chinese search engine is the least productive as far as cached copies of websites are concerned, but it should not be ignored. It will be explained further during a later discussion about international engines. The results of a search on Baidu are mostly in Chinese, but can still be valuable to those that cannot read the text. At the bottom of each search result is a green link to the website that hosts the content of the result. While this also includes a drop-down menu, the cache option is not there. Instead, look for a word in Chinese directly to the right of this link. In Figure 9.07 (fourth image) it is displayed as 百度快照. Clicking this link will open a new tab with the cache result, which Baidu refers to as a snapshot. In my experience, the presence of this linked option does not always mean that a cached version exists.

The Wayback Machine (archive.org/web/web.php)

The Wayback Machine will provide a much more extensive list of options for viewing a website historically. Searching for phonelosers.org displayed a total of 1,280 captures of the site dating from 12/21/1997 through 6/10/2019 (Figure 9.08). Clicking the links presents quite a display of how the site has changed. Graphics are archived as well, proving that we should always think twice about which photos we post to the internet. Each view of the archived page will allow the user to click through the links as if it were a live page on the original web server. Clicking through the timeline at the top of each page will load the viewed page as it appeared on the date selected.

Wayback Search

Until 2016, you could not search keywords across Wayback Machine data. You had to know the exact URL of a target website, or at least the domain name. Today, we can search any terms desired and connect directly to archived data. At the time of this writing, a search bar was present at the top of every Wayback Machine page. If that should change, you can also conduct a search via a direct URL. The following address searched "Michael Bazzell" throughout the entire archive of information.

https://web.archive.org/web/*/Michael Bazzell

The results identify over twenty websites that include these terms. Within those sites are dozens of archived copies of each. This data represents decades of content at your fingertips. Much of it is offline and unavailable on the current public internet. Many domains have completely shut down. Furthermore, websites that I own appear within the results, even though I have specifically blocked archiving them through a configuration file on my server. You would not find these by searching the domains directly through the Wayback Machine. This is a reminder that we should check all available resources before completing our investigations.

Searching All Resources

Occasionally, there are websites that surface claiming to be able to extract and rebuild entire websites from online caches. In my experience, none of these have ever provided a complete historical view versus a manual approach. Engines such as Bing and Yandex generate a unique code when a cache is displayed. This action prevents most automated search tools from collecting archived information. I do not believe any option, other than navigating to each resource, will present you with the content that you need. I bookmark each of these services in an individual folder titled Archives and open each tab when I have a domain as a target. I have also created an online tool that will collect your target domain and forward you to the appropriate archive page. This will be explained later when discussing domain searches.

Finally, it is important to acknowledge that these resources can be beneficial when everything on a website appears to be present and unaltered. While caches work well on websites that have been removed and are completely empty, they also can tell a different story about websites that appear normal. Any time that I find a website, profile, or blog of interest, I immediately look at caches hoping to identify changes in content. These minor alterations can be very important. They highlight information that was meant to be deleted forever. These details can be the vital piece of your investigation puzzle. Most people have no idea that this technique exists.

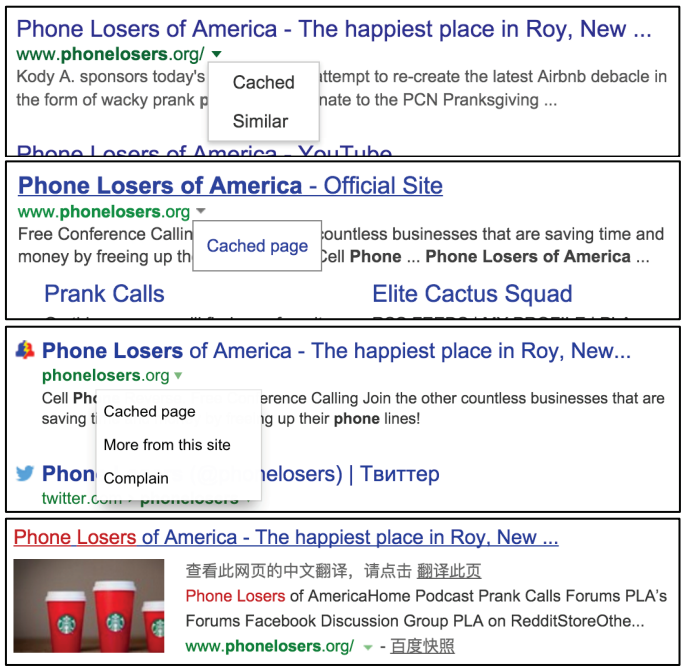


Figure 9.07: Cache menu options on Google, Bing, Yandex, and Baidu.

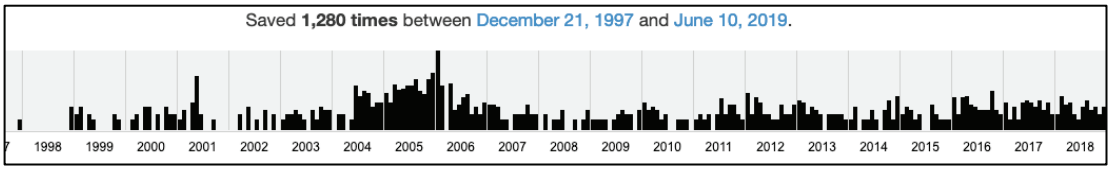


Figure 9.08: Wayback Machine results for an archived website.

Non-English Results

Not every piece of information that will be useful to you will be obtained by standard searches within English websites. Your target may either be from another country or have associates and affiliations in another country. While Google and Bing try to pick up on this, the technology is not perfect. Google has a search site and algorithm that change by location. For example, google.fr presents the French search page for Google. While this may produce the same overall results, they are usually in a different order than on google.com. Google no longer maintains a page with links to each international version of its search, but I have a preferred method.

2Lingual (2lingual.com)

This page will allow you to conduct one search across two country sites on Google. The Google search will display a plain search box and choices of two countries. The results will display in single columns next to each other. Additionally, the foreign results will be automatically translated to English. This feature can be disabled, if desired. The first few sponsored results (ads) will be similar, but the official results following should differ.

This site can also be helpful when demonstrating to someone the importance of searching targets through multiple countries.

Google Translator (translate.google.com)

Many websites exist in non-English languages. As internet enthusiasts, we tend to focus on sites within our home area. There is a wealth of information out there on sites hosted in other countries which are presented in other languages. Google Translator will take text from any site or document and translate the text to a variety of languages. Usually, the service will automatically identify the language of the copied and pasted text. Selecting the desired output will provide the translation. Alternatively, you can translate an entire website in one click which will give a native view of the layout of the site. Instead of copying individual text to the search box, type or paste in the exact URL (address) of the website you want translated. Clicking the "Translate" button will load a new page of the site, which will be translated to English. This translation is rarely, if ever, perfect. However, it should give you an idea of the content presented on the page. This will also work on social network sites such as Twitter and Instagram.

Bing Translator (bing.com/translator)

A few years after Google introduced free translation services, Bing created their own product. At first glance, it looks like a replica of Google's offering. However, Bing's translations are usually slightly different than Google's results. Similar to Google, you can also type or paste an entire foreign website to conduct a translation of everything on the target page.

DeepL (deepl.com/translator)

While smaller than Google or Bing, this may be the most accurate translator service I have found. The page appears and functions identical to the previous options, but the results may be substantially different.

PROMT Online Translator (online-translator.com)

There are dozens of additional online translation tools available. Almost all of them allow translation of a small amount of text at a time. Some use either the Google or Bing translation service. One last online translation tool worth mentioning is PROMT Online Translator. It is unique from the dozens of other options in that it allows translation of entire websites similar to Google and Bing. This service provides an independent translation and can be considered a third source.

I am often asked during training which of the services I use during investigations. My answer is all of them. This is important for two reasons. The obvious benefit is that you will receive four unique translations that will be very similar. The minor variations may be important, especially when translating Tweets and other shortened messages that may not be grammatically correct in any language. The second reason is to show due diligence during my investigation. I always want to go above and beyond what is required. Translating a foreign web page through four different services emphasizes my desire to conduct an unbiased investigation.

Google Input Tools (google.com/inputtools/try)

There is one last feature regarding foreign language searching that I have found useful. Google's Input Tools allow you to type in any language you choose. Upon navigating to the above website, choose the language of your target search. In Figure 9.09, I have chosen Arabic as the language and typed "Online Investigation" on a standard English keyboard. The result is how that text might appear in traditional Arabic letters. I have had the most success with this technique on Twitter. When supplying any search term on Twitter, the results are filtered by the presence of the keywords entered and only in the language provided. Searching "Online Investigation" on Twitter only provides results that have that exact spelling in English characters. However, searching the Arabic output provides Tweets that include the Arabic spelling of the selected words. This technique is extremely

important when you have located a username in a foreign language. As with all computer-generated translation services, the results are never absolutely accurate. I expect this technology to continue to improve.

Figure 9.09: A Google Input Tools translation from English to Arabic.

Google News Archive (news.google.com)

This can be an amazing resource of information about a target. In the past, if someone relocated to a new geographical area, he or she could leave the past behind and start over. Today, that is difficult. Google's News Archive is continually adding content from both online archives and digitized content from their News Archive Partner Program. Sources include newspapers from large cities, small towns, and anything in between. The link referenced above will allow for a detailed search of a target's name with filters including dates, language, and specific publication. In order to display this menu, click on the down arrow to the right of the search box. This can quickly identify some history of a target such as previous living locations, family members through obituaries, and associates through events, awards, or organizations.

Google Newspaper Archive (news.google.com/newspapers)

The previous option focused solely on digital content, such as your local newspaper website. Google's Newspaper archive possesses content from printed newspapers. All results on this site consist of high-resolution scanned newspaper pages. In my experience, this collection is not as extensive as the next option discussed. However, it is definitely worth a look, and will likely continue to grow.

Newspaper Archive (newspaperarchive.com)

This paid service provides the world's largest collection of newspaper archives. The high-resolution PDF scans of entire daily newspapers range in date from the 1800's until present. The first four editions of this book explained a method of using the Google Site operator and cached results to obtain practically any page of this newspaper collection without paying or subscribing. These vulnerabilities have all been patched and none of those techniques work today. Fortunately, Newspaper Archive still offers a 14-day free trial with unlimited access to every archive. While multiple trials can be obtained, each require a unique credit card number and email address. Many libraries have asked this service to scan their entire microfilm archives and make them freely available online. You will not find any mention of this free alternative on their home page, but a bit of searching will guide you to the right place. The following search on Google identifies hundreds of public libraries that pay for your access to their archives.

site:newspaperarchive.com "This archive is hosted by" "create free account"

The first part of the search tells Google to only look at the website newspaperarchive.com. The second part mandates that the exact phrase "This archive is hosted by" appears in the result. The final piece isolates only the newspaper collections that are available for free and without a credit card. This identifies the landing pages of the various libraries that have made their collections freely available. While you will still be required to register through the service, payment is not required for these collections. Consider the following usage that will likely present you with free views of Newspaper Archive whenever you need them.

On 12/13/2017, I navigated to newspaperarchive.com/advancedsearch/ and conducted an advanced search for anyone named Michael Williams from Cedar Rapids, Iowa. Newspaper Archive presented several results from the Cedar Rapids Gazette. Clicking on any of these results prompted me to create an account and forced me to

enter a valid credit card number to proceed. I could not create an account from any of the pages without providing payment. Instead, I conducted the following Google search.

site:newspaperarchive.com "This archive is hosted by" "cedar rapids gazette"

The first result was a direct connection to crpubliclibrary.newspaperarchive.com. Clicking this link presented a page dedicated to searching over 40 newspapers within the Cedar Rapids and Des Moines areas. In the upper right corner was a link titled "Create Free Account". I clicked this link and provided generic details and a throwaway email address. The membership choices now include a completely free option, which will only allow access to the Iowa newspapers. After creating my free online account, I returned to the portal at crpubliclibrary.newspaperarchive.com and repeated the search of my target. Every link allowed me full unrestricted access to the high-resolution images.

While still logged in to this account, I navigated to delawarecolib.newspaperarchive.com, the direct page associated with the Delaware County Library (which I found through the original Google search in this section). I was not authorized to view this newspaper collection. However, after clicking "Create Free Account" on this page, I entered the same data as previously provided to the Iowa newspaper. After verifying my email address, I was allowed immediate access to this series of newspapers.

This technique will not obtain access to every collection on Newspaper Archive. However, it will provide a surprising amount of free access to huge collections internationally. During an hour of downtime, I created a free account on every library collection I could locate, using the same credentials on each. I can now log in to my single Newspaper Archive account and navigate the site from any page. When I reach a newspaper of interest after a search, I will be given full access if it is within a free collection. This is all thanks to the local libraries that have paid this site to give free access to the public. If the free trial of Newspaper Archive or the free library collections do not offer enough content, consider the following options.

Old Fulton (fultonhistory.com/Fulton.html): 34,000,000 scanned newspapers from the United States and Canada.

Library of Congress US News Directory (chroniclingamerica.loc.gov):
Scanned newspapers from the United States dated 1836-1922.

Library of Congress US News Directory (chroniclingamerica.loc.gov/search/titles):
Scanned newspapers from the United States dated 1690-Present.

Small Town Newspapers (stparchive.com):
Scanned and text versions of small town newspapers since 1890

Note that the search features on all of these options are mediochre at best. Always consider a Google search, such as site:stparchive "michael bazzell".

Google Advanced Search (google.com/advanced_search)

If the search operators discussed in this chapter seem too technical, Google offers an advanced search page that simplifies the process. Navigating to the above website will present the same options in a web page that are possible by typing out the operators. This will help you get familiar with the options, but it will be beneficial to understand the operators for later use. The Advanced Search page will allow you to specify a phrase for which you are searching, just like the quotes in a search will allow. The site and filetype operators used earlier can be achieved by entering the desired filters on this page. It should be noted that the file type option on this page is limited to popular file types, whereas the filetype operator can handle many other file extensions.

Bing Advanced Search (search.yahoo.com/web/advanced)

Bing does not technically provide an advanced search page similar to Google's. However, since Yahoo uses Bing's search, you can use Yahoo's advanced search page as a replacement. This page will allow you to easily create a search that filters by individual terms, exact phrases, omitted terms, specific domains, file formats, and languages.

Additional Google Engines

Google isolates some search results into specialized smaller search engines. Each of these focuses on a unique type of internet search. The following engines will likely give you results that you will not find during a standard Google or Bing search. While some results from these unique searches will appear within standard Google results, the majority will be hidden from the main page.

Google Blogs (google.com)

Google removed its original blog search in 2014. It was quite helpful and focused mostly on personal websites, especially those with a blogging platform. Today, this is nothing more than a subsection of Google News. You can load the "Blogs" option under the " News" menu within the "Tools" option on any Google News results page. Alternatively, you can navigate to the following address, replacing TEST with your search terms.

google.com/search?q=TEST&tbm=nws&tbs=nrt:b

The website above displays a standard Google search option, but the results appear much differently. A standard Google search of my name reveals my website, Twitter, and Amazon pages in the first results. The Google Blogs option reveals several personal and professional (media) blogs that mention my name. These results are likely buried within the standard Google search.

Google Patents (google.com/?tbm=pts)

Google probably has the best patent search option on the internet. It allows you to search the entire patent database within any field of a patent. This can be useful for searching names associated with patents or any details within the patent itself. If you need further help, Google offers an advanced patent search at google.com/advanced_patent_search.

Google Scholar (scholar.google.com)

Google Scholar is a freely accessible web search engine that indexes the full text of scholarly literature across an array of publishing formats. It includes most peer-reviewed online journals of Europe's and America's largest scholarly publishers, plus many books and other non-peer reviewed journals. My favorite feature of this utility is the case law and court records search. I have located many court records through this free website that would have cost money to obtain from private services.

Keyword Tool (keywordtool.io)

Keyword Tool displays autocomplete data from Google, Bing, YouTube, and the App Store. You have likely noticed that Google quickly offers suggestions as you type in your search. This is called autocomplete. If I were to type "macb" into Google, it would prompt me to choose from the most popular searches when people typed those letters. This information may lead you to new terms to search in reference to your investigation. The advantage of Keyword Tool over Google is that Google only provides the five most popular entries. Keyword Tool provides the ten most popular entries. Additionally, you can choose different countries to isolate popular terms. You can also see results from similar searches that Google does not display.

Real World Application: I have successfully used this technique during the investigation of many businesses. I was once asked by a medium-sized business to investigate reports of a faulty product that they had recently recalled. They wanted to see customer complaints. After searching the typical review websites, I conducted a search with Keyword Tool. I discovered that the 9th most popular search involving this specific product name included a term that was a misspelling of the product name. It was different enough in spelling that my searches were missing this content. Knowing this information, I was able to locate more relevant data for the client.

This can also be very valuable for marketing and promotion. Assume I want to know what additional terms people search when they start with the word osint. Maybe I want to buy Google ads or tweak my website to be noticed more often. With this tool, I now know that the following are the most popular osint-related searches on Google.

osint meaning
osint websites
osint techniques
osint training

The YouTube tab tells me that people are searching for videos related to the following terms.

osint tools
osint investigations
osint phone number
osint analysis

Finally, I see that Bing users seem to be a bit more focused with the following queries.

osint resources
osint api
osint mind map
osint michael bazzell

Other Alternatives

Google and Bing are great, but they do not do it all. There will always be a need for specialized search engines. These engines usually excel in one particular search method which justifies the lack of search power in other areas. The sites listed in this next section represent the extreme minority when it comes to search traffic. It is often sites like these that implement the technologies that we later take for granted in more popular engines.

Searx (searx.be)

This is considered a meta-crawler, as it presents results from Google, Bing, and others. It often gets dismissed as another comparison search site, but there are many other advantages to using this service. First, conducting a search will provide results from the main search engines, but will remove duplicate entries. This alone is a quick way to conduct your due-diligence by checking Google and Bing. Next, the top row of options will allow you to repeat this redundancy-reducing option by checking results on Images, News, and Videos sections. Next to each result on any search page is a "cached" link. Instead of opening the Google or Bing cache, clicking this will open the cached page of the target website through the Wayback Machine. Finally, a "proxied" option next to each result will connect you to the target website through a proxy service provided by Searx. This is basically a layer of privacy preventing the website owner from collecting data about you, such as your IP address. Technically, Searx.me opened the target site, and their data would be tracked instead of yours. There are ways for adversaries to bypass this "anonymity", but it is decent protection for most sites.

The final benefit of this service over all others is the easy ability to export search results as a file. The "Links" section to the right of all search pages displays options to download a csv, json, or rss file of the results. The csv option is a simple spreadsheet that possesses all of the search results with descriptions and direct links. I find this helpful when I have many searches to conduct in a short amount of time, and I do not have the ability to analyze the results until later.

Exalead (exalead.com/search)

Headquartered in Paris, this search engine has gained a lot of popularity in the United States. The main search engine provides many results on popular searches. I have found that individual targets without a strong internet presence do not get many, if any, results on this site. However, this site excels in two areas. It works well in finding documents that include the target mentioned within the document. The "filetype" operator used in other engines works the same here. Voxalead, an Exalead search engine, searches within audio and video files for specific words. This is thanks to speech to text technologies. Voxalead will search within all of the spoken audio of a file for references to the text searched. The results are presented in a timeline view. Currently, the majority of the results of this new product link to news media and public news video files.

DuckDuckGo (duckduckgo.com)

This search engine with a clean interface offers two unique services. It has gained a lot of popularity because it does not track anything from users. Engines, such as Google, record and maintain all of your search history and sites visited. This can be a concern to privacy advocates and those with sensitive investigations. Additionally, it uses information from crowd-sourced websites such as Wikipedia and Wolfram Alpha to augment traditional results and improve relevance. You will receive fewer results here than at more popular search engines, but the accuracy of the results will improve.

Start Page (startpage.com)

Similar to DuckDuckGo, Start Page is a privacy-focused search engine that does not reveal your connection information to traditional search engines. The difference here is that Start Page only includes Google results versus DuckDuckGo's collaboration of multiple sources. The benefit to this is the ability to use Google's advanced search options while still protecting your identity. This includes filtering by date, images, and videos. Another benefit is the ability to open any result through a "proxy" link. This option, labeled "Proxy" next to each search result, opens the linked page through Start Page's servers and displays the content within their site. This protects your IP address from anyone monitoring connections at the target website. While this technique is not foolproof, it provides a valid layer of protection. My search strategy involves Start Page whenever I have a sensitive search that I do not want to associate with my computer or internet connection. This might include investigations that involve highly sensitive topics such as tech-savvy stalker suspects.

Qwant (qwant.com)

Qwant attempts to combine the results of several types of search engines into one page. It was launched in 2013 after two years of research. It has an easily digestible interface that displays results in columns titled Web, News, Images, Videos, Maps, and Music. There is a Google "feel" to it and the layout can be changed to your own preferences. A default search of my own name provided the expected results similar to Google and Bing. Clicking on the tabs at the top introduced new results not found on the other engines. The results included recent posts from Twitter, Facebook, LinkedIn, and Myspace from and about people with my name.

Million Short (millionshort.com)

This website offers a unique function that is not found on any other search engine. You can choose to remove results that link to the most popular one million websites. This will eliminate popular results and focus on lesser-known websites. You can also select to remove the top 100,000, 10,000, 1,000, or 100 results.

Tor Search Engines

Tor is free software for enabling anonymous communication. The name is an acronym derived from the original software project name The Onion Router. Tor directs internet traffic through a free, worldwide, volunteer network consisting of more than six thousand relays to conceal a user's location and usage from anyone conducting network surveillance or traffic analysis. Using Tor makes it more difficult for internet activity to be traced back to the user. This also applies to a website that is hosted on the Tor network. Usually, these sites include illegal drug shops, child pornography swaps, and weapon sales. Because these sites are not hosted on publicly viewable networks, they are hard to locate and connect. Tor-based search engines and a proxy aid this process.

Ahmia (ahmia.fi)

This is a very powerful Tor search engine. While no engine can index and locate every Tor website, this is the most thorough option that I have seen. It should be the first engine used when searching Tor related sites. The links in the results will not load if searching through a standard browser and connection. Using the Tor Browser discussed previously is the ideal way to use this service.

Tor Link (tor.link)

This engine appeared in 2022 and is quite promising. When I conducted a search of "OSINT" within Ahmia, I received 5 results. The same query Tor Link revealed more results. It appears to index Tor sites well, and I received many results when querying email addresses of targets. This has complemented Ahmia well for many of my Tor-based investigations.

Onionland Search (onionlandsearchengine.com)

This service relies on Google's indexing of Tor sites which possess URL proxy links. However, I find some "hidden" sites with this utility which were not presented by the previous options.

Tor2Web (www.tor2web.org / onion.ly)

Whenever you see a URL like libertygb2nyeyay.onion, it is a Tor Onion website. As mentioned earlier, you cannot connect directly to these links without being connected to the Tor network. However, you can replace ".onion" within the address to ".onion.ly" in order to view the content. In the above example, navigating to the website libertygb2nyeyay.onion.ly will display the content using the Tor2Web proxies. This connects you with Tor2web, which then talks to the onion service via Tor, and relays the response back. This is helpful when locating Tor links on Twitter.

Tor Search Sites

I believe some of the strongest Tor search engines exist only on the Tor network. You cannot access them from a standard internet connection, and the Tor Browser is required for native use. My favorite is "Torch", which can be found at the following address if connected to Tor.

http://torch4st4l57l2u2vr5wqwvwyueucvnrao4xajqr2klmcmicrv7ccaad.onion

Since Tor2Web allows us to use their proxy, we can connect to "Torch" by navigating directly to the following Tor2Web proxy address, without being on the Tor Browser.

http://torch4st4l57l2u2vr5wqwvwyueucvnrao4xajqr2klmcmicrv7ccaad.onion.ly

This presents the home page of the search site, and allows for a keyword search. However, searching through this portal while being connected through the Tor2Web proxy can present difficulties. Instead, consider conducting a search within a URL submission. In the following web address, I am connecting to Tor2Web's proxy of the search engine and requesting a search results page for the term OSINT.

http://torch4st4l57l2u2vr5wqwvwyueucvnrao4xajqr2klmcmicrv7ccaad.onion.ly/index.php?q=OSINT

This type of submission will be much more reliable than counting on the proxy to conduct your search and return an additional proxy-delivered page. An alternative to Torch is **Haystack**, which can be accessed at http://haystak5njsmn2hqkewecpaxetahtwhsbsa64jom2k22z5afxhnpxfid.onion. Similar to the previous query, we can search this Tor service without the Tor browser with the following standard URL.

http:// haystak5njsmn2hqkewecpaxetahtwhsbsa64jom2k22z5afxhnpxfid.onion.ly/?q=osint

I have also had limited success with Tor66 at the following URL.

http://tor66sewebgixwhcqfnp5inzp5x5uohhdy3kvtnyfxc2e5mxiuh34iid.onion

All of the options presented here, and several others, are available for automatic queries within your custom search tool, as presented in a moment. Please note that these sites appear, disappear, and reappear often.

Search Engine Collections

I believe I could fill several chapters with tutorials for the hundreds of search engines available today. Instead, I point you toward two of the best collections I have found.

Search Engine Colossus (searchenginecolossus.com)

This website is an index of practically every search engine in every country. The main page offers a list of countries alphabetically. Each of these links connects to a list of active search engines in that country. I stay away from this service when searching American-based subjects. However, if my target has strong ties to a specific country, I always research the engines that are used in that area through this website.

Fagan Finder (faganfinder.com)

This website offers an interactive search page which populates your query into hundreds of options. Enter your search term in the field and click any of the hundreds of buttons to begin a search. Many of the search services are targeted toward niche uses, but you may find something valuable there which you do not see on the custom offline search tool provided at the end of this chapter.

FTP Search

I believe that the searching of File Transfer Protocol (FTP) servers is one of the biggest areas of the internet that is missed by most online researchers. FTP servers are computers with a public IP address used to store files. While these can be secured with mandated access credentials, this is rarely the case. Most are public and can be accessed from within a web browser. The overall use of FTP to transfer files is minimal compared to a decade ago, but the servers still exist in abundance. I prefer the manual method of searching Google for FTP information. As mentioned earlier, Google and Bing index most publicly available data on FTP servers. A

custom search string will be required in order to filter unwanted information. If I were looking for any files including the term "confidential" in the title, I would conduct the following search on Google and Bing.

inurl:ftp -inurl (http|https) "confidential"

The result will include only files from ftp servers (inurl:ftp); will exclude any web pages (-inurl: (http|https); and mandate that the term "confidential" is present (""). I have located many sensitive documents from target companies with this query. The above search yielded 107,000 FTP results. However, these specific hits are not the only valuable data to pursue. Consider the following example. I want to locate PDF documents stored on FTP servers that contain "cisco" within the title or content, and I conduct the following search on Google.

inurl:ftp -inurl (http|https) "cisco" filetype:pdf

This results in 20,000 options within multiple FTP servers hosted on numerous domains. The first result is hosted on the Southwest Cyberport FTP server and connects to a PDF document at the following address. It appears to be a chapter of a textbook.

ftp://ftp.swcp.com/pub/cisco/03chap01.pdf

Manually changing the last "01" to "02" loads the second chapter of the book. However, it is easier to eliminate the document name altogether and browse the directory titled "cisco". The first of the following addresses displays the contents of that folder, while the second displays the content of the "pub" folder. Copy these directly into a web browser to see the results.

ftp://ftp.swcp.com/pub/cisco/
ftp://ftp.swcp.com/pub/

This type of manual navigation will often reveal numerous publicly available documents that traditional searches withhold. I have located extremely sensitive files hosted by companies, government agencies, and the military. Most File Transfer Protocol (FTP) servers have been indexed by Google, but there are other third-party options that are worth exploring. At the end of each description, I identify the number of results included for the search "Cisco" "PDF".

Napalm FTP (searchftps.org)

This FTP search engine often provides content that is very recent. After each result, it displays the date that the data was last confirmed at the disclosed location. This can help locate relevant information that is still present on a server. While it generated the most results of all four services, many of them were no longer available on the target FTP servers. Some could be reconstructed with cached copies, but not all.

"Cisco" "PDF": 3,384

Mamoht (mmnt.ru)

This Russian FTP server allows you to isolate search results by the country that is hosting the content. This is likely determined by IP address. While most of the filtered results will be accurate, I recommend searching through the global results before dismissing any foreign options. My favorite feature of this engine is the "Search within results" option. After conducting my search, I checked this option and my search field was cleared. I entered "router" and clicked search again. I was prompted with the 436 results within my original hits that also included the word router. While this could have been replicated manually, I appreciate the option.

"Cisco" "PDF": 789

For comparison, Google found 19,600 results for inurl:ftp -inurl:(http|https) "Cisco" "PDF".

FreewareWeb (freewareweb.com/ftpsearch.shtml)

This service is less robust than the previous options, but it should not be ignored. In 2022, I was searching for documents with a unique file extension. This was the only service which presented the documents I needed, which were contracts signed by my target.

Nerdy Data (nerdydata.com/reports/new)

Google, Bing, and other search engines search the content of websites. They focus on the data that is visually present within a web page. Nerdy Data searches the programming code of a website. This code is often not visible to the end user and exists within the HTML code, JavaScript, and CSS files with which most users are not familiar. This code can be extremely valuable to research in some scenarios. Viewing the source code of a website can be done by right-clicking on the background of a page and selecting "View Source". The following two examples should explain a small portion of the possibilities with this service.

In later chapters, you will learn about free services that try to identify additional websites that may be associated with your target website. The backbone of these services relies on the indexing of programming data of websites. Nerdy Data may be the purest way of searching for this data. If you were to look at the source code of one of my previous websites (no longer online), you would have seen at the bottom that I used a service called Google Analytics. This service identifies the number of visitors to a website and the general area where they are located. The following is the actual code that was present.

```
<script type="text/javascript">
try {var pageTracker = _gat._getTracker("UA-8231004-3");
pageTracker._trackPageview();
} catch(err) {}</script>
```

The important data here is the "UA-8231004-3". That was my unique number for Google Analytics. Any website with which I used the service would have needed to have that number within the source code of the page. If you searched that number on Nerdy Data a few years prior, you would have received interesting results. Nerdy Data previously identified three websites that were using that number, including computercrimeinfo.com and two additional sites that I maintained for a law firm. You can often find valuable information within the source code of your target's website.

Many web designers and programmers steal code from other websites. In the past, this would be very difficult to identify without already knowing the suspect website. With Nerdy Data, you can perform a search of the code of concern and identify websites that possess the data within their own source code. In 2013, I located a custom search website at the YGN Ethical Hacker Group that inspired me to create my own similar search service. I was curious if there were any other search websites that possessed this basic code that might give me more ideas. I looked at the source code of the website and located a small piece of code that appeared fairly unique to that service. I conducted a search on Nerdy Data for the following code.

```
<li>http://yehg.net/q?[keyword]&c=[category]  (q?yehg.net&c=Recon)</li>
```

This code was within the JavaScript programming of the search website. The search results identified 13 websites that also possessed the same code. Two of these results were hosted on the creator's website, and offered no additional information. Three of the results linked to pages that were no longer available. Three of the results linked to pages that were only discussing the code within the target website and how to improve the functionality. However, four of the results identified similar search services that were also using the programming code searched. This revealed new search services that were related to the website in which I was interested.

This same technique could be used to identify websites that are stealing proprietary code; locate pages that were created to attempt to fool a victim into using a cloned site; or validate the popularity of a specific programming function being used on hacking websites globally.

IntelTechniques Search Engines Tool

At this point, you may be overwhelmed with the abundance of search options. I can relate to that, and I do not take advantage of every option during every investigation. During my initial search of a target, I like to rely on the basics. I first search Google, Bing, Yandex, and the smaller search engines. In order to assist with this initial search, I created a custom tool that will allow you to quickly get to the basics. Figure 9.10 displays the current state of this tool available in your downloads or at **https://inteltechniques.com/tools**. The search options will allow you to individually search directly through Google, Bing, Yahoo, Searx, Yandex, and many others. Across all options, each search that you conduct will open within a new tab within your browser. The search all takes place on your computer within your browser, directly to the sources.

The "Submit All" option will allow you to provide any search term that will be searched across all of the services listed. Each service will populate the results within a new tab in your internet browser. Regardless of the browser that you use, you must allow pop-ups in order for the tool to work. You can also use any of the search operators discussed previously within this tool, including quotation marks. I present a similar search tool at the end of most chapters which summarizes and simplifies the query processes for the techniques explained. I encourage you to become familiar with each of these. Once proficient, you can query target data across all options within a few minutes. This saves me several hours every week.

Figure 9.10: The IntelTechniques Search Engines Tool.

CHAPTER TEN
SOCIAL NETWORKS: FACEBOOK

I hesitate writing anything within this chapter. It seems that most valuable Facebook search techniques disappeared in 2019. There are still many methods we can apply, but the future outlook for targeted queries is dim. I worry this chapter will become outdated before anything else, but there are some stable searches which should have longevity. Before we dive in, let's take a look at the Facebook search issues timeline.

June 6th, 2019: This was the big date which began the decline of the Facebook graph. Previous profile queries all failed, and everything seemed doomed. Our ability to view "hidden" content via URL modifications was gone and we all scrambled for new techniques.

June 17th, 2019: Various researchers developed online search tools and browser extensions which brought back most of the Facebook Graph functionality. Hundreds of OSINT researchers flocked to these and we restored our missing techniques.

August 1st, 2019: On this date, all browser-based extensions which leveraged the Facebook Graph stopped working. Facebook implemented new encryption which terminated all functionality.

August 2nd, 2019: The Facebook username to user ID conversion tool on my website stopped working. It appeared that my web server was being blocked by Facebook. I switched the user ID conversion tool to a new web server, and all appeared to be working again.

August 3rd, 2019, 12:38 pm ET: Facebook began blocking my new web server.

September 8th, 2019: On this date, we saw methods such as using the Facebook Messenger application to search telephone numbers disappear, as well as most email search options. This appeared deliberate, and more evidence of Facebook's desire to lock down the platform.

2020: Facebook drastically changed their layout, removed the wildcard (*) search operator, blocked some of the "base64" methods (explained in a moment), and continued to aggressively monitor the OSINT community's response to their actions.

2022: Many search methods have disappeared, reappeared, and disappeared again. You have been warned that some of the techniques here may no longer work by the time you read this. However, we still have many options!

Fortunately, many people still share intimate details of their lives within social networks such as Facebook. Information that was once held privately within a small group of friends or family is now broadcasted to the world via public profiles. This chapter should identify new techniques which can be applied to any Facebook target. It will explain numerous ways to obtain user information which is public data, even if not visible within their official profile.

Official Facebook Options

Facebook's redesign in 2020 presented many new search options which can benefit online investigators. Once logged in, a simple search field will be present at the top of any Facebook page. This is a generic starting point. I encourage you to think of Facebook's current search landscape in two parts. The KEYWORD search is any generic term, name, location, or entity of interest. The FILTER search is the options which eliminate unwanted results. Let's start with a demonstration where we are looking for a profile of a person.

Typing in a target's real name should lead to results, many of which are unrelated to your investigation. Unlike other social networks, Facebook users typically use their real name when creating a profile. This profile is usually linked to an employer, graduating high school class, college alumni, or general interests. With billions of active users, it will be likely that you will locate several user profiles under the same name as your target. There are a few things that you can do to find the right person.

If your target's name is Tom Johnson, you have your work cut out for you. This does not mean that you will never find his Facebook page, but you will need to take additional steps to get to your target. When searching the name, several possibilities may appear in the results. This is obviously not the complete list of Tom Johnsons that are present on Facebook. At the bottom of this list is an option to "See All" the profiles with your target name. This is also not the complete list. Scrolling down should automatically populate more profiles. You could look through these and hope to identify your target based on the photo, location, or additional information displayed in this view. Instead, consider adding filters within the left menu.

Figure 10.01 displays my keyword search for "Tom Johnson" who lives in Chicago, Illinois, attended Vashon High School, and currently works at Foot Locker. This located only one result, as this was a very targeted query. The filters helped me get from thousands of targets to only one. However, this is not as easy as it sounds. There are multiple Vashon high schools and dozens of Foot Locker Facebook pages. In a moment, we will force Facebook to focus on specific entities.

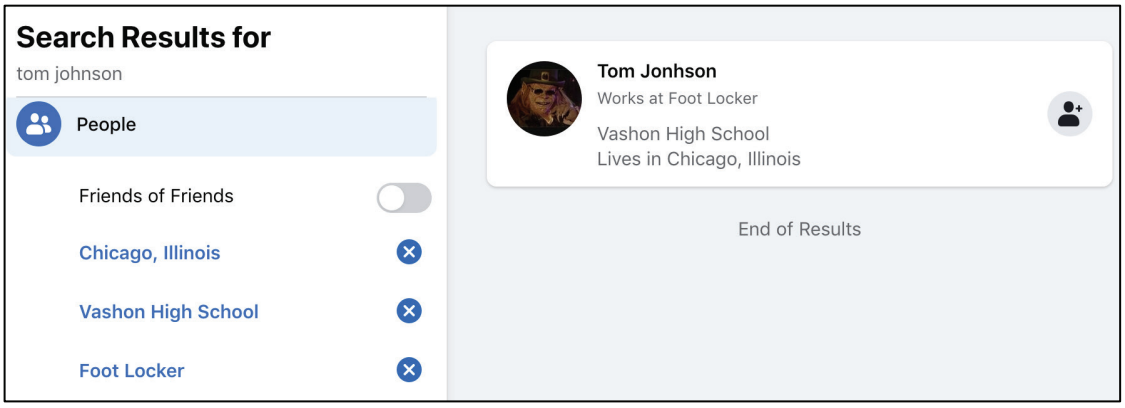


Figure 10.01: A Facebook keyword search with filters applied.

Once a user's profile is located, the default view is the "timeline" tab. This will include basic information such as gender, location, family members, friends, relationship status, interests, education, and work background. This page will also commonly have a photo of the user and any recent posts on their page. Clicking through this data may uncover valuable evidence, but you may be missing other data. I will explain numerous methods throughout this chapter which should help identify all available content relevant to your investigation.

First, consider using the traditional filter options available on most Facebook pages. Figure 10.02 displays the main filter bar on the top of every Facebook profile page. This will seek Photos, Videos, Places, Groups, and other options associated with the profile. You may be able to click through the various sections in order to reveal all publicly available content. My preference is to query via direct URL so that I know I did not miss anything. Assume that your target is Mark Zuckerberg. His profile is available at the following URL.

https://www.facebook.com/zuck

This indicates his Facebook username is "zuck". We can apply this to the following URLs, each which connect directly to the associated public information page. These direct URLs will be beneficial to our Facebook tool presented at the end of the chapter. Since "zuck" does not share much, I replaced him with a random username of "mike", which is not me.

Timeline: https://www.facebook.com/mike
About: https://www.facebook.com/mike/about
Employment: https://www.facebook.com/mike/about?section=work
Education: https://www.facebook.com/mike/about?section=education
Locations: https://www.facebook.com/mike/about?section=living
Contact Info: https://www.facebook.com/mike/about?section=contact-info
Basic Info: https://www.facebook.com/mike/about?section=basic-info
Relationships: https://www.facebook.com/mike/about?section=relationship
Family Members: https://www.facebook.com/mike/about?section=family
Bio: https://www.facebook.com/mike/about?section=bio
Life Events: https://www.facebook.com/mike/about?section=year-overviews
Friends: https://www.facebook.com/mike/friends
Following: https://www.facebook.com/mike/following
Profile Photos: https://www.facebook.com/mike/photos
Photo Albums: https://www.facebook.com/mike/photos_albums
Videos: https://www.facebook.com/mike/videos
Reels: https://www.facebook.com/mike/reels_tab
Check-Ins: https://www.facebook.com/mike/places_visited
Visits: https://www.facebook.com/mike/map
Recent Check-Ins: https://www.facebook.com/mike/places_recent
Sports: https://www.facebook.com/mike/sports
Music: https://www.facebook.com/mike/music
Movies: https://www.facebook.com/mike/movies
TV Shows: https://www.facebook.com/mike/tv
Books: https://www.facebook.com/mike/books
Apps & Games: https://www.facebook.com/mike/games
Likes: https://www.facebook.com/mike/likes
Events: https://www.facebook.com/mike/events
Facts: https://www.facebook.com/mike/did_you_know
Reviews: https://www.facebook.com/mike/reviews
Reviews Given: https://www.facebook.com/mike/reviews_given
Reviews Written: https://www.facebook.com/mike/reviews_written
Places Reviews: https://www.facebook.com/mike/place_reviews_written
Notes: https://www.facebook.com/mike/notes

The Facebook search tool presented in a moment will allow you to execute all of these URL queries easily in order to minimize the effort to explore a profile. Let's conduct another example of searching through Facebook's official channels. Assume you want to see photos posted by Mark Zuckerberg while he was in San Francisco in 2015. First, we need to obtain access to Facebook's filters, which are not visible when looking at a target's profile. My preferred way to do this is to conduct a search on Facebook for "photos" (without the quotes). This confuses Facebook a bit and tells it to present the "top" photos, but also presents the filter menu to the left. I can now click on "Photos" and enter the desired filters. Figure 10.03 displays my results.

This method was sloppy since I am forcing Facebook to include the keyword of "photos" within my search. It likely has no impact since we are seeking photos, but it could limit our search. If my target was posts, I would have begun my search with that term. At the time of this writing, there is no way to search Facebook's site by filters only. You must include a search term. In a moment, our tool will bypass this restriction with a new method. Photos can also be searched by generic location. Figure 10.04 demonstrates the "Photos" option after searching "Chicago". The filter on the left can filter results based on author, type, location, and date.

Let's conduct another keyword search within the official site. Assume you wanted to find any posts including the term "OSINT". After conducting the basic search, you should click "Posts". This expands the search and opens the Posts filter options. From there, you can filter by year or location. You could also click through the

other categories such as Videos or Groups. All basic Facebook filters can be applied using direct URLs. Considering an interest in the term "OSINT", the following addresses replicate each of the filters of a standard Facebook search.

All: https://www.facebook.com/search/top/?q=osint
Posts: https://www.facebook.com/search/posts/?q=osint
People: https://www.facebook.com/search/people/?q=osint
Photos: https://www.facebook.com/search/photos/?q=osint
Videos: https://www.facebook.com/search/videos/?q=osint
Marketplace: https://www.facebook.com/search/marketplace/?q=osint
Pages: https://www.facebook.com/search/pages/?q=osint
Places: https://www.facebook.com/search/places/?q=osint
Groups: https://www.facebook.com/search/groups/?q=osint
Events: https://www.facebook.com/search/events/?q=osint
Links: https://www.facebook.com/search/links/?q=osint
Watch: https://www.facebook.com/watch/search/?q=osint

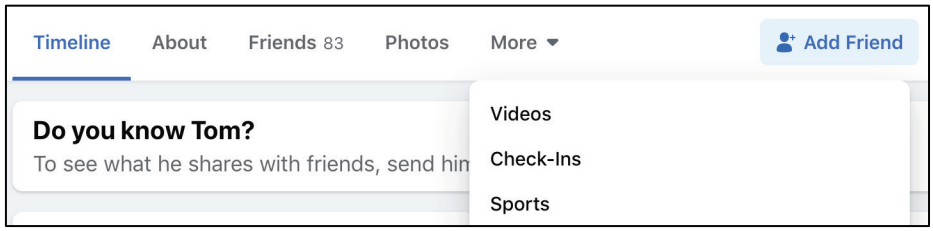


Figure 10.02: Facebook's main filter options.

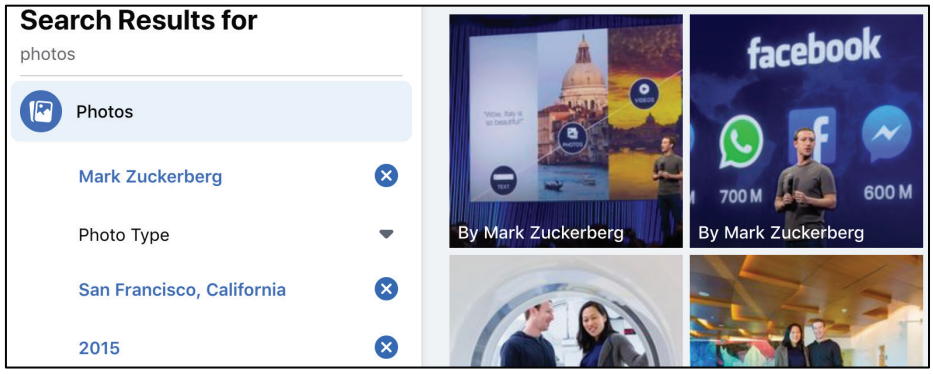


Figure 10.03: Facebook Photo filters in use.

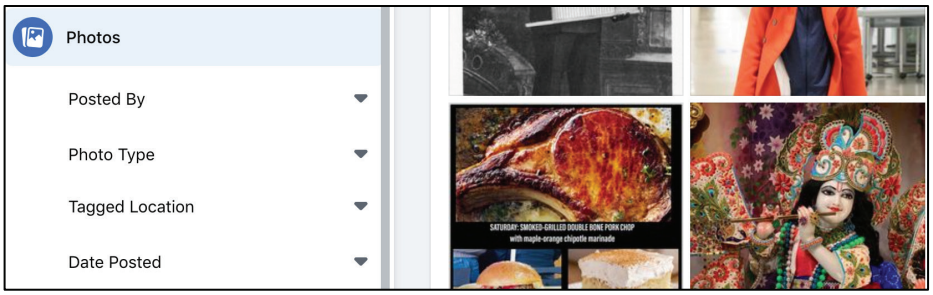


Figure 10.04: Photo results from Facebook filter options.

This presents the end of basic Facebook search techniques through the official site. Everything presented until now should apply for a while, and these URL structures should not change soon. Analyzing all of the evidence identified through these URLs should present substantial information. Many of these pages, such as a person's timeline, will load continuously. Pressing the space bar on your keyboard should load everything until the end. I have found taping down the space bar helpful for long pages. From here, we must dig deeper into profile data and apply some fairly technical methods in order to get to the next level.

Profile Details

At this point, you should be able to locate a target's profile by name with filters; analyze the publicly available content; and search by topic. That is just the tip of the iceberg. Facebook collects a lot of additional information from everyone's activity on the social network. Every time someone "Likes" something or is tagged in a photo, Facebook stores that information. Extracting these details can be difficult.

In order to conduct the following detailed searches, you must know the user number of your target. This number is a unique identifier that will allow us to search otherwise hidden information from Facebook. Prior to mid-2015, the easiest way to identify the user number of any Facebook user was through the Graph API. While you were on a user's main profile, you could replace "www" in the address with "graph" and receive the profile ID number of that user. This no longer works because Facebook removed the ability to search their graph API by username. However, we can still obtain this powerful number through a manual search option.

This technique involves viewing the source code of any user's Facebook profile. The process for this will vary by browser. In Firefox and Chrome, simply right-click on a Facebook profile page (not an image) and select "View Page Source". Be sure not to hover on any hyperlinks during the right-click. A new tab should open with the text-only view of the source code of that individual profile. Within the browser, conduct a search on this page for "userID". This will identify a portion of the code within this page that contains that specific term. As an example, the following is the source code visible in Zuck's profile.

"userID":"4"

Sometimes Firefox will search through the source code and highlight data, but the results don't match the terms searched. I do not know why this happens. When necessary, I copy all of the source code and paste it into a text editor, then search through it. In this example, the user ID of this profile is 4. We will use this number for numerous searches within the next instruction. Some users prefer to look at the URLs of a target's photos in order to identify the user ID, but I believe this is bad practice. If a user has no photos, this will not work. Also, Facebook's photo displays often hide this information from plain sight. I prefer to rely on the source code view or my Facebook tools for this identification. This number will allow us to obtain many more details about the account. Until July of 2019, there were dozens of online search tools which would identify the user ID number (4) when supplied a username (zuck). Almost all of these stopped functioning, including my own, when Facebook began aggressively blocking these search tools.

While you may find an online option which still functions, we should not rely on these. If you are exhausted from searching within each profile's source code in order to locate the user ID, I reluctantly offer the following site which currently attempts to automatically replicate this process.

https://findidfb.com/

Facebook Base64 Encoding

Prior to June of 2019, a simple URL would display content posted by an individual. As an example, the following URL would display all photos posted by a specific user (4).

https://facebook.com/search/4/photos-by

This technique no longer works, and the replacement method is much more difficult. Instead of the URL structure of "facebook.com/search", our base URL is as follows. Note a search term (q=photos) is required.

https://facebook.com/search/photos/?q=photos&epa=FILTERS&filters=

This is followed by the structure of the following.

{"rp_author":"{\"name\":\"author\",\"args\":\"[USERID]\"}"}

However, it must be presented in Base64 format, which would be the following.

eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJhdXRob3JcIixcImFyZ3NcIjpcIjRcIn0ifQ

Therefore, the entire URL would be the following.

https://www.facebook.com/search/photos/?q=photos&epa=FILTERS&filters=eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJhdXRob3JcIixcImFyZ3NcIjpcIjRcIn0ifQ

Confused? I sure was. It took me a while to process what was going on here. Let's start over, and approach the creation of a custom URL in three phases. First, let's tackle the Facebook search URL. In the previous example, I displayed the following.

https://facebook.com/search/photos/?q=photos&epa=FILTERS&filters=

Let's break this down. The following explains each section of this URL

https://facebook.com/	The Facebook domain
search/	Instructs Facebook to conduct a search
photos/	Specifies the type of information desired
?q=photos	Searches any photos (videos and posts works here too)
&epa=FILTERS&filters=	Finishes the URL with a filter demand

Next, we must formulate our target data and convert it to a specific type of encoding called Base64. This is likely used because it is extremely common and can be generated by any browser. Previously, I presented the following data.

{"rp_author":"{\"name\":\"author\",\"args\":\"[USERID]\"}"}

This tells Facebook that we are searching for information from a specific profile (author), and the [USERID] should contain your target's user ID number as determined earlier. If we were targeting "zuck", and knew his user number was "4", we would have the following data.

{"rp_author":"{\"name\":\"author\",\"args\":\"4\"}"}

Notice the position of "4" as the user number. Now, we must convert this data into Base64. I prefer the website https://codebeautify.org/base64-encode and we can check our work at the decoding version at https://codebeautify.org/base64-decode. When I copy and paste the previous data into this website, I receive the following.

eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJhdXRob3JcIixcImFyZ3NcIjpcIjRcIn0ifQ==

The final two "==" are optional, and not necessary. When I paste this value into the decoder on this website, I receive the exact data originally entered. Let's take a look at the screen captures. Figure 10.05 (top) displays the

desired text data, including my target's user ID number. Figure 10.05 (bottom) displays the result coded in Base64. Figure 10.06 (top) displays the opposite technique by converting the previous Base64 result back to the original desired data in Figure 10.06 (bottom). I realize this is confusing, but our search tools will simplify all of this in a moment.

Figure 10.05: Encoding text to Base64 on https://codebeautify.org/base64-encode.

Figure 10.06: Decoding text to Base64 on https://codebeautify.org/base64-decode.

Let's take a look at the entire URL as follows. Below it is a breakdown.

https://www.facebook.com/search/photos/?q=photos&epa=FILTERS&filters=eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJhdXRob3JcIixcImFyZ3NcIjpcIjRcIn0ifQ

https://facebook.com/	The Facebook domain
search/	Instructs Facebook to conduct a search
photos/	Specifies the type of information desired
?q=photos	Searches any photos (videos and posts works here too)
&epa=FILTERS&filters=	Finishes the URL with a filter demand
eyJycF9hdXRob3IiOiJ7X…	{"rp_author":"{\"name\":\"author\",\"args\":\"4\"}"}

Figure 10.07 displays the results of this URL. Facebook has provided images posted by our target, some of which do not appear on his profile. Clicking "See All" opens even more images.

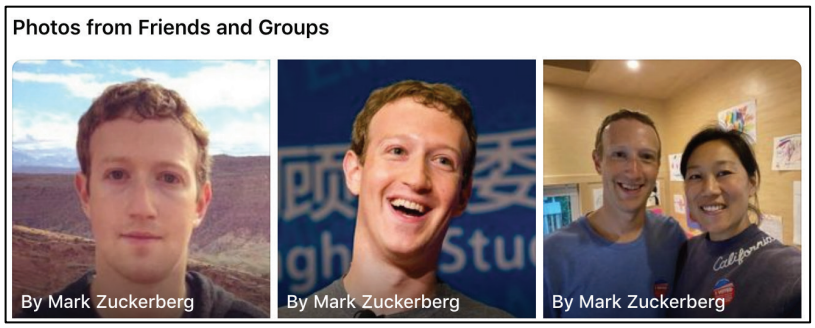


Figure 10.07: Image results from a Base64 search.

Let's start over with another example, this time focusing on public posts.

Assume we want to identify posts created by "zuck". We know his user number is "4", so we want to combine "https://facebook.com/search/posts/?q=posts&epa=FILTERS&filters=" with the Base64 encoding of "{"rp_author":"{\"name\":\"author\",\"args\":\"4\"}"}". This creates the following URL.

https://www.facebook.com/search/posts/?q=posts&epa=FILTERS&filters=eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJhdXRob3JcIixcImFyZ3NcIjpcIjRcIn0ifQ

A partial result can be seen in Figure 10.08. The results are public posts, but some may not be from his profile. In one investigation, I had a suspect with absolutely no posts within his timeline. However, this method identified hundreds of posts he made on other people's profiles. These included comments within posts and public messages to others.

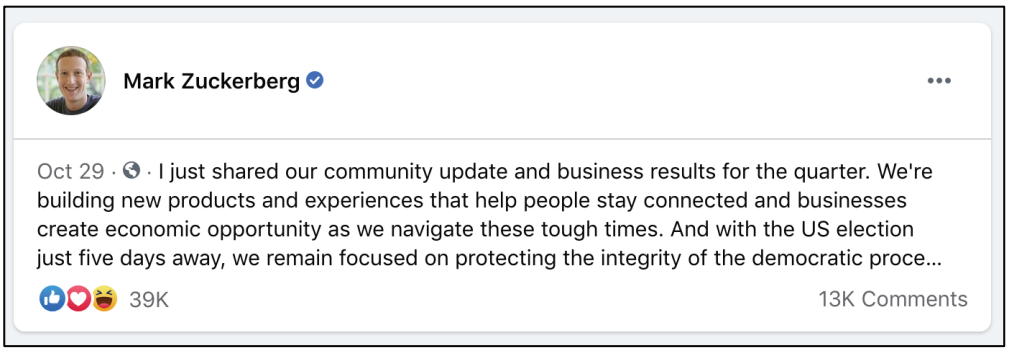


Figure 10.08: Post results from a Base64 search.

Let's make this simpler. In a moment, I will present the most common URL queries which I have found valuable to my investigations. Then, I will present the portion of my custom Facebook tools which automates this entire process. First, below is a summary of each search by section.

Posts by User/Year:	Posts by a specific user on any page (filtered by year)
Photos by User/Year:	Photos by a specific user on any page (filtered by year)
Videos by User/Year:	Videos by a specific user on any page (filtered by year)
Posts by Location/Year:	Posts submitted from a specific location (filtered by year)
Photos by Location/Year:	Photos submitted from a specific location (filtered by year)
Videos by Location/Year:	Videos submitted from a specific location (filtered by year)
Posts by Keyword/Year:	Posts matching a search term (filtered by year)
Photos by Keyword/Year:	Photos matching a search term (filtered by year)
Videos by Keyword/Year:	Videos matching a search term (filtered by year)
Events by Location:	Current and upcoming events by location (city or entity)
Profiles by Employer:	Profiles associated with employees of a specific business
Profiles by Location:	Profiles associated with residents of a specific city
Profiles by School:	Profiles associated with students of a specific school
Posts by Date:	Posts by keyword (filtered by date range)
Photos by Date:	Photos by keyword (filtered by date range)
Videos by Date:	Videos by keyword (filtered by date range)

If your target was the Facebook username "zuck" with user number "4", the first three URLs would be as follows. Notice that the Base64 encoding is identical on each and that I bolded the unique portions.

Posts by User:
https://www.facebook.com/search/**posts**/?q=**posts**&epa=FILTERS&filters=eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJhdXRob3JcIixcImFyZ3NcIjpcIjRcIn0ifQ

Photos by User:
https://www.facebook.com/search/**photos**/?q=**photos**&epa=FILTERS&filters=eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJhdXRob3JcIixcImFyZ3NcIjpcIjRcIn0ifQ

Videos by User:
https://www.facebook.com/search/**videos**/?q=**videos**&epa=FILTERS&filters=eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJhdXRob3JcIixcImFyZ3NcIjpcIjRcIn0ifQ

If your target was the city of Chicago, user number "108659242498155", the desired URLs would be as follows if no year was chosen. Selecting a specific year only changes the Base64 encoding.

Posts by Location:
https://www.facebook.com/search/**posts**/?q=**posts**&epa=FILTERS&filters=eyJycF9sb2NhdGlvbiI6IntcIm5hbWVcIjpcImxvY2F0aW9uXCIsXCJhcmdzXCI6XCIxMDg2NTkyNDI0OTgxNTVcIn0ifQ

Photos by Location:
https://www.facebook.com/search/**photos**/?q=**photos**&epa=FILTERS&filters=eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJsb2NhdGlvblwiLFwiYXJnc1wiOlwiMTA4NjU5MjQyNDk4MTU1XCJ9In0

Videos by Location:
https://www.facebook.com/search/**videos**/?q=**videos**&epa=FILTERS&filters=eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJsb2NhdGlvblwiLFwiYXJnc1wiOlwiMTA4NjU5MjQyNDk4MTU1XCJ9In0

We can now drill down within our queries. The following URLs would identify any time our target (zuck - user number 4) mentions "OSINT" within a post, photo, and video. Notice how the bolded areas have changed from the previous queries.

Posts by Keyword:
https://www.facebook.com/search/**posts**/?q=**OSINT**&epa=FILTERS&filters=eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJhdXRob3JcIixcImFyZ3NcIjpcIjRcIn0ifQ%3D%3D

Photos by Keyword:
https://www.facebook.com/search/**photos**/?q=**OSINT**&epa=FILTERS&filters=eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJhdXRob3JcIixcImFyZ3NcIjpcIjRcIn0ifQ%3D%3D

Videos by Keyword:
https://www.facebook.com/search/**videos**/?q=**OSINT**&epa=FILTERS&filters=eyJycF9hdXRob3IiOiJ7XCJuYW1lXCI6XCJhdXRob3JcIixcImFyZ3NcIjpcIjRcIn0ifQ%3D%3D

We can now use additional filters to find events and profiles, as follows.

Events (Protest) by Location (City of Chicago-108659242498155):
https://www.facebook.com/search/**events**/?q=**protest**&epa=FILTERS&filters=eyJycF9ldmVudHNfbG9jYXRpb24iOiJ7XCJuYW1lXCI6XCJmaWx0ZXJfZXZlbnRzX2xvY2F0aW9uXCIsXCJhcmdzXCI6XCIxMDg2NTkyNDI0OTgxNTVcIn0ifQ%3D%3D

Profiles (Tom) by Employer (Harvard-105930651606):
https://www.facebook.com/search/**people**/?q=**tom**&epa=FILTERS&filters=eyJlbXBsb3llciI6IntcIm5hbWVcIjpcInVzZXJzX2VtcGxveWVyXCIsXCJhcmdzXCI6XCIxMDU5MzA2NTE2MDZcIn0ifQ

Profiles (Tom) by Location (City of Chicago-108659242498155):
https://www.facebook.com/search/**people**/?q=**tom**&epa=FILTERS&filters=eyJjaXR5Ijoie1wibmFtZVwiOlwidXNlcnNfbG9jYXRpb25cIixcImFyZ3NcIjpcIjEwODY1OTI0MjQ5ODE1NVwifSJ9

Profiles (Tom) by School (Harvard-105930651606):
https://www.facebook.com/search/**people**/?q=**tom**&epa=FILTERS&filters=eyJzY2hvb2wiOiJ7XCJuYW1lXCI6XCJ1c2Vyc19zY2hvb2xcIixcImFyZ3NcIjpcIjEwNTkzMDY1MTYwNlwifSJ9

In 2022, we saw a change to the way Facebook handles searching by date. I have found searching by a single date to be unreliable, but searching within a range of dates, even if the beginning and end are the same date, seem to function better. Because of the two dates, the URL created by the custom tools will be longer (twice the data). At the time of this writing, video queries were blocked, but searching for "top" entries was functioning and appeared to be displaying mostly videos. There is no limit to the date range, but these focus on one day.

Posts (OSINT) by Date (September 5, 2022-September 5, 2022):
https://www.facebook.com/search/**posts**/?q=**osint**&filters=eyJycF9jcmVhdGlvbl90aW1lIjoie1wibmFtZVwiOlwiY3JlYXRpb25fdGltZVwiLFwiYXJnc1wiOlwie1xcXCJzdGFydF95ZWFyXFxcIjpcXFwiMjAyMlxcXCIsXFxcInN0YXJ0X21vbnRoXFxcIjpcXFwiMjAyMi05XFxcIixcXFwiZW5kX3llYXJcXFwiOlxcXCIyMDIyXFxcIixcXFwiZW5kX21vbnRoXFxcIjpcXFwiMjAyMi05XFxcIixcXFwic3RhcnRfZGF5XFxcIjpcXFwiMjAyMi05LTVcXFwiLFxcXCJlbmRfZGF5XFxcIjpcXFwiMjAyMi05LTVcXFwifVwifSIsInJwX2F1dGhvciI6IntcIm5hbWVcIjpcIm1lcmdlZF9wdWJsaWNfcG9zdHNcIixcImFyZ3NcIjpcIlwifSJ9

Photos (OSINT) by Date (September 5, 2022-September 5, 2022):
https://www.facebook.com/search/**photos**/?q=**osint**&filters=eyJycF9jcmVhdGlvbl90aW1lIjoie1wibmFtZVwiOlwiY3JlYXRpb25fdGltZVwiLFwiYXJnc1wiOlwie1xcXCJzdGFydF95ZWFyXFxcIjpcXFwiMjAyMlxcXCIsXFxcInN0YXJ0X21vbnRoXFxcIjpcXFwiMjAyMi05XFxcIixcXFwiZW5kX3llYXJcXFwiOlxcXCIyMDIyXFxcIixcXFwiZW5kX21vbnRoXFxcIjpcXFwiMjAyMi05XFxcIixcXFwic3RhcnRfZGF5XFxcIjpcXFwiMjAyMi05LTVcXFwiLFxcXCJlbmRfZGF5XFxcIjpcXFwiMjAyMi05LTVcXFwifVwifSIsInJwX2F1dGhvciI6IntcIm5hbWVcIjpcIm1lcmdlZF9wdWJsaWNfcG9zdHNcIixcImFyZ3NcIjpcIlwifSJ9

Videos (OSINT) by Date (September 5, 2022-September 5, 2022):
https://www.facebook.com/search/**top**/?q=**osint**&filters=eyJycF9jcmVhdGlvbl90aW1lIjoie1wibmFtZVwiOlwiY3JlYXRpb25fdGltZVwiLFwiYXJnc1wiOlwie1xcXCJzdGFydF95ZWFyXFxcIjpcXFwiMjAyMlxcXCIsXFxcInN0YXJ0X21vbnRoXFxcIjpcXFwiMjAyMi05XFxcIixcXFwiZW5kX3llYXJcXFwiOlxcXCIyMDIyXFxcIixcXFwiZW5kX21vbnRoXFxcIjpcXFwiMjAyMi05XFxcIixcXFwic3RhcnRfZGF5XFxcIjpcXFwiMjAyMi05LTVcXFwiLFxcXCJlbmRfZGF5XFxcIjpcXFwiMjAyMi05LTVcXFwifVwifSIsInJwX2F1dGhvciI6IntcIm5hbWVcIjpcIm1lcmdlZF9wdWJsaWNfcG9zdHNcIixcImFyZ3NcIjpcIlwifSJ9

All of these queries will be easily created using our custom Facebook search tool in a moment, including filtration by dates for full profile analysis. As a reminder, I obtained the user ID number for the profile of "zuck" by searching the source code of the profile for "userID". However, location profiles do not possess this data. Fortunately, the user ID for cities is within the URL. When I searched for "City of Chicago" on Facebook, I clicked "Places" category and selected the official City of Chicago profile which possessed the following URL.

https://www.facebook.com/places/Things-to-do-in-Chicago-Illinois/108659242498155/

The last group of numbers (108659242498155) is the user ID number.

Business pages, such as those associated with Harvard, Pepsi, and others, do not present this data within the URL. Instead, we must search within the source code for "pageID" (including the quotes). When searching this within the source code for the Harvard page, I observed the following.

"pageID":"105930651606"

Pepsi's page (pepsiUS) included the following.

"pageID":"56381779049"

Event profile numbers can be found by searching "eventID".

Group profile numbers can be found by searching "groupID".

While I always prefer manual extraction from the source code of my target Facebook profile, the Facebook ID tools previously presented should also identify profile numbers of business, event, and group pages.

It is extremely important to note that I did not discover this Base64 conversion technique. The online researcher NEMEC (https://twitter.com/djnemec) was the first to post about this method. Practically every online search tool generating these types of URLs, including mine, is due to the work of this person. I send many "thanks" on behalf of the entire OSINT community for this work.

IntelTechniques Facebook Tool

If you have not already skipped to the next chapter after all of this confusion, it is now time to simplify the process. Finally, let's use the custom Facebook search tool to replicate all of this work. This interactive document is included within the "Tools" you previously downloaded or at **https://inteltechniques.com/tools**. Figures 10.09 and 10.10 display the current version of the tool. The following pages explain each section with examples.

The first input box in the "Facebook Profile Data" section allows you to enter a Facebook username, such as zuck, and the tool will populate that data within all of the search options in that section. This makes it easy to begin your queries. You can then use the submit buttons to search each entry, such as Timeline or Photos. Note there is no "Submit All" option here. This is due to Facebook's scrutiny into accounts which automate any queries. Submitting too many requests simultaneously will result in a suspended account.

The next section titled "Facebook Search Data" allows you to replicate the instruction on keyword searches throughout all twelve sections of Facebook, such as Posts, Photos, and Pages. This presents immediate results without relying on Facebook to present your desired data from a generic keyword search.

The "Base64 Conversion Queries" section presents the true power behind this tool. All of the technical search techniques presented within the previous pages are replicated here without the need to do any of the work. Instead of summarizing this section, let's run through a demonstration of each option.

Posts/Photos/Videos by User

I entered the user ID number of "4" (for the user zuck) in the first portion, as seen in Figure 10.11 (top). This allows me to query posts, photos, and videos posted by that user to any public pages. The drop-down menu allows me to filter by year, if desired, which is helpful if the target posted a lot of content. This option is available for all three queries. This often identifies content unavailable within the target's own profile.

Posts/Photos/Videos by Location

I entered the user number of "108659242498155" (Chicago) in the first portion, as seen in Figure 10.11 (bottom). This allows me to query posts, photos, and videos posted from that location to any public pages. The drop-down menu allows me to specify a year, if desired. This often identifies content unavailable through traditional search techniques.

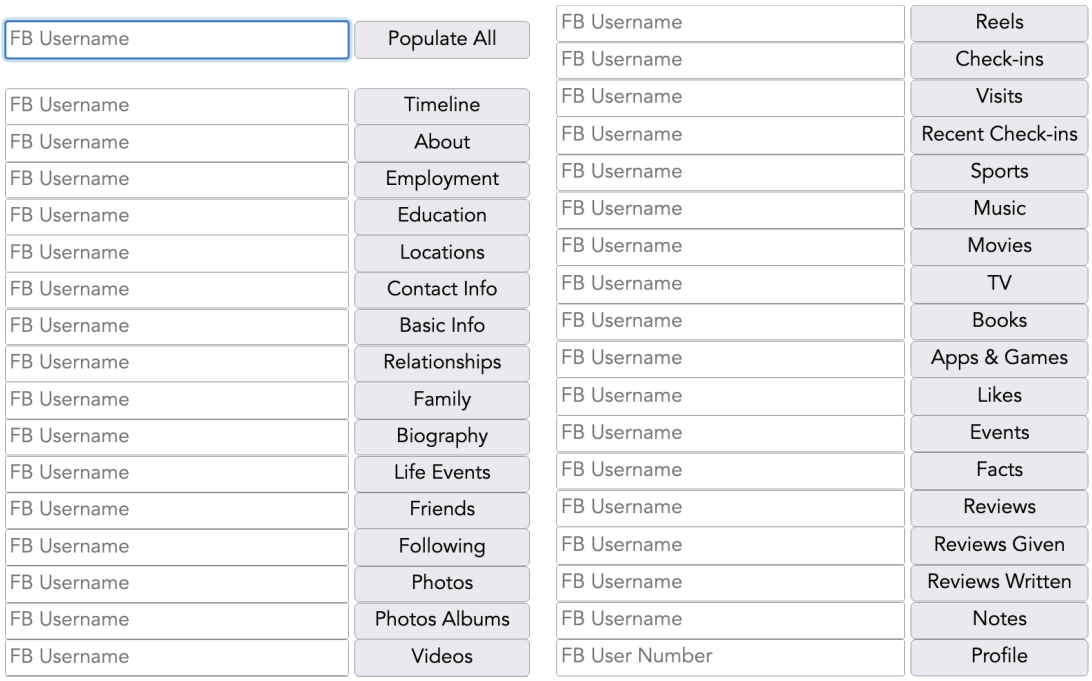


Figure 10.09: The IntelTechniques Facebook Tool.

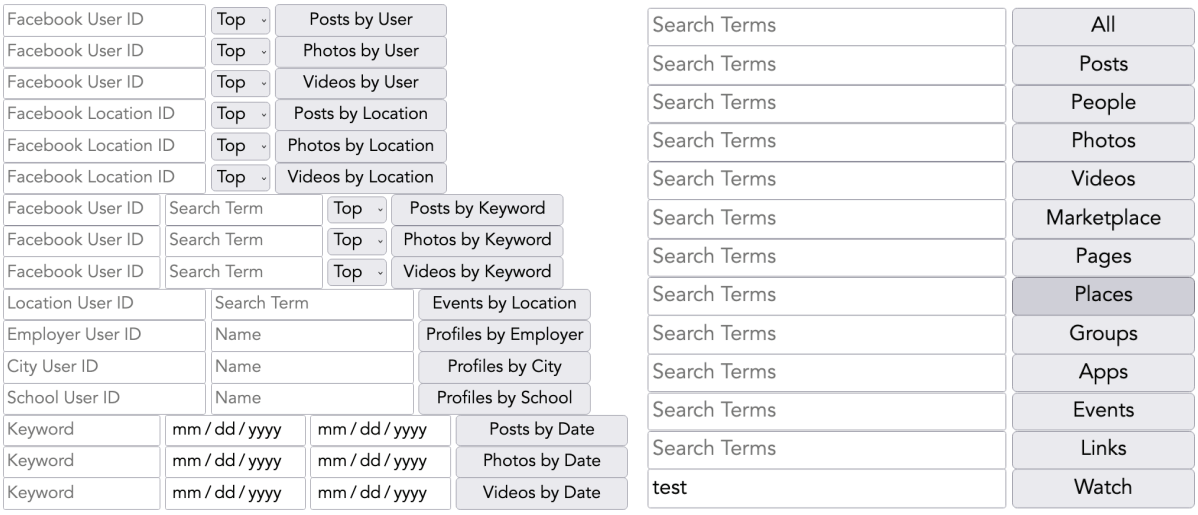


Figure 10.10: The IntelTechniques Facebook Tool.

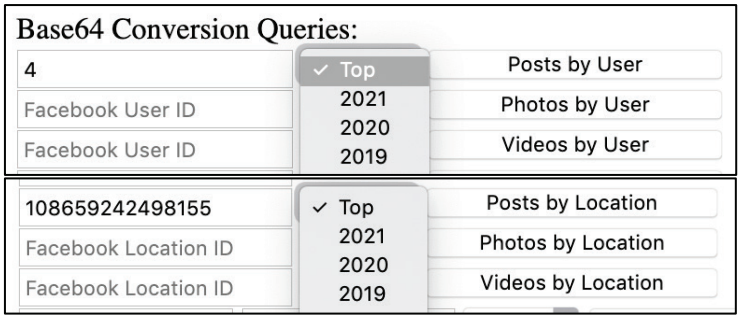


Figure 10.11: The IntelTechniques Facebook Tool options with date filtering.

Posts/Photos/Videos by Keyword

I entered the user number 4 (zuck) and a search term of "OSINT" in Figure 10.12. I can now select a year and filter only those posts, photos, or videos.

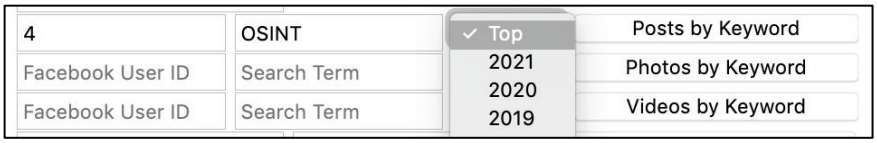


Figure 10.12: The IntelTechniques Facebook Tool year filters.

Events by Location

I entered the user number of "108659242498155" (Chicago) and added the keyword "protest". I was immediately presented dozens of active and future events matching my criteria.

People by Employer, City, and School

I entered the user number of "105930651606" (Harvard) in the first and third boxes, as seen in Figure 10.13. This allows me to query people's profiles by their employer or school affiliation. These search options require at least one additional piece of information. In this example, I searched the name "Tom". Notice the difference in the results within Figure 10.14. The left result identifies people named Tom who WORK at Harvard, while the right displays people named Tom who ATTENDED Harvard. This search option can quickly identify your target when you only know a small amount of information, or if they are using a false last name.

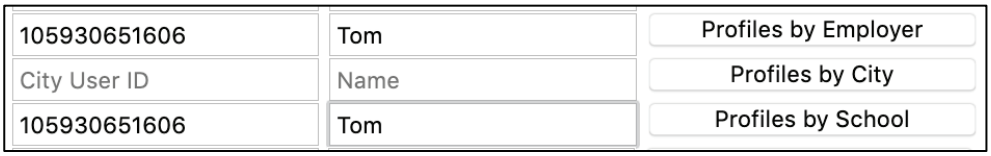


Figure 10.13: The IntelTechniques Facebook Tool options for profile search.

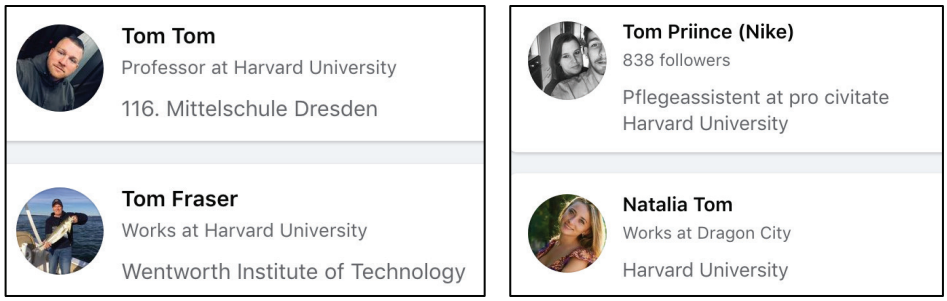


Figure 10.14: Facebook results from an IntelTechniques Tool query.

I suspect that these queries will change over time. I do not present the source code of these specific queries within the book because they are quite complicated and lengthy. If this tool should be updated, I will make an announcement at **https://inteltechniques.com/osintbook10**.

Manual vs. Tool Queries

If you are paying close attention, you can see that there are many similarities between conducting a search directly within the official Facebook website and the options within my custom search tools. On the surface, these tools

may seem redundant and unnecessary. However, I assure you that there are some key advantages to the Facebook tool. I present three quick examples where you can find content through the tool which is not available on the site.

Videos by Date: The official Facebook site allows filtration of videos by date, but you can only choose from "Today", "This Week", or "This Month". The tool allows detailed filters by year or exact date. The example below is from 2015.

Check out this 360 video timelapse of Facebook's campus. I'm really enjoying these 360 videos. They...
Check out this 360 video timelapse of Facebook's campus. I'm really enjoying these 360 videos. They feel like you're really...
Mark Zuckerberg
Nov 17, 2015 · 2.5M Views

Posts by Location: I can query the official LAX page at facebook.com/LAInternationalAirport, but I cannot query for posts from other people at that location. However, the search tool allows me to focus on posts from any location which can be filtered by year. The following example was presented after entering location ID "74286767824" and choosing "2018".

Dec 26, 2018 · · So we are waiting to board and start our adventure to Rome! Sorry in advance for all the posts!

Posts by Date: I can search for any term on Facebook, but I can only filter by year. With the tools, I can focus on any individual date. In the following example, I searched for anyone mentioning my site on the day that it was briefly shut down due to legal demands, I quickly found evidence.

Jun 6, 2019 · · @IntelTechniques What a shame man, hate to hear that. Hope to have someway to access the same

I encourage you to experiment with both manual and automated search options. There is a small learning curve when querying Facebook, but the results are often fascinating.

Facebook Phone Number Search

In previous editions, I have included various techniques to identify a Facebook profile associated with a target telephone number. At one time, entering a cell number into a Facebook search showed you the name, email address, and photo of the user. That luxury is gone, but we do currently have one method which may be helpful. The following URL requests Facebook to locate an account for a password reset. Entering a telephone number will confirm if an account is associated to it, and will sometimes display the profile image for the user.

https://mbasic.facebook.com/login/identify/?ctx=recover

However, clicking "Try Another Way" within this technique should present even more details. While testing this method on an unknown target's number, I received the following.

Send code via email
n***@g******
n***@y******

This confirms that a Facebook account is associated with the number; there are two email addresses associated with the account; both email addresses begin with "n"; and the email service providers begin with "g" and "y". My guess is that this user has a similar email address with both Yahoo and Gmail.

Facebook ID Creation Date

Digital forensics enthusiast and private investigator Josh Huff at LearnAllTheThings.net has conducted a lot of research into the assignment of user ID numbers to Facebook profiles. We know that these numbers are assigned in chronological order, but the intelligence goes much further beyond that. His research could fill several pages and, in some situations, he can identify the month when a specific user account was created. For the sake of space and most useful details, he has provided the following to me for publication.

Facebook transitioned from 32-bit numbers, such as 554432, to 64-bit numbers that begin with 100000 between April and December of 2009. Therefore, if your target's number is less than 100000000000000, the account was likely created prior to April 2009. An account with a 15-digit ID number would have been created after December 2009. We can break down the numbers by year. The following are rough estimates that should only be used as a general guideline. Facebook appears to have begun issuing random numbers outside of these ranges in 2018.

2006: Numbers less than 600400000
2007: 600400000 - 1000000000
2008: 1000000000 - 1140000000
2009: 1140000000- 100000628000000
2010: 100000629000000 - 100001610000000
2011: 100001611000000 - 100003302000000
2012: 100003303000000 - 100004977000000
2013: 100004978000000 - 100007376000000
2014: 100007377000000 - 100008760000000
2015: 100008761000000 - 100010925000000
2016: 100010926000000 - 100014946000000
2017: 100014947000000 - 100023810000000
2018: 100023811000000 -

Facebook Friends Extraction

I was recently presented a Facebook scenario without any obvious solution. Assume that you find the Facebook profile of your target, and there is a large list of "Friends". You want to document this list, and a screenshot is not good enough. You want to capture the hyperlinks to each account and have a data set that can be manipulated or imported elsewhere. There are several outdated online tools that claim to "scrape" this data, but none of them work. In 2016, Facebook changed their Terms of Service (TOS) which now blocks any type of scraping of Friend data. Furthermore, the Facebook API no longer allows the display of this data either. There is a solution, and it will involve minimal geek work. First, identify the page of your target. For this example, I will use public profile at https://www.facebook.com/darya.pino/friends.

She has several friends, so I will hold down the space bar on my keyboard to load the entire page. I will now highlight her entire friends list and use "Ctrl" + "C" (or right-click > copy) to copy the data. I find it best to click directly above the left side of the first friend and hold until the lower right area of the last friend. The friends list should highlight. Now, open Microsoft Excel. Click on the "B" in column B to highlight the entire column. Paste the content with either "Ctrl" + "V" (or right-click > paste). This will appear disorganized, but the data is there. The images will be on top of the user data, which will not work for a final report. Use F5 to launch the "Go To" menu and select "Special". Select "Objects" and click OK. This will select all of those images. Hit the delete key to remove them. You will now see only the text data (with hyperlinks). Now click on

the "A" in column A and paste the friend content again with either "Ctrl" + "V" (or right-click > paste). Right-click any cell in this column and choose "Clear Contents". This will remove any text, but keep the images.

Place your mouse in between columns A and B and resize column A to be a bit larger than one of the images. Do the same with Column B to fit in all of the text. Use the "Find and Replace" feature to find every instance of "Add Friend" and replace it with nothing. This will remove those unnecessary entries. In the "Home" menu, choose "Format" and then "Auto Fit Row Height". This will eliminate unnecessary spacing. Select Column B and Left Justify the text. Your final result will be a clean spreadsheet with all of the images, names, and active links from your target's Facebook "friends" page. This is not the cleanest way of doing things, but it will work.

Email Search

As stated previously, we lost all standard email address search options within Facebook in 2019. However, there is one remaining technique which allows submission of an email address and provides identification of an associated profile. However, it is not simple or straightforward. It will take some work to set up, but will then be available as needed. This is often referred to as the "Page Role" trick. The premise is that you create a Facebook business page and then assign another profile to possess management rights. When you enter the email address of the target, Facebook confirms the name and profile to make sure you truly want to give away authorization to the person. We can then cancel the request without any notification to the target. The following steps replicate this technique.

- Open "Menu" and click "Page" in the upper-right corner, then create a new page.
- Assign a random name to your profile, select any category, and click "Continue".
- Skip all optional steps.
- Once you see the new profile, click the "Manage" button in the upper-right.
- On the new menu, click "Page Access" in the left column.
- In the "People with Facebook access" section, click "Add new".
- Enter the target email address.

This should present any Facebook profiles associated with the address entered. Figure 10.15 displays a result. I entered an email address in this field and was presented the only Facebook profile which was associated with the account. I can now search this full name within Facebook, look for the image previously displayed, and scour the target profile for valuable information. Facebook continuously makes both minor and major changes to their search functions. Some of these instructions may not work one day and work fine the next. Your mileage will vary as Facebook scrutinizes your covert profiles, VPN protected networks, and overall "vibe" as a fake user. Hopefully, this chapter has given you new ideas on ways to completely analyze your next target on Facebook and ideas to circumvent the next roadblocks.

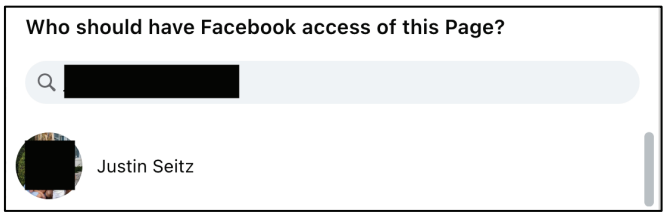


Figure 10.15: An email search through Facebook under the Page Access menu.

Facebook investigations are always a moving target. Throughout all of the services explained in this book, Facebook is the most likely to change often. Fortunately, any changes always bring new investigation opportunities. When we lost Graph search in 2019, we gained Base64 methods. When we lose the current strategies, something else will become available. As with all OSINT techniques, daily practice and understanding of the resources is more vital than the occasional nugget of data which is displayed on our screens.

CHAPTER ELEVEN
SOCIAL NETWORKS: TWITTER

Twitter is a social network and microblogging service that limits most posts to 280 characters. In 2022, Twitter reported that there were over 500 million Twitter posts, or "Tweets", posted every day. Basically, users create a profile and post Tweets announcing their thoughts on a topic, current location, plans for the evening, or maybe a link to something that they feel is important. A user can "follow" other users and constantly see what others are posting. Likewise, a user's "followers" can see what that user is up to on a constant basis. The premise is simply sharing small details of your life for all of your friends to see, as well as the rest of the world. Most people utilize the service through a mobile app within a cellular phone. Obtaining information from Twitter can be conducted through numerous procedures, all of which are explained here. Similar to Facebook, we will start with official search options through Twitter's website. Unlike Facebook, you do not need to be logged in to their service in order to conduct queries on their website.

Twitter Search (twitter.com/explore)

This is the official site's search interface, but it is nothing different than the search field at the top of any Twitter profile or search result. I only present this because navigating to twitter.com often offers a signup page, but no option to search. We will use this standard search bar to conduct specific queries in a few moments.

Twitter Advanced Search (twitter.com/search-advanced)

This page will allow for the search of specific people, keywords, and locations. The problem here is that the search of a topic is often limited to the previous seven to ten days. Individual profiles should display Tweets as far back as you are willing to scroll. This can be a good place to search for recent data, but complete archives of a topic will not be displayed. The following explains each section.

All of these words: The order of wording is ignored here, and only the inclusion of each of the words entered is enforced.

This exact phrase: Every Twitter search takes advantage of quotes to identify exact word placement. Optionally, you can conduct the search here to get precise results without quotes.

Any of these words: You can provide multiple unique terms here, and Twitter will supply results that include any of them. This search alone is usually too generic.

None of these words: This box will filter out any posts that include the chosen word or words.

These Hashtags: This option will locate specific posts that mention a topic as defined by a Twitter hashtag. This is a single word preceded by a pound sign (#) that identifies a topic of interest. This allows users to follow certain topics without knowing usernames of the user submitting the messages.

From these accounts: This section allows you to search for Tweets from a specific user. This can also be accomplished by typing the username into the address bar after the Twitter domain, such as twitter.com/JohnDoe92. This will display the user's profile including recent Tweets.

To these accounts: This field allows you to enter a specific Twitter username. The results will only include Tweets that were sent to the attention of the user. This can help identify associates of the target and information intended for the target to read.

Mentioning these accounts: While these messages might not be in response to a specific user, the target was mentioned. This is usually in the form of using "@". Anyone mentioning me within a Tweet may start it with @inteltechniques.

Dates: The final option allows you to limit a search to a specific date range. We will do this manually in just a moment.

Overall, I do not ever use the Twitter Advanced Search page. You can replicate all of these queries with manual search operators within the search field available on every Twitter page. I will explain each as we go through the chapter. Knowing the Twitter search operators instead of relying on the advanced search page has many benefits. First, we can use these techniques to monitor live Twitter data, as explained later in this chapter. Next, manual searches allow us to better document our findings. This is vital if your discoveries will be used in court.

Later in the chapter, we will combine all of our new techniques within our own custom tool options, which will be much more powerful than these standard solutions. The results of any of these searches can provide surprisingly personal information about a target, or generic content that includes too much data to be useful. We must learn each technique within a specific order before we can combine them into powerful queries.

Twitter data can be used for many types of investigations. Law enforcement may use this data to verify or disprove an alibi of a suspect. When a suspect states in an interview that he was in Chicago the entire weekend, but his Twitter feed displays his Tweet about a restaurant in St. Louis, he has some explaining to do. Private investigators may use this content as documentation of an affair or negative character. Occasionally, a citizen will contact the authorities when evidence of illegal activity is found within a person's Tweets. The possibilities are endless. First, let's find your target's Twitter profile.

Twitter Person Search

Locating your target's Twitter profile may not be easy. Unlike Facebook, many Twitter users do not use their real name in their profiles. You need a place to search by real name. I recommend Twitter's official search page before relying on third parties. After searching any real name or username within the standard Twitter search field, click through the five menu options, which are each explained below.

Top: This displays popular tweets matching your query. If your target is popular, you may see something here. I typically avoid this tab as it allows Twitter to determine what evidence I should see instead of selecting content based on specific queries.

Latest: This presents a reverse-chronological list of data which matches your search. It always begins with the most recent post and goes backward. This works well when searching a topic, but not when locating a target's profile.

People: Scrolling through this list allows you to look through the photo icons and brief descriptions to identify your target. Clicking on the username will open the user's Twitter profile with more information. This is the best option for searching by real name or username.

Photos: This searches for any photos matching your query. This has been helpful when attempting to identify images posted by unknown people who mentioned your target by name.

Videos: This searches for any videos matching your query. Similar to the previous example, this often locates videos associated with your target.

If searching by real name through the previous methods does not produce your target, your best option is to focus on the potential username. I will discuss username searches at length in a later chapter (Usernames). Overall, any associated usernames from other networks such as Instagram, Snapchat, and YouTube, should be tried on Twitter in the format of twitter.com/username.

Search by Email Address

Technically, Twitter does not allow the search of a username by providing an email address. If you attempt this type of search on Twitter, you will receive no results, even if a profile exists for a user with that email address. To bypass this limitation, you can use a feature offered within the mobile app version of Twitter. This technique will require a new Twitter account and the Twitter app present within a virtual Android device, as previously explained. This could also be replicated on a physical mobile device, but that is not recommended.

Within Android, open the Contacts application and add the email address of your target within a new contact and save the entry. Open the Twitter app, navigate to the left menu and select "Settings and privacy", click on "Privacy and safety", then "Discoverability and contacts". You can then manage your data and allow Twitter to sync your Android contacts to their service.

Enabling this feature will not display your contacts within the application. Instead, look at your notification bar within Android. You should see a pending notification which states "Find your friends on Twitter". Clicking this should present the "Suggested Followers". Twitter will encourage you to add these profiles, but do not choose that option. If you do, your target will be sent a notification from your account. In 2022, I witnessed this strategy stop working within newer accounts, but appeared to still function within aged accounts. I suspect we will see this technique continue to fade away.

Search Operators

Similar to the way that search engines use operators as mentioned in previous chapters, Twitter has its own set of search operators that will greatly improve your ability to effectively search Twitter. Two of the most powerful options are the "to" and "from" operators. I use these daily in order to filter results. Consider the following examples using our target Twitter username of twitter.com/IntelTechniques. We can obviously navigate directly to her page, but it is full of promoted Tweets, ReTweets, and whatever content she wants us to see. Instead, the following search within any Twitter window will limit our results only to her outgoing Tweets from her account. Clicking the "Latest" option in the Twitter menu will place these in reverse-chronological order.

from:IntelTechniques

This provides a better view into her thoughts. Both her Twitter profile and this results page give us insight to the messages she is sending out, but what about the incoming content? With most traditional Twitter investigations, we tend to focus on one side of the conversation. There is often a plethora of associated messages being sent to the attention of the target that go unchecked. In order to see all of the posts being publicly sent to her, we would search the following.

to:IntelTechniques

We now see all of those incoming messages that she cannot control. While she can prohibit them from being seen on her profile, she cannot block us from this search. When I have a missing person or homicide victim, I would much rather see the incoming messages versus the outgoing. We can also combine these options to create an extremely targeted query. At first glance, I did not see many incoming messages to IntelTechniques from protonprivacy. However, the following Twitter search tells a different story. It isolates only these posts.

to:IntelTechniques from:protonprivacy

We can also filter by replies with the following search. This would only display the tweets from our target which were replies to someone else.

from:IntelTechniques filter:replies

I do not find this very helpful. I prefer the opposite of this approach. The following search within Twitter would display only the tweets by my target which do NOT include replies.

from:IntelTechniques -filter:replies

Media, Likes, Lists, Moments, Topics, Links, and Followers

You may want to filter all results from a target Twitter profile and only see those which have some type of media embedded. There is not a search operator to force this, but the following direct URL will display only these posts.

https://twitter.com/IntelTechniques/media/

A trend that has seen rapid adoption over the past few years is the "Liking" of posts. When a user wants to acknowledge something said by another user, but does not necessarily want to respond or ReTweet, clicking the small heart icon indicates that the post was "liked". The following direct URL displays all of the posts that our target liked. For unknown reasons, this query requires you to be logged in to an account.

https://twitter.com/IntelTechniques/likes/

Twitter Lists appeared recently, and there are two types which we might have interest. A traditional List is created by the target and may include people they know or like. The structure is as follows.

https://twitter.com/IntelTechniques/lists/

If your target is on someone else's List, which is something they cannot control, we can see that with the following.

https://twitter.com/IntelTechniques/lists/memberships

Once you identify a Twitter List of interest, you can dig deeper into the members. As an example, I am included on a List called Open Source Intelligence, which I identified from the previous URL. Clicking that List opened a new page with the following URL.

https://twitter.com/i/lists/1588523704245489664

1588523704245489664 is the List number. The following two URLs display the members of the List (the accounts included within the list) and the followers of the List (people who monitor the List for content).

https://twitter.com/i/lists/1588523704245489664/members
https://twitter.com/i/lists/1588523704245489664/followers

Twitter Moments are also fairly new and seldom used. They encourage users to offer a story which would otherwise be inappropriate for a limited text Tweet. The structure is as follows.

https://twitter.com/IntelTechniques/moments

Twitter Topics is a feature which can quickly identify the interest of a Twitter user. When users add topics for their feeds, we cannot see their choices within their profiles. However, the following URL displays them all within one place.

https://twitter.com/IntelTechniques/topics

Next, we can focus only on tweets from a user which include a link to additional media. This could be an image, video, or website. The following presents only tweets with links from my own account.

from:IntelTechniques filter:links

We can also filter for popular posts. The following searches for any tweets from my own account which received at least 150 likes.

from:IntelTechniques min_faves:150

The following searches for any tweets from my own account which received at least 100 replies.

from:IntelTechniques min_replies:100

Finally, we can easily display a user's Followers (the people monitoring their profile) and Following (the people the target monitors) with the URLs below.

https://twitter.com/IntelTechniques/followers
https://twitter.com/IntelTechniques/following

Now, let's combine a few searches. The following query would identify any tweets sent to our target (IntelTechniques), from protonprivacy, during the month of March of 2018, which included a link, and were replies from another post.

to:IntelTechniques from:protonprivacy since:2018-03-01 until:2018-03-31 filter:links filter:replies

Replicating this search reveals only one result, which may have been otherwise buried within the thousands of posts. Hopefully you can see the value of targeted queries. **Later in the chapter, I present the custom offline Twitter search tools which automate most of these queries.** However, it is vital to understand why these work in order to explain your methods when required.

Search by Location

If you are investigating an incident that occurred at a specific location and you have no known people involved, Twitter will allow you to search by GPS location alone. The Twitter Advanced Search allowed us to search by zip code, but that can be too broad. The following specific search on any Twitter page will display Tweets known to have been posted from within one kilometer of the GPS coordinates of 43.430242,-89.736459.

geocode:43.430242,-89.736459,1km

There are no spaces in this search. This will be a list without any map view. They will be in order chronologically with the most recent at top. The "1km" indicates a search radius of one kilometer. This can be changed to 5, 10, or 25 reliably. Any other numbers tend to provide inaccurate results. You can also change "km" to "mi" to switch to miles instead of kilometers. If you want to view this search from the address bar of the browser, the following page would load the same results.

https://twitter.com/search?q=geocode:43.430242,-89.736459,1km

You can add search parameters to either of these searches if the results are overwhelming. The following search would only display Tweets posted at the listed GPS coordinates that also mention the term "fight". Notice that the only space in the search is between "km" and "fight".

geocode:43.430242,-89.736459,1km "fight"

It would be inappropriate to finish this section without a discussion about the lack of geo-enabled Tweets. Several years prior, this search would have been highly productive, as an alarming number of Twitter users were unknowingly sharing their locations with every post. Today, it is the opposite. The default option for Twitter is NOT to share location. A user must enable this option in order to appear within these search results. In my experience, catching a criminal from a location-enabled Tweet is extremely rare. However, we should be aware of the possibility.

Mandatory and Optional Search Terms

You may have a scenario that requires a search of both mandatory and optional terms. Twitter does not provide a published solution for this. However, it does support this type of search. Assume you are investigating threats against your target named Michael Parker. You believe that people may be tweeting about him with reference to violence. Searching his name alone produces too many results. Placing the name within quotes forces Twitter to only give you results on those exact terms, which is your "mandatory" portion. Additional optional words could be added with the term "OR" between each. This term must be uppercase, and will only require one of the optional words be present within the search results. Consider the following query on Twitter.

"Michael Parker" kill OR stab OR fight OR beat OR punch OR death OR die

Date Range Search

If you are searching vague terms, you may want to filter by date. This option is now available on the advanced search page, but I believe it is important to understand how Twitter performs this task. Assume that you are investigating a bomb threat that occurred several weeks, months, or years ago. A search on Twitter of the terms "bomb threat" will likely apply only to recent posts. Instead, consider a date specific search. The following query on any Twitter page would provide any posts that mention "bomb threat" between January 1, 2015 and January 5, 2015.

since:2015-01-01 until:2015-01-05 "bomb threat"

My favorite use of this search technique is to combine it with the "to" operator or a name search (or both). This allows you to go further back in time than standard profile and Twitter feed searches typically allow. Consider an example where Twitter user humanhacker is your target. You can visit his live Twitter page and navigate back through several thousand Tweets. However, you will reach an end before obtaining all Tweets. This could be due to Twitter restrictions or browser and computer limitations. He currently has 13,000 Tweets. Even if you could make it through all of his Tweets, you are not seeing posts where he is mentioned or messages sent publicly to him. I recommend splitting this search by year and including mentions and messages directed toward him. The following search within Twitter displays all Tweets from the Twitter name humanhacker between January 1, 2012 and December 31, 2012.

from:humanhacker since:2012-01-01 until:2012-12-31

This may create a more digestible collection of Tweets that can be collected and archived appropriately. There may be no other way of identifying these messages since you cannot likely scroll back that far. In my investigations involving targets with several thousand posts, I conduct multiple searches within Twitter that span several years. The following would collect yearly sets of Tweets posted by humanhacker since 2006.

from:humanhacker since:2006-01-01 until:2006-12-31
from:humanhacker since:2007-01-01 until:2007-12-31
from:humanhacker since:2008-01-01 until:2008-12-31
from:humanhacker since:2009-01-01 until:2009-12-31
from:humanhacker since:2010-01-01 until:2010-12-31
from:humanhacker since:2011-01-01 until:2011-12-31
from:humanhacker since:2012-01-01 until:2012-12-31
from:humanhacker since:2013-01-01 until:2013-12-31
from:humanhacker since:2014-01-01 until:2014-12-31
from:humanhacker since:2015-01-01 until:2015-12-31
from:humanhacker since:2016-01-01 until:2016-12-31
from:humanhacker since:2017-01-01 until:2017-12-31
from:humanhacker since:2018-01-01 until:2018-12-31
from:humanhacker since:2019-01-01 until:2019-12-31
from:humanhacker since:2020-01-01 until:2020-12-31
from:humanhacker since:2021-01-01 until:2021-12-31
from:humanhacker since:2022-01-01 until:2022-12-31

This same technique can be modified to display only incoming Tweets to humanhacker for these years. Replace "from" with "to" to obtain these results. The 2008 posts would appear as follows.

to:humanhacker since:2008-01-01 until:2008-12-31

You can combine all of these options into a single result, but I only recommend this after you have attempted the more precise options mentioned previously. While the next search should theoretically display all of his outgoing Tweets, incoming Tweets, and mentions, it is not always complete. The following would include 2008.

"humanhacker" since:2008-01-01 until:2008-12-31

There are many uses for a date range search. Any supported Twitter search should work combined with dates. This might include a location search for a specific date related to an investigation. As a test of the possibilities, consider that you want to identify an email address for this target. His live Twitter page will not reveal this, as he no longer posts his email, likely to prevent spam. However, the following search is quite productive.

from:humanhacker email since:2006-01-01 until:2009-12-31

This isolates only his posts from the beginning of Twitter until the end of 2009. Only four results are present, including the Tweet as seen in Figure 11.01 (top). These queries do not need to encompass an entire year. You could use this technique to focus on a specific month or day. Figure 11.01 (bottom) tells me what Chris was up to on October 2, 2017 with the following query.

from:humanhacker email since:2017-10-02 until:2017-10-03

Figure 11.01: An old Twitter post including an email address of the target.

Deleted, Suspended, and Missing Tweets

Twitter users may delete their own accounts if there is suspicion that an investigation is under way. If this happens, searching on Twitter will not display any of the posts. Furthermore, a person might only delete individual Twitter posts that are incriminating, but leave non-interesting posts on the profile to prevent raising suspicion associated with deleting an entire account. Some users may find their accounts suspended for violating Twitter's terms of service. In any of these scenarios, it is still possible to retrieve some missing posts using various techniques.

If I encounter a Twitter user that has recently deleted some or all of their messages, I conduct a cache search of their profile. There are various ways to do this, and I will demonstrate the most common. In this example, I conducted a search on Twitter for "deleted all my Tweets" on December 15, 2017. This provided many users who posted that they had just deleted all of their content. This helped me identify a good target for this type of demonstration. The first user I located was "WestCornfield". He had one Tweet and it referenced deleting all of his posts, and it is seen in Figure 11.02.

I attempted a search on Twitter of from:WestCornfield, which provided no results. I conducted a search of to:WestCornfield, which provided dozens of incoming messages from his friends. This was a good start. I then went to Google and conducted a search for "Twitter WestCornfield". The first search result was a link to the user's live Twitter page. Instead of clicking on the link, I chose the Google Cache view of his profile by clicking the small green "down arrow" next to the URL and selecting "Cached". This view identified twenty deleted Tweets from this account. Two of these posts can be seen in Figure 11.03. Google identified this capture as taken on December 12, 2017.

This may be enough for your investigation. Occasionally, I need to identify content that was deleted weeks or months before my investigation. The previous technique will likely not provide much assistance because the Google Cache is probably a recent copy of their live page. The cache may be missing Tweets you want to see. I next replicated this process on Bing and Yandex. Bing's cached view was taken on December 7, 2017 while Yandex's cached view was collected on November 3, 2017. Each of these possessed unique posts and images. Figure 11.04 displays a recovered post from Bing. Next, I returned to Google to obtain further data. I searched the following, which provided only results that possess a URL that begins with twitter.com, then my target's username, then "status". This will force Google to present direct links to actual posts.

site:twitter.com/westcornfield/status

The result was 56 posts. When I clicked on each of these, Twitter informed me that the post had been deleted. However, opening the cached version from the Google result displayed each of the posts. In Figure 11.05, you can see that Google is now identifying deleted posts as far back as October 2017. This process should also be repeated using the cached view options of Bing and Yandex. Next, we should check the Wayback Machine as mentioned in Chapter Nine. If you recall, you can search their archives by keywords or direct URL. The following address connects us directly to their archive of his account.

http://web.archive.org/web/*/twitter.com/WestCornfield

This identified a capture of his profile on December 6, 2017. Opening this archive displayed his Twitter profile dating back to November 8, 2017. Figure 11.06 displays this deleted Tweet.

While our target removed his content from his profile, he did not remove his history. In order to see the Twitter posts that he had previously liked before wiping out his page, we can navigate to the following URL. In this example, we see the hundreds of messages that identify his interests.

https://twitter.com/WestCornfield/likes/

While every investigation is unique, I wanted to demonstrate the importance of checking every source. These searches took less than three minutes using my custom Twitter search tool discussed in the next section. While you will likely never rebuild an entire deleted account, the posts obtained with this technique are something you did not have before.

Figure 11.02: A live Twitter post announcing Tweet deletion.

Figure 11.03: A Google cached Twitter posts recovered after deletion.

Figure 11.04: A Bing cached Twitter post recovered after deletion.

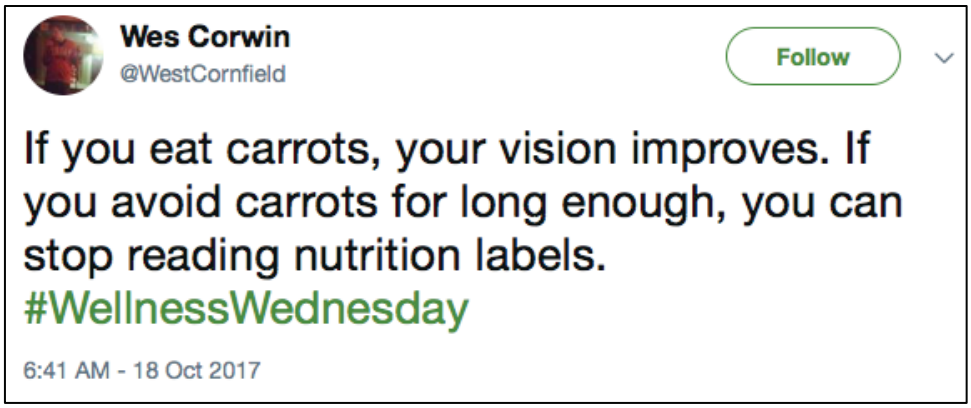


Figure 11.05: A Google cached Twitter message URL of a deleted post.

Figure 11.06: A recovered deleted Tweet from the Wayback Machine.

Twitter Post Details

Assume that you have identified an individual Tweet of interest. The URL of the message will appear similar to https://twitter.com/IntelTechniques/status/1318928116253814785. You have the tools to create a screen capture, but you may want a few more details. If there is an image embedded into the Tweet, you can click on it to see a larger version. However, this is sometimes not the original image size. In order to see the original full-size version, right-click the image and choose "View image". This will load a new URL such as the following.

https://pbs.twimg.com/media/Ek3FxAhVcAEJt_t?format=jpg&name=small

After removing "&name=small" from the URL, the full image is available. You can save this image by right-clicking and choosing "Save image as". This example is as follows.

https://pbs.twimg.com/media/Ek3FxAhVcAEJt_t?format=jpg

If we return to the message, we can see the date and time of the post directly below the content. In this scenario, it was posted on October 21, 2020 at 8:52 AM. This is likely local time to the account owner, but we should never assume this. If we right-click on the Tweet and choose "View Source", we can search within the source code of that Tweet. While in your browser, strike "Ctrl" + "F" (Windows/Linux) or "command" + "F" (Mac) and search for "time_zone". This should result in one entry. In this case, it reads "Eastern Time (US & Canada),utc_offset:-14400,tzinfo_name:America/New_York". We now know this Tweet was sent at 8:52 AM EST. While this may seem minor, it could be a huge deal if your investigation makes its way to a courtroom. Knowing exact dates and times is mandatory.

Tweet Deck (tweetdeck.twitter.com)

Tweet Deck is owned by Twitter, and it can take advantage of the live Twitter "Firehose". This huge stream of data contains every public post available on Twitter. Many Twitter services do not have access to this stream, and the results are limited. Tweet Deck requires you to create and log in to an account to use the service. The "Create Account" button on the website will walk you through the process. Alias information is acceptable and preferred. The plus symbol (+) in the upper left area will add a new column to your view. There are several options presented, but the most common will be "Search" and "User". The "Search" option will create a column that will allow you to search for any keywords on Twitter. The following is a list of search examples and how they may benefit the investigator.

"Victim Name": A homicide investigator can monitor people mentioning a homicide victim.
"School Name": A school can monitor anyone mentioning the school for suspicious activity.
"Subject Name": An investigator can monitor a missing person's name for relevant information.
"Event": Officials can monitor anyone discussing a special event such as a festival or concert.

The "User" option will allow you to enter a Twitter username and monitor all incoming and outgoing public messages associated with the user. If several subjects of an investigation are identified as Twitter users, each of the profiles can be loaded in a separate column and monitored. Occasionally, this will result in two of the profiles communicating with each other while being monitored.

You can also use the Geo search mentioned earlier within Tweet Deck. A column that searches "geocode:43.430242,-89.736459,1km" will display a live feed of Tweets posted within the specified range. A more precise search of "geocode:43.430242,-89.736459,1km fight" would add the keyword to filter the results. Figure 11.07 displays Tweet Deck with several searches.

The columns of Tweet Deck are consistently sized. If more columns are created than can fit in the display, the "Columns" option with left and right arrows will provide navigation. This allows for numerous search columns regardless of screen resolution. This is an advantage of Tweet Deck over the other services discussed. Tweet

Deck is one of my Twitter staples. I use it at some point during every investigation. I recommend familiarizing yourself with all of the features before needing to rely on it during your searches.

Real World Application: In 2019, I was investigating a death threat toward a celebrity. I launched Tweet Deck and began monitoring. First, I created a search column with "to:myclient". This started a stream of numerous people mentioning my client, which was too much to monitor. Next, I created a column of "to:myclient kill OR die OR shoot OR death". This presented very few tweets being sent to the celebrity including specific hateful words. However, it did identify a suspect. A tweet was sent to the celebrity stating "I hope you die a fiery death tonight". I then created another column of "from:suspect to:myclient". This identified every tweet he was sending to the celebrity. Since I had to move on to other resources, I set one final Tweet Deck column of "from:suspect to:myclient kill OR die OR shoot OR death" and added an alert. This instructed Tweet Deck to play an audible sound if any new messages met the criteria. The same suspect was arrested a week later while attempting to burglarize her apartment.

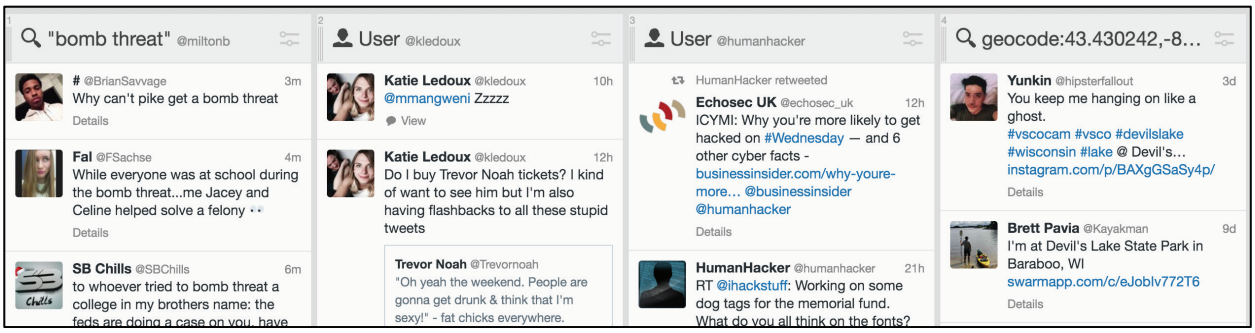


Figure 11.07: A Tweet Deck search screen.

Third-Party Resources

We have reached the end of the Twitter search options within the official website. However, we are far from done. Next, we will focus on third-party search tools. Most of these services require you to authenticate with a Twitter account before any queries can be submitted. This is largely due to Twitter's safeguards against fraud and abuse. I always recommend using a "junk" Twitter account when this is required. You will be asked to give the third-party service access to and control of the account in order to complete any tasks. This can be a security and privacy risk if using your true personal account. I will never give any service access to my personal Twitter account, but I often allow services to authenticate with one of my investigation accounts.

All My Tweets (allmytweets.net)

This website provides a clean display of all of a user's Twitter posts on one screen. It will start with the most recent post and list previous posts on one line each. This view will display up to 3,200 messages on one scrollable screen. This provides two analysis methods for digesting large amounts of data. Pressing "Ctrl" + "F" on the keyboard will present a search box. Any search terms can be entered to navigate directly to associated messages. The page can also be printed for archiving or distribution to assisting analysts. This is my preferred way of reading through the Tweets of an active user. This also prevents you from constantly loading more Tweets at the end of every page throughout a profile. Currently, I prefer TweetBeaver for this task, which is explained next.

TweetBeaver (tweetbeaver.com)

This is currently my absolute favorite third-party Twitter reporting tool. It is the most robust online option available to us for exporting content from an account or researching associations. There are currently fourteen unique options within this site, and I will explain each of them with usage scenarios. Note that you must be

logged in to a Twitter account for any of these to work, as they all leverage the Twitter API in order to function. Please only use covert accounts, and never your own personal Twitter login information.

Convert Name to ID: This is the most reliable way to take a Twitter username, such as jms_dot_py, and convert it to the corresponding user number, which is 817668451. This can be vital for investigations. Users can always change their username at any time, but the user number cannot be modified.

Convert ID to Name: This is the opposite of the above technique. If you had previously identified jms_dot_py as your target Twitter account only to find later that he had changed his username, you could easily locate his profile by providing the assigned user number (817668451).

Check if two accounts follow each other: As the title implies, this option quickly sorts out whether two users follow each other. An actual output appears below.

@inteltechniques does not follow @jms_dot_py
@jms_dot_py follows @inteltechniques

Download a user's favorites: This is the first tool where we can choose to either display the results on the screen or download them as a CSV spreadsheet. This option simply extracts a user's favorites (or likes) as discussed earlier. The results include the original author, date and time, text of the message, a direct URL to the post, and the author's bio, as seen in Figure 11.08.

Tweet author	Date posted	Text text	URL	Tweet author's biography
@TerrorFanatics	Wed Dec 06 02:20:06 +0000 2017	👍 https://t.co/S4MDS86jLf	www.twitter.com/TerrorFanatics/statuses/938231545235636225	Tweeting (and retweeting) the best #Horror articles, videos, memes, writers, podcasts, filmmakers, etc.

Figure 11.08: A TweetBeaver result for a user's favorites.

Search within a user's favorites: If the previous technique produces too many results, this option allows you to filter by keyword. Since you could search within the file you downloaded or on the screen of results, I find this feature to be of little use.

Download a user's timeline: This may be the most useful of all of these options. Provide a target Twitter name and TweetBeaver will extract the most recent 3,200 posts from the account. Furthermore, it will include the date and time of each post and the direct URL to the message. When I have a Twitter target of interest, I run this tool on the account daily. It has helped me obtain the new posts every day, and identify previous posts deleted after my initial acquisition. Figure 11.09 displays the first line of a result.

Tweet author	Date posted	Text text	URL
@jms_dot_py	Fri Dec 15 20:58:36 +0000 2017	@_prasket @TeriRadichel thanks so much you guys!	www.twitter.com/jms_dot_py/statuses/941774513942945792

Figure 11.09: A TweetBeaver user timeline result.

Search within a user's timeline: Similar to the favorites search tool, I find this one redundant.

Get a user's account data: This utility provides a great summary of the account information publicly available from any Twitter account. The benefit of this method of obtaining the data is that it is quick and presented in a

standard view. I can collect this information about many users, and all results will have the same format. This can aid in presentation during prosecution. Figure 11.10 displays the actual result from this target.

Screen name	Twitter ID	Name	Biography	Account created date
jms_dot_py	817668451	Justin Seitz	Creator of @Hunchly. Blogging & training #OSINT techniques. Wrote a couple of @nostarch books. @Bellingcat contributor. @C4ads fellow.	Tue Sep 11 15:44:20 +0000 2012
Location	**URL**	**Time zone**	**Geo enabled**	**Language**
Saskatoon, Saskatchewan	http://automatingosint.com/blog	Central Time (US & Canada)	not set	en
Verified	**Tweets**	**Followers**	**Friends**	
not verified	9041	7095	2112	

Figure 11.10: A TweetBeaver user account export.

Bulk lookup user account data: Similar to the previous, but allowing bulk submissions, as demonstrated in a moment.

Download a user's friends list: This option collects a target's list of accounts that he or she follows. This is similar to a typical friends list on Facebook, but approvals on either end are not required.

Download a user's followers list: This is a list of the people who follow the target on Twitter. This is less likely to contain actual friends, but all associated accounts should be investigated.

Find common followers of two accounts: This is a newer feature which can quickly identify people of interest based on co-targets. Consider the following. Twitter user jms_dot_py has been identified as a suspect in your embezzlement investigation. He seems to be very friendly with Twitter user humanhacker, and he once discussed tax evasion within public posts with this new second person of interest. Humanhacker has 33,000 followers and jms_dot_py has 12,500. There is no great way to look through these in order to find other suspects. However, TweetBeaver quickly identified the 371 people who follow both targets. This can be much more manageable, but is still a lot of data.

Find common friends of two accounts: Similar to the above, this only looks at the "friends" of each target. In other words, the people each target follows on Twitter. At the time of this writing, TweetBeaver identified the 57 people of interest. The following were two of the accounts.

Justin Seitz	@jms_dot_py	ID 854674794482216960
Michael Bazzell	@IntelTechniques	ID 257644794

Find conversations between two users: Once you have identified two people of interest, you can focus on the conversations between them. You could replicate this with the previous Twitter operators, but this route presents a nice spreadsheet of the results. In my scenario, I located six conversations between me and jms_dot_py. Figure 11.11 displays one result.

@jms_dot_py	Fri Nov 23 16:24:48 +0000 2018	@IntelTechniques @replyall @AGoldmund @frankahearn Not gonna lie, I am fanboying a bit over @AGoldmund 😊	https://twitter.com/jms_dot_py/statuses/1066004649721954304	Twitter Web Client

Figure 11.11: A message filtered by TweetBeaver.

Find accounts in a user's followers and friends list: This newer feature allows us to isolate only the Twitter profiles which follow our target and our target follows back. This might reduce the profiles which need investigated.

Find accounts followed by one user that also follow another named user: This feature is very specific, but could be helpful. Assume you wanted to find any profiles which are followed by jms_dot_py, which also follow IntelTechniques. You would enter these two profiles in that order, and would be notified that 59 profiles currently fit this scenario. You would also receive immediate full details of each account, as displayed in Figure 11.12. We now know that this user is followed by Justin and follows me.

Name	Screen name	Twitter ID	Location	Biography	Account created
David Bombal	davidbombal	ID 56191110	England	Free content and courses: YouTube: https://t.co /xGPVEj6ULN Discord: https://t.co /GZI30F45va Website: https://t.co /MpcS9ylBrV	Sun Jul 12 21:06:30 +0000 2009
Followers	**Following**	**Tweets**	**Favorites**	**Website**	**Time zone**
97230	489	10253	25434	https://www.davidbombal.com	
Geo enabled	**Verified**	**Language**	**Protected**	**Profile image URL**	
Disabled	Not verified		Not protected	https://pbs.twimg.com/profile_images/1499344306460573697/i-qaz8vr_normal.jpg	

Figure 11.12: Follower information from TweetBeaver.

Get the first 25 users to follow an account: If your target has thousands of recent followers, you may want to go back in time and learn of the first 25 people who began monitoring the profile. This can help identify true friends versus internet strangers. An analysis of my own account reveals the first person to follow me was Mike Downey (Thanks Mike).

Get the first 25 accounts followed by a user: Similar to the previous option, we can identify the first 25 people our target began following on Twitter. I now know the first person jms_dot_py followed was the Saskatchewan Roughriders.

Bulk Account Data Example

In early 2018, TweetBeaver introduced the bulk lookup feature which can accept up to 15,000 Twitter usernames. This is quite impressive, and I have put it through many tests. This feature allows you to input numerous accounts within one query. As a demonstration, I added every Twitter account mentioned in this chapter up to this point into the TweetBeaver bulk lookup utility. The entry appeared as follows.

Inteltechniques
jms_dot_py
SultryAsian
HydeNS33k
humanhacker
WestCornfield

This immediately created a CSV spreadsheet which was downloaded to my computer. Figures 11.13 and 11.14 display screen captures of the data. This gives me an immediate view into the accounts. If I had hundreds or thousands of Twitter usernames, this would allow me to sort by location or popularity. I could also sort by creation date in order to identify newly created accounts. This is possibly the most useful third-party tool when you have numerous accounts of interest.

Screen Name	Twitter ID	Name	Description	Account created date
@screen_name	ID 257644794	Michael Bazzell	Open Source Intelligence (OSINT) Training and Tools. International Privacy Consultant.	Fri Feb 25 21:46:04 +0000 2011
@screen_name	ID 817668451	Justin Seitz	Creator of @Hunchly. Blogging & training #OSINT techniques. Wrote a couple of @nos	Tue Sep 11 15:44:20 +0000 2012
@screen_name	ID 3308415728	SultryAsian	Known to most as M. Not as serious as I look.	Fri Aug 07 03:49:08 +0000 2015
@screen_name	ID 7711241370	Jek Hyde	Infosec Auntie ðŸ‡¨ Red Team Analyst at Fortune 1 ðŸ•µï¸ Thief ðŸ¯ Security Quality	Wed Aug 31 23:14:55 +0000 2016
@screen_name	ID 46998400	HumanHacker	This is the official Twitter account of all things SEORG - The SEVillage, SEPodcast, and t	Sun Jun 14 00:47:39 +0000 2009
@screen_name	ID 368971022	Wes Corwin	The Dean Malenko of Stand-Up Comedy and The Kazuyuki Fujita of Roast Comedy. See	Tue Sep 06 14:58:56 +0000 2011

Figure 11.13: Bulk Twitter data provided by TweetBeaver.

Location	URL	Time Zone	Geo-enabled	Language	Verified	Tweets	Followers	Following
Washington D.C.	http://inteltechniques.com	Eastern Time (US & Canada)	not set	en	verified	266	5456	0
Saskatoon, Saskatchewan	http://automatingosint.com/blog	Central Time (US & Canada)	not set	en	not verified	9163	7316	2123
not set	not set	Pacific Time (US & Canada)	not set	en	not verified	2248	1894	622
Dallas, TX	not set	not set	not set	en	not verified	3820	12817	404
USA	http://www.social-engineer.org	not set	not set	en	not verified	11498	25274	273
Dallas, TX	http://wescorwincomedy.wordpress.com	Pacific Time (US & Canada)	enabled	en	not verified	42	2725	2193

Figure 11.14: Bulk Twitter data provided by TweetBeaver.

Real World Application: The day this feature was released, I was investigating a suspicious Twitter account associated with violent threats toward a celebrity. The suspect had sanitized the account which prohibited obtaining valid location data. However, the suspect had numerous followers and people he was following. Using TweetBeaver, I could extract these accounts easily and supply them to the bulk lookup utility. I then sorted his friends by location which revealed a strong presence in a specific Midwest city. Of those target accounts of interest, I could see that a few were Geo-enabled. This provided home addresses for two people.

The person search tools described later identified full names and telephone numbers of these two friends. Group photos found online of these targets identified one person always nearby, but without a Twitter account associated with this group. The same photos on Facebook identified my new target by name. This process led to a positive identification of my suspect. Bulk search tools help tremendously. If you only use one third-party Twitter analysis tool, I recommend TweetBeaver. Hopefully, future API changes will not block this service.

Followerwonk (followerwonk.com)

This service offers options unavailable through TweetBeaver. The second tab at the top of the page, titled "Compare Users", will allow you a thorough search. Figure 11.15 displays the analysis of three subjects. You can see that the first and second subject do not have any people in common that they follow on Twitter. This can indicate that they may not know each other in real life, or that they simply have different tastes in the people whom they find interesting. However, the first and third subjects have 79 people in common that they follow on Twitter. This is a strong indication that they know each other in real life and have friends in common. Clicking on the link next to this result will display the identities of these people as seen in Figure 11.16.

This default search on Followerwonk is a good start. A more valuable search is to analyze the people who follow these users. The previous example identified people whom our targets followed. This will often include celebrities, businesses, and profiles that probably have no impact on your investigation. However, the people who follow your targets are more likely to be real people who may have involvement in your investigation. Figure 11.17 displays the results of the same targets when the search criteria was changed to "Compare their followers" in the drop-down menu next to the search button. We now see that the first and second subject still have no one in common. The first and third subjects have 200 people who follow both of their Twitter feeds. You can click the result link to identify these 200 people.

Followerwonk possesses other search capabilities for user analysis. The first tab at the top of the screen will search any term or terms to identify any Twitter bios that contain those words. This may identify profiles that were missed during the search on twitter.com for messages. Twitter's People option can be great when you know the exact name that a target used when creating an account. If you are unsure of the real name, or if the target has a very common name, Followerwonk can help you identify the profile that you are seeking. This service allows you to search Twitter profiles for any keyword that may help you locate a profile of interest. You can choose the default "Twitter Bios Only" option or switch to "Search Twitter Profiles". The "More Options" under the main search box will display numerous fields including Location, Name, and Follower details. A search of "John Smith" reveals 21,156 Twitter profiles. However, a search of "John Smith" from "New York City" reveals only 125 profiles. Filtering out profiles that have not posted at least 100 Tweets reveals only 44 profiles. This is a manageable number of profiles that can be viewed to identify the target.

The third tab, titled "Analyze", allows you to enter a single Twitter handle and either analyze the people the user follows or the people who follow that user. The second option usually provides more relevant results. The information provided during this search will display numerous pie charts and graphs about the user. The most useful is the map that identifies the approximate location of the people connected to the person's Twitter account. Figure 11.18 displays a map for one of the users searched previously. This provides a quick indication of the regions of interest to the target. Figure 11.19 displays a detailed level of an area near Denver. Each small dot identifies an individual Twitter account of a person that follows the target and lives or works in the general area. This location data is very vague and does not usually correctly correlate with the address on the map. This should only be used to identify the general area, such as the town or city, of the people who are friends with the target on Twitter. In the past, I have used this data to focus only on people in the same area as my homicide victim. I temporarily eliminated people who lived in other states and countries. This helped me focus on subjects that could be contacted quickly and interviewed.

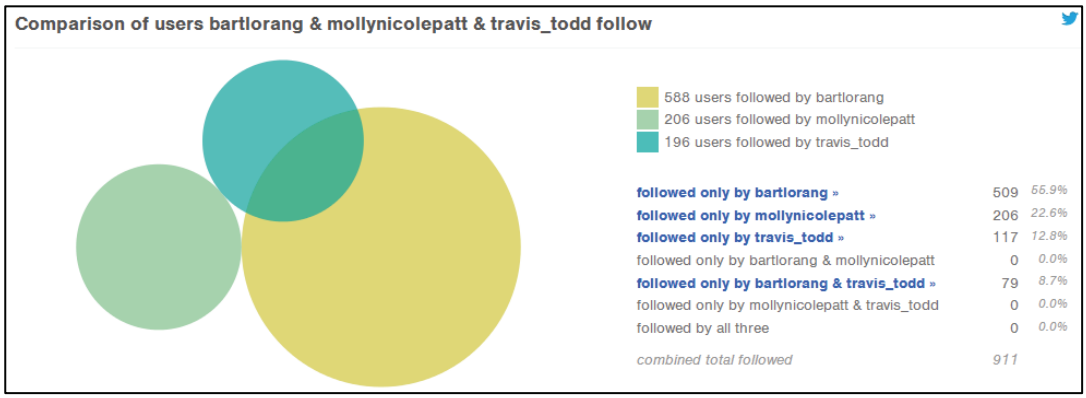


Figure 11.15: A Followerwonk user comparison.

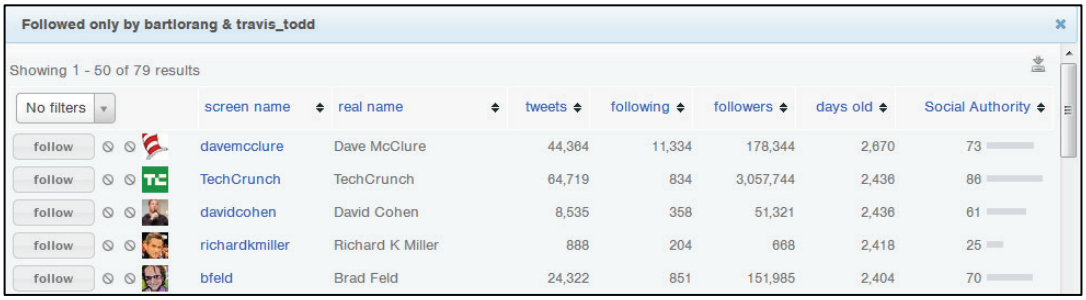


Figure 11.16: A Followerwonk list of users.

In 2019, Twitter began blocking access to user analytics if the account was marked as "private". Due to this change, Followerwonk will likely fail on any protected accounts. If this happens, you will receive a warning in the upper portion of this page stating, "This report contains information on protected accounts. To view, you must be an authorized follower". Fortunately, most Twitter profiles are publicly visible.

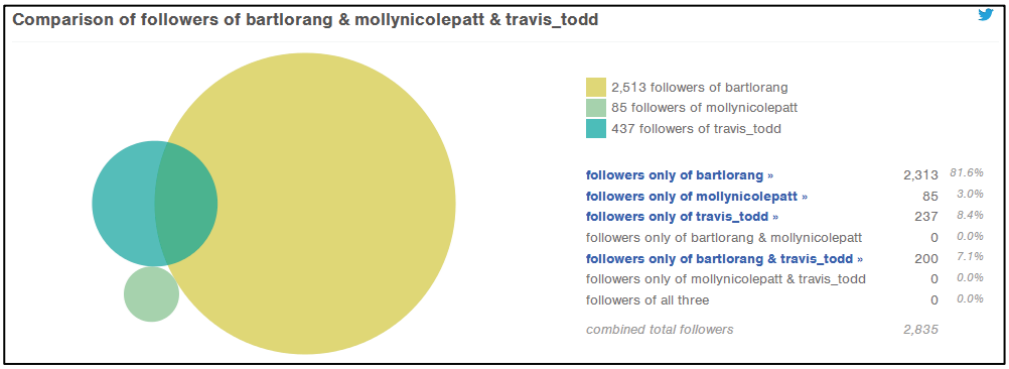


Figure 11.17: A Followerwonk user comparison.

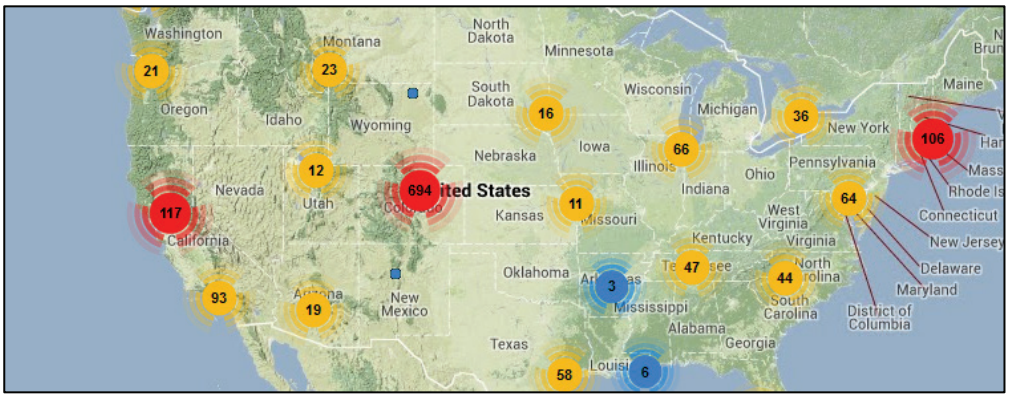


Figure 11.18: A Followerwonk map of users connected to a target.

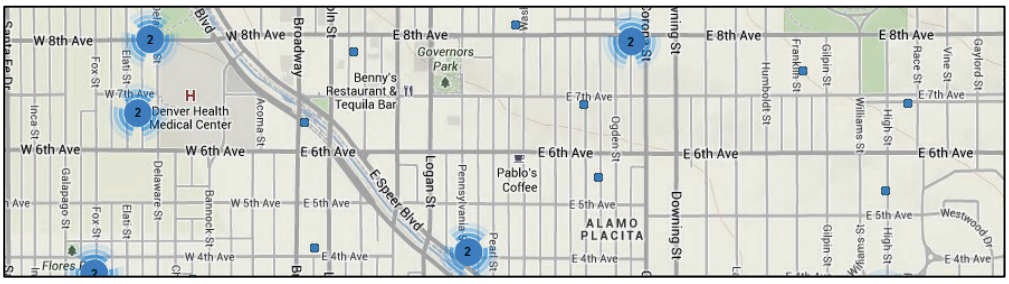


Figure 11.19: A detailed view of a Followerwonk map of connected users.

Social Bearing (socialbearing.com)

This robust solution combines the features of many resources within this chapter into one search. It relies on the Twitter API, so it will only analyze the most recent 3,200 Tweets. While searching my own account, I received a summary which included the following. Investigative benefits are next to each.

- Total audience reach: This tells me whether the target has true followers or "fakes".
- Total impressions: This tells me an accurate size of the target's audience.

- Total ReTweets: This discloses if the target's audience engages with the content.
- Total audience favorites: This confirms engagement from the target's audience.
- Tweet sentiment: This indicates positive or negative tone within comments.
- Tweet types: This identifies new content versus ReTweets of others.
- Tweet sources: This discloses the source of submission, such as mobile, desktop, or API.
- Domains shared: This summarizes the web links posted within Tweets.
- Word cloud: This provides a summary of the most common words posted.

The graphical output of this resource is impressive, but the CSV export is more useful. Additionally, we can query an account directly via static URL. This allows us to include this resource in our automated tools. My account can be seen at the following URL.

https://socialbearing.com/search/user/inteltechniques

On a final note about Twitter, I believe it is under-utilized by most investigators. We tend to find our target's profile on Twitter, scroll a bit, and quickly dismiss it. In my experience, the information most valuable to my case is never present in this live official view.

Twitter Biography Changes (spoonbill.io)

Similar to the way that users delete Tweets and comments, they also modify the information within their Twitter biography on their profile page. Several sites have come and gone which attempt to record these modifications, and my current favorite is **Spoonbill**. Use of this free service from the home page requires you to log in to your Twitter account. However, a direct URL query will display any stored results. If you were researching my Twitter handle, the following address would bypass the account login.

https://spoonbill.io/data/inteltechniques/

This page displays several changes I made to my account on 9/8/16, including changing my name and location. Searching more active people will usually reveal many pages of changes, and will always display the previous content as stricken, and highlight any changes in green. I do not recommend creating an account to use this service, it will demand access to the people you follow, which could jeopardize your investigation.

Real World Application: In 2017, I assisted a law enforcement agency with a missing juvenile case. Authorities had suspicion initially that she may have run away, but became more concerned when all of her social networks became dormant and her cell phone stopped being used. Her Twitter profile possessed no valuable Tweets, and I could not find any deleted content to recover. I supplied her name to Spoonbill and immediately received a log of changes to her Twitter profile. Two months prior to her disappearance, she listed her Snapchat username on her biography. This led me to her Snapchat profile photo. This photo and username had not been present in any of my previous investigations, and revealed a whole new world of leads. The username was associated with an email address, which was used by the missing person to create another Facebook profile. The username from Facebook revealed her user ID number, which was used in the Facebook search tools mentioned previously. The "hidden" images connected to this account provided many interesting suspects. Within an hour, a single suspect had emerged from the new discoveries, and the juvenile was located safely nearby. I cannot overstate the necessity to retrieve modified and deleted content during every Twitter investigation.

Twitonomy (twitonomy.com)

One Twitter analytics website that stands out from the rest is Twitonomy. This is the most complete analytics service that I have found for a single Twitter handle. A typical user search would fill four pages of screenshots. A search of one user revealed the following details.

He has posted 8,689 Tweets
He is following 170 people
He has 17,079 followers
He joined Twitter on June 14, 2009
He averages 5 Tweets per day
He has mentioned 4,175 other Twitter users
He replies to 34% of posts to him
He has ReTweeted 621 posts (15%)

The remaining sections of this page identify current posts, followers, people following, favorites, and lists. The main analytics portion identifies the average number of posts by day of the week and by hour of the day. It also displays from which platforms the user Tweets. This data discloses that the target has an Android and an iPhone, and that the majority of his Twitter time is spent on a Mac computer. This also identifies his preferred web browser, check-in utility, photo sharing service, and video sharing service. Other information includes his Tweets that are most "favorited" and "ReTweeted"; the users to whom he replies most often; and the users whom he mentions more than others.

Twitter Location Information

While privacy-aware individuals have disabled the location feature of their accounts, many users enjoy broadcasting their location at all times. Identifying a user's location during a Twitter post is sometimes possible through various methods. Prior to 2014, identifying the GPS details of every post of many users was simple. Today, most of these identification techniques are no longer working. A manual Twitter search method for identifying posts by location was explained earlier. That technique is best for current or live information, and is limited to only recent posts. You may have a need for historical details from previous posts from a specific location. I have had better success with historical data than current content in regard to geolocation. I believe this is because most people unknowingly shared their location while Tweeting for many years. When Twitter changed the default location option to "Disabled", most users never intentionally re-enabled the feature. Fortunately, there are third-party websites that collected this historical data, and can assist with easy searching. The following options will work best when you are investigating events that occurred several years prior. While you may get lucky and receive some recent posts, the majority could be quite old.

Omnisci (www.heavy.ai/demos/tweetmap)

Omnisci is a massive database platform developed through collaboration between MIT and Harvard. Historically, each college had their own interface into this data, which supplied Twitter post locations from past Tweets. Each interface provided new ways of searching information. Both websites have been disabled, and the entire project has warped into Omnisci. This website can search by topic, username, or location. It can also combine all three options to conduct a detailed search. Results appear as blue dots on a dark map. Each dot represents a Tweet which possesses location data to the chosen area.

One Million Tweet Map (onemilliontweetmap.com)

This service only displays the most recent one million Tweets on an international map. They do not have access to every Tweet available, often referred to as the "firehose", but they offer new Tweets every second. I would never rely on this map for complete data about a location. However, monitoring a large event can provide live intelligence in an easily viewed format. I recommend using a mouse scroll wheel to zoom into your location of interest. Once you are at a level that you can see single Tweets, you can click any of them to see the content. The page will automatically refresh as new information is posted.

Tweet Mapper (keitharm.me/projects/tweet)

If your target possesses Tweets with location data, but the previous two options failed to locate anything, this map could reveal the data. The project appears to be abandoned, but I receive occasional results**. Social Bearing** (socialbearing.com), which was previously explained, also offers geo-search options.

Fake Followers

There are a surprising number of Twitter accounts that are completely fake. These are bought and sold daily by shady people who want to make their profiles appear more popular than they really are. I have seen a lot of target profiles that have been padded with these fake followers. There are two websites that will assist in distinguishing the authentic profiles from the fraudulent. They both require you to be logged in to a Twitter account, and I will compare the results of each.

SparkToro (sparktoro.com/tools/fake-followers-audit)

The most robust option is SparkToro. Analyzing my own account, it declared that 28% of my followers were "accounts that are unreachable and will not see the account's tweets". It also provided some metrics and a full explanation as to how it achieves its results. This is something the others do not disclose. Once you have logged in to a Twitter account, you can query further users at the following static URL.

https://sparktoro.com/fake-followers/inteltechniques

Twitter Audit (twitteraudit.com)

This option identified 9% of my followers as "fake" and provided very few details. It also allows for submission through a static URL, but you still need to request an audit once you get to the page. The following would display results for my own page.

https://www.twitteraudit.com/inteltechniques

Miscellaneous Twitter Sites

Every week, a new site arrives that takes advantage of the public data that Twitter shares with the world. These sites offer unique ways of searching for information that Twitter does not allow on their main page. This partial list is a good start to finding information relevant to your target. I encourage readers to follow my blog, podcast, and Twitter account for any updates which surface after this book has been published.

Sleeping Time (sleepingtime.org)

This site allows for a search of an exact Twitter profile name, and provides the average time period that this user sleeps. The historical Tweets are analyzed according to the times of posts. Data is then presented that suggests when the user is usually sleeping due to lack of posting during a specific time period. A query of Kevin Mitnick revealed that he is likely to be sleeping between 12am and 7am according to his Tweets. Although the idea was probably executed as a fun site, it can be quite useful.

Twiangulate (twiangulate.com)

Earlier, I explained how I use TweetBeaver to filter most of my Twitter friend and follower data. If it should become unavailable, there are two additional websites which can assist. Twiangulate identifies mutual friends on two specific accounts. In one example, 521 people were friends with one of my subjects. However, only 15 were friends with both targets of my investigation. This can quickly identify key users associated within an inner circle of subjects. All 15 subjects were listed within the results including full name, photo, bio, and location. While Twiangulate has assisted me in the past, I now recommend Followerwonk as a superior solution when you have multiple suspects.

Tinfoleak (tinfoleak.com)

This Twitter analytics tool provides a simple yet thorough report. The website requires that you log in to your Twitter account through this service, and the report includes the following information. My own can be seen at https://tinfoleak.com/reports2/inteltechniques.html.

Twitter ID #
Account Creation Date
# of Followers
Following
Location
Time Zone
Number of Tweets
Twitter Clients Used
Hashtags
User Mentions
Metadata from Images
Geo-Location Data

FollerMe (foller.me)

This service is very similar to the previous Twitter analytics options. Providing a Twitter username presents the typical bio, statistics, topics, hashtags, and mentions analysis that you can find other places. I find the following option of most interest to my investigations. I previously explained Sleeping Time as a resource to learn about a target's usual sleep pattern based on posting times. FollerMe provides a bit more detail including posting patterns per hour. Note that the results are displayed in Universal Time, so you will need to convert as appropriate for your suspect. Since I am on the east coast (UTC -5), my own example indicates that I tend to never post before 8:00 a.m. or after 11:00 p.m. My peak Tweeting time is at 11:00 a.m. There is a very obvious sleep pattern present within this result.

TweetTopic (tweettopicexplorer.neoformix.com)

This simple tool provides one feature that I have found helpful in my investigations. Once you supply the target Twitter username, it collects the most recent 3,200 Tweets and creates a word cloud. This identifies the most common words used within posts by the target. There are several sites that do this, but this service takes it a vital step further. Clicking on any word within the result displays only the Tweets that include the selected term. Clicking on any of the text circles would immediately identify posts related to those terms. I have used this when I have a target with too many posts to read quickly. TweetTopic allows me to quickly learn what my target posts about and immediately delve into any topics of interest to my investigation.

Memory.lol (github.com/travisbrown/memory.lol)

This project is a web service that provides historical information about social media accounts. Currently, it can be used to look up 542 million historical screen names for 443 million Twitter accounts. Most of this data has been scraped from either the Twitter Stream Grab or the Wayback Machine. As an example, the following URL produces the result seen immediately below it.

https://api.memory.lol/v1/tw/IntelTechniques

```
accounts
0
id          257644794
id_str      "257644794"
screen_names
IntelTechniques
0           "2015-04-06"
1           "2022-11-08"
```

I now know that the first time this profile, number 257644794, was seen within these resources was April 6th, 2015. I can also reverse this query when I know the user number. The following URL queries by number and identifies the username and same account details, as seen following the URL.

https://api.memory.lol/v1/tw/id/257644794

```
id          257644794
id_str      "257644794"
screen_names
IntelTechniques
0           "2015-04-06"
1           "2022-11-08"
```

This provides a quick way to identify usernames from profile IDs and vice versa.

IntelTechniques Twitter Tool

I found myself using many of these manual Twitter techniques daily. In order to prevent repetitive typing of the same addresses and searches, I created a custom tool with an all-in-one solution. Download the archive previously mentioned or visit **https://inteltechniques.com/tools/Twitter.html** to access this resource. This page includes embedded JavaScript that will structure and execute web addresses based on your provided information.

In effort to reduce redundancy and minimize page count, I will not present a visual depiction of every custom tool at the end of each chapter, as I have done in previous editions. You can view every tool mentioned throughout this entire book within your offline downloads or online at **https://inteltechniques.com/tools**.

Any utilities that do not look familiar will be described in the remaining pages of this book. The Twitter tool will replicate many of the Twitter searches that you have read about here. The first option on the left side populates all of the search fields with your target's Twitter name when entered there. This should save time and avoid redundant pasting of data. Clicking the button next to each option executes the query in a new tab in your browser. As a reminder, this page collects absolutely no information from your searches. They are all conducted within your own browser, and no data is captured by my server.

CHAPTER TWELVE
SOCIAL NETWORKS: INSTAGRAM

While Facebook and Twitter are likely going to offer the most bang for your buck in terms of web-based social network presence, Instagram has captured a large share of the market over the past two years. Instagram is a photo-sharing service that is now owned by Facebook. With well over 1 billion active monthly users, the amount of content available here is overwhelming. This application works alone or in correlation with Facebook and Twitter to distribute the photos. This service is very popular with most photo sharing Twitter users and should not be ignored. My preferred method of downloading a person's entire Instagram profile is through the Python tools Instalooter and Instaloader, as previously explained in the Linux, Mac and Windows chapters. These utilities will download all the images from a profile, but there is much more data out there which may be valuable to your investigations. You will need to be logged in to an Instagram account for most techniques in this chapter. First, let's start with some basics.

Surprisingly, there is no search feature on the Instagram home page. If you want to search Instagram, you must connect directly to an account such as Instagram.com/mikeb. However, this search field only identifies users and hashtags related to the search terms. It does not provide a true keyword search. We will use search engines for this in a moment.

In previous editions of this book, I detailed several third-party Instagram search options that unlocked a lot of hidden content within user accounts. On June 1, 2016, Instagram tightened their API, and this killed most of the useful websites. My own Instagram tools page suffered drastically from the new restrictions in 2018 and 2020, and I had to start over with new options. Fortunately, you still have many search options for your next Instagram investigation. Let's start with keyword searching. I have found greater success with a custom Google search instead of an Instagram search field. The following query on Google will produce 963 results that display Instagram posts that mention "OSINT" within the post title or comments.

site:instagram.com "OSINT"

This same term searched on an Instagram page only displayed users that have the term within the username. When searching "#OSINT" on Instagram, I was provided a list of hashtags that include the keyword. Each of these hashtags are associated with multiple Instagram posts. Consider the following example in order to identify the benefits of searching away from the Instagram website. While on an Instagram page, I searched for my target "Darren Kitchen" from Hak5. This presented several profiles associated with people having that name, but none of them were my target. Instead, I went to Google and conducted the following search.

site:instagram.com darren kitchen hak5

This produced one result, but it was not connected to my target. Since people seldomly use last names on Instagram, I modified my search to the following.

site:instagram.com darren hak5

The first result was my target's profile (@hak5darren). Similar to Facebook and Twitter, a person's Instagram profile only tells a small portion of the story. Conducting the following search on Google revealed 215 results.

site:instagram.com "hak5darren"

These are the various pages and posts that contain the text of my target's username. Many of these are posts from the target, which we have already seen by looking at his profile. A few are from associates of the target

and may otherwise be missed. I now only want posts that contain "@hak5darren" within the content. This is more likely to find other people mentioning my target.

site:instagram.com "@hak5darren"

Similar to the previous search, it forces Google to ignore the target's profile, and only focus on people that are posting "to" the target with @hak5darren instead of just his username alone. These searches can be modified to include or exclude keywords, usernames, real names, or any other terms with which you have interest. You could also repeat these on Bing and Yandex for better coverage. Figure 12.01 displays the difference in search results while Figure 12.02 identifies a post targeted toward a user. Hopefully, you now have a target profile of interest. Now we can dig into our target within Instagram and through third-party resources.

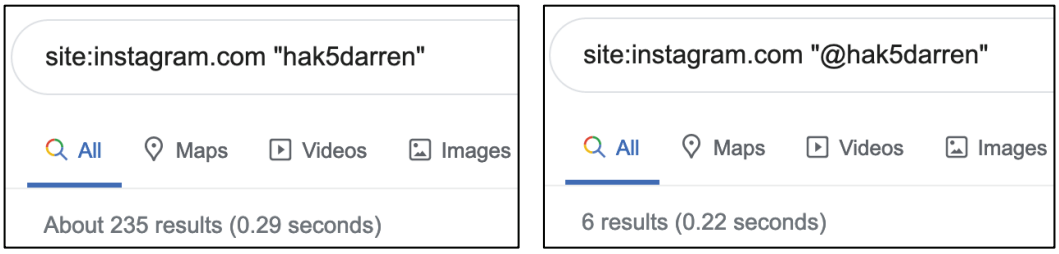


Figure 12.01: Google search queries for Instagram data.

Figure 12.02: Google search results of Instagram data.

Instagram Images

There is one obstacle with Instagram searches that needs to be addressed. When you identify a profile of interest, there will always be a profile image at the top of the page. This is usually a 150x150 pixel profile image and a full page of lower-resolution post images, all of which are small and heavily compressed. In 2015 and 2018, Instagram changed the way these thumbnails and original images are stored. The goal of this technique is to locate these low-resolution images and identify the full high-resolution images that were originally uploaded. For an example, I will pick on the user "hak5darren". His profile is at instagram.com/hak5darren. When I right-click on his profile picture, I am blocked from opening the image alone in a new tab. In fact, there is no right-click menu whatsoever. If I right-click on the images, I also receive no option to view the image alone, which previously would grant us a larger version. Instagram is actively blocking us, so we will need to dig into the source code of the page.

When I right-click anywhere else on the profile, I have the option to "View Page Source". This opens a new tab which displays the HTML source code (text) behind the profile page. I can also navigate directly to view-source:https://www.instagram.com/hak5darren/ to see this result. Within this text, I can search for "og:image" and receive only one result. The extremely long URL within this line should appear similar to the following.

https://scontent-ord5-2.cdninstagram.com/v/t51.2885-15/13534173_1620017484977625_1916767107_a.jpg?stp=dst-jpg_s100x100&_nc_cat=107&ccb=1-7&_nc_sid=8ae9d6&_nc_ohc=y-uchsMyXNsAX_5vNhj&_nc_ht=scontent-ord5-2.cdninstagram.com&oh=00_AfA78t3nlbeIEW-LpJBL9yMsIdUC4uVIw97itb5C_ng5OQ&oe=63753A41

Notice the section of "s100x100". This tells Instagram to only display a 100-pixel square icon of the target. This is a highly compressed thumbnail image. In previous years, we could remove that section and display the full-sized image, but that no longer functions. Instead, we can only enlarge the profile image slightly. In the previous URL, notice the number directly before ".jpg". The last set of numbers is "1916767107". Back in the page of source code for the profile, conduct a search for these numbers. You should receive three results, the third is a URL similar to the following.

https:\/\/scontent-ord5-2.cdninstagram.com\/v\/t51.2885-15\/13534173_1620017484977625_1916767107_a.jpg?stp=dst-jpg_s200x200&_nc_cat=107&ccb=1-7&_nc_sid=8ae9d6&_nc_ohc=y-uchsMyXNsAX_5vNhj&_nc_ht=scontent-ord5-2.cdninstagram.com&oh=00_AfDJgRUwCm4PkSk_o8VtVhGg7YcJu9DiyPayKC7KiYBeRQ&oe=63753A41

Copying this URL into your browser likely presents "Bad URL timestamp" or "Unsigned URL". This is because Instagram wants to make it difficult to obtain images with higher resolution. Fortunately, some minor modification of the URL returns the feature. If we simply replace every instant of "\/" within the URL with "/", it loads fine. This example would be the following.

https://scontent-ord5-2.cdninstagram.com/v/t51.2885-15/13534173_1620017484977625_1916767107_a.jpg?stp=dst-jpg_s200x200&_nc_cat=107&ccb=1-7&_nc_sid=8ae9d6&_nc_ohc=y-uchsMyXNsAX_5vNhj&_nc_ht=scontent-ord5-2.cdninstagram.com&oh=00_AfDJgRUwCm4PkSk_o8VtVhGg7YcJu9DiyPayKC7KiYBeRQ&oe=63753A41

This URL presents a profile image which is now twice as large as the original image provided within the profile (200x200). This is not the full-size image, but it possesses a higher resolution. Fortunately, we can do much better in regard to the images from his posts. When clicking on the first image on his profile, we are given a new URL of https://www.instagram.com/p/CR854ZigEhR/ which displays the image and comments. The image is highly compressed. Clicking on the image does nothing, and right-clicking gives us no options associated with photos. Instead, right-click on this post image and select "View Page Source" again. This opens a new tab similar to the following.

view-source:https://www.instagram.com/p/CR854ZigEhR/

Conduct a search within this text for "jpg". The third URL which contains "jpg" within it, which also includes "OG:Image" before it, should appear similar to the following.

https://scontent-ord5-2.cdninstagram.com/v/t51.29350-15/228704624_1648970191970827_5792204972766774242_n.jpg?stp=dst-jpg_s640x640&_nc_cat=105&ccb=1-7&_nc_sid=8ae9d6&_nc_ohc=bToTyKGo058AX_tI8Q-&_nc_oc=AQmy71YJjzxne3IeDZkOi-IJ8Hb9tpU-NnLnXRCJVLPpJEumH2gOiCJ0pmIz7h0gM8U&_nc_ht=scontent-ord5-2.cdninstagram.com&oh=00_AfCGGqa5HTvcAd_l9Uos1VKnFQSQZULX24Sy8KGOmTQrDQ&oe=6374F47D

If we replace every instant of "&" within the URL with "&", it loads fine. This example would be the following.

https://scontent-ord5-2.cdninstagram.com/v/t51.29350-15/228704624_1648970191970827_5792204972766774242_n.jpg?stp=dst-jpg_s640x640&_nc_cat=105&ccb=1-7&_nc_sid=8ae9d6&_nc_ohc=bToTyKGo058AX_tI8Q-&_nc_oc=AQmy71YJjzxne3IeDZkOi-IJ8Hb9tpU-NnLnXRCJVLPpJEumH2gOiCJ0pmIz7h0gM8U&_nc_ht=scontent-ord5-2.cdninstagram.com&oh=00_AfCGGqa5HTvcAd_l9Uos1VKnFQSQZULX24Sy8KGOmTQrDQ&oe=6374F47D

This image might be higher resolution, and could be all you need. However, I want the "HD" version of the image. The second to last (usually fifth) jpg file within this source code appeared as follows.

https:\/\/instagram.fbek1-1.fna.fbcdn.net\/v\/t51.2885-19\/44884218_345707102882519_2446069589734326272_n.jpg?efg=eyJybWQiOiJpZ19hbmRyb2lkX21vYmlsZV9uZXR3b3JrX3N0YWNrX3RpZ29uX21uc19oaWdoX3NhbXBsZV9yYXRlX3FwbF8zOmNvbnRyb2wifQ\\u0026_nc_ht=instagram.fbek1-1.fna.fbcdn.net\\u0026_nc_cat=1\\u0026_nc_ohc=tHpMVc_ZlbIAX8uAwil\\u0026edm=AEsR1pMBAAAA\\u0026ccb=7-5\\u0026ig_cache_key=YW5vbnltb3VzX3Byb2ZpbGVfcGlj.2-ccb7-5\\u0026oh=00_AfB90TAb2DABOx_f0hiGIpXLH2gsRCSavJnb_w7mpDAKGA\\u0026oe=6374F04F\\u0026_nc_sid=3f45ac\

This gave the message of "Unsigned URL". If we simply replace every instant of "\/" within the URL with "/", the example would be the following.

https://instagram.fbek1-1.fna.fbcdn.net/v/t51.2885-19/44884218_345707102882519_2446069589734326272_n.jpg?efg=eyJybWQiOiJpZ19hbmRyb2lkX21vYmlsZV9uZXR3b3JrX3N0YWNrX3RpZ29uX21uc19oaWdoX3NhbXBsZV9yYXRlX3FwbF8zOmNvbnRyb2wifQ\\u0026_nc_ht=instagram.fbek1-1.fna.fbcdn.net\\u0026_nc_cat=1\\u0026_nc_ohc=tHpMVc_ZlbIAX8uAwil\\u0026edm=AEsR1pMBAAAA\\u0026ccb=7-5\\u0026ig_cache_key=YW5vbnltb3VzX3Byb2ZpbGVfcGlj.2-ccb7-5\\u0026oh=00_AfB90TAb2DABOx_f0hiGIpXLH2gsRCSavJnb_w7mpDAKGA\\u0026oe=6374F04F\\u0026_nc_sid=3f45ac\

This might load fine, or might give the message of "Bad URL Hash". If we simply replace every instant of "\\u0026" within the URL with "&", it should load. This example would be the following.

https://instagram.fbek1-1.fna.fbcdn.net/v/t51.2885-19/44884218_345707102882519_2446069589734326272_n.jpg?efg=eyJybWQiOiJpZ19hbmRyb2lkX21vYmlsZV9uZXR3b3JrX3N0YWNrX3RpZ29uX21uc19oaWdoX3NhbXBsZV9yYXRlX3FwbF8zOmNvbnRyb2wifQ&_nc_ht=instagram.fbek1-1.fna.fbcdn.net&_nc_cat=1&_nc_ohc=tHpMVc_ZlbIAX8uAwil&edm=AEsR1pMBAAAA&ccb=7-5&ig_cache_key=YW5vbnltb3VzX3Byb2ZpbGVfcGlj.2-ccb7-5&oh=00_AfB90TAb2DABOx_f0hiGIpXLH2gsRCSavJnb_w7mpDAKGA&oe=6374F04F&_nc_sid=3f45ac\

This is often our best option for viewing and archiving. Once you load this URL in the browser, you can right-click on the image and easily save it. Does this seem like too much trouble for every image of interest on Instagram? Yes, no argument here. If you prefer to allow a third-party service to replicate this for you, the following sites will download the best quality image available within a post. Regardless of your method, you should understand the functions allowing for the capture of these high-resolution images. **I suspect that these methods will change slightly after this book is published.** However, the replacement strategies explained in the examples should carry over into the unknown changes Instagram will apply next.

https://instadownloader.co/
https://igram.io/
https://igdownloader.com/
https://toolzu.com/downloader/instagram/photo/

Metadata Details

You should also consider digging into the source code of your evidence in order to identify further details that could be valuable. First, I like to identify the user number of an account. Similar to Facebook and Twitter, people can change their username on Instagram, but not their user number. Right-click on your target profile page and select the option to view the source code. This will open a new tab with a lot of pure text. Using either "Ctrl" + "F" or "command" + "F", conduct a search for exactly the following. The numbers directly after this data will be the user number of your target. In our previous example, his user number is 340416780.

"profile_id"

Sometimes Firefox will search through the source code and highlight data, but the results don't match the terms searched. I do not know why this happens. When necessary, I copy all of the source code and paste

it into a text editor, then search there. Your target may change their username, but they cannot change their user ID. If you know that your target's user ID is 340416780, but you can no longer access their profile based on the username, we can query the Instagram API to get the new details. However, you must first spoof your user-agent. In Chapter Two, I explained the Firefox extension called User-Agent Switcher and Manager. Open this menu within your Firefox browser; change the second drop-down menu to "Android"; change the first drop-down menu to "Instagram"; select the first option; click "Apply"; and navigate to the following page.

https://i.instagram.com/api/v1/users/340416780/info/

The text-only result should display pages of details, but I find the following of most interest.

```
"pk": 340416780,
"username": "hak5darren",
"full_name": "Darren",
"is_private": false,
"media_count": 949,
"geo_media_count": 0,
"follower_count": 5855,
"following_count": 257,
"external_url": "http://hak5.org",
"has_videos": true,
"has_highlight_reels": true,
"has_guides": false,
"can_be_reported_as_fraud": false,
"is_business": false,
```

If the target had changed his username, we would see it here. While inconvenient, this is the most stable way to translate a user ID back into a username. Next, let's tackle exact timestamps. Instagram might tell you a photo was posted 67 weeks ago, and hovering over that notification should tell you the exact date, but it never tells you the time of the post. This detail can be crucial to a legal investigation. For this example, I located an image posted by our target at the following address.

https://www.instagram.com/p/CR854ZigEhR/

Right-click the profile and select the "Inspect" option within your browser. Click on the "Network" tab and reload the page. Click the small magnifying glass to conduct a search (not the filter URL option). Search for "taken" and notice the data which follows. My example displayed the following.

"items":[{"taken1627648763,"pk":"2629230846841735249","id":"262923

The number following "taken", which was 1627648763 is a Unix timestamp. Converting this from Unix to standard time on the website located at epochconverter.com reveals the image was uploaded on Friday, July 30, 2021 at 12:39:23 in universal time (GMT) format.

I realize these details are very minor. However, documenting this content can announce the difference between an average OSINT investigator and an expert. If you plan to serve any legal process later, knowing the user number and exact time of activity will be crucial.

Hashtags

A hashtag is a word or phrase preceded by a hash sign (#) used on social media websites and applications, especially Twitter and Instagram, in order to identify messages on a specific topic. If I wanted to view all posts tagged with "osint", the following direct URL would be appropriate.

https://www.instagram.com/explore/tags/osint/

I do not pay much attention to hashtags within my investigations. I prefer to focus more on search terms. Many people do not properly tag a post, and you can miss a lot by limiting your search to a hashtag. If I search for hashtags of #shooting, I may see valuable data. If a potential target only posts the term "shooting" without the "#", I would miss the post. This is why I rely much more heavily on searching within Google, as explained momentarily.

Followers & Following

Instagram now requires users to be logged in to an account in order to view the followers of a target or the profiles that a target follows (friends). Viewing these lists is not a challenge, but proper documentation can be tricky. The following explains how I choose to view and document every Instagram follower and friend of my target account, using hak5darren as an example.

After logging in to your account and navigating to the target profile, you will be able to simply click on "Followers" or "Following". In this demonstration, I chose the people he is "following", often referred to as friends. This opened a new window on my screen with the first twenty people displayed. Since hak5darren follows 215 people, I can scroll down the entire list to load all of the accounts. You will not be able to see all of the accounts at once because the window only displays ten within its boundaries. This causes screen capture tools to be useless in this scenario. If you are using either the Firefox browser we created within Linux or your own version of Firefox with the recommended Add-ons, you can easily copy this list.

After scrolling down to load all of the users, press either "Ctrl" + "A" (Windows & Linux) or "command" + "A" (Mac) on your keyboard. This will highlight or "select" the entire list. Next, right-click within this window and choose "Copy selected links". This copies all of the Instagram account hyperlinks within the target's following list into your computer clipboard. Open your desired documentation software, such as Word, Excel, or any text editor, and paste the contents into the application. I prefer Excel (or Calc within the free LibreOffice suite) for this task. Pasting the results confirmed 199 accounts, four of which are shown in Figure 12.03. Repeat the process with "Followers".

196	https://www.instagram.com/dragorn/
197	https://www.instagram.com/smacktwin/
198	https://www.instagram.com/serafinamoto/
199	https://www.instagram.com/moon_spot/

Figure 12.03: A partial spreadsheet of people being followed by the target.

In my experience, Instagram limits the number of accounts within this window to 1,000. This should be sufficient for most friends of the target, but may be limited when looking at the followers of famous people. While Instagram offers a text-only view of followers and following, it appears limited to 50 random profiles. I included these long links within the search tools, but I do not rely on them. I prefer this manual method. If your target's user ID was 340416780, the following URL presents the first 50 people he is following.

https://www.instagram.com/graphql/query/?query_hash=d04b0a864b4b54837c0d870b0e77e076&variables={%22id%22:%22**340416780**%22,%22include_reel%22:true,%22fetch_mutual%22:false,%22first%22:50}

As you can see, this is quite messy and only presents partial information.

Likes & Comments

Similar to Twitter, users can "Like" or comment on each post. Unique from Twitter, this is only seen when you load the target post in your browser. In other words, if a user posts a comment to a photo published by our target, we do not see that comment within the account of the person who made it. We only see it within the

specific post directly from the target. We also cannot search within Instagram by keyword in order to identify comments, but we can with Google as explained momentarily. First, let's focus on Likes.

Viewing an individual Instagram post reveals a heart icon below any comments. Clicking that icon "likes" the post from your account, so this should be avoided. Directly below the heart is a summary such as "557 Likes". Clicking on this opens a new window, which identifies each account that liked the post. Slowly scrolling through this list expands it until all accounts are present. Similar to the previous method of capturing friends and followers of a target, a manual approach is best here. Pressing "Ctrl" + "A" or "command" + "A" on the keyboard selects all of the content, and "Ctrl" + "C" or "command" + "C" copies it to your computer memory, which can be pasted into a spreadsheet or word processor. You can repeat this process for the comments within the post.

If you locate a comment which is valuable to your investigation, you may want to document the date and time of the post. This is not available within the standard view. As an example, look at the post at https://www.instagram.com/p/Bqi6r9uA0Y3. Inside the comments includes a response of "now I gotta go buy some beer" from a friend. At the time of this writing, Instagram displayed "207w" under the post, indicating it was posted 207 weeks previous to the current date. This is not sufficient for our needs. While on the page, right-click and choose the "Inspect" option as we did previously. While on the "Inspector" tab, search for the terms (now I gotta go buy some beer) and expand the text immediately after the result, as displayed in Figure 12.04. You can now see that this comment was made on November 24, 2018, at 14:27:02 GMT.

Figure 12.04: "Like" data within an Instagram post.

Complete Post Analysis

Assume that you have located your suspect (https://www.instagram.com/hak5darren) and found an incriminating post on his account (https://www.instagram.com/p/BK1KWEthQkb). This is a high-priority investigation, and extensive manual documentation is justified. You have already attacked his account with Instaloader, Instalooter, and Osintgram as explained in previous chapters. You possess all of the images from the account. The following outlines the next steps I would take toward this target.

- View the source code of https://www.instagram.com/p/BK1KWEthQkb.
- Copy all jpg URLs and apply modifications as previously explained.
- Paste the modified URLs into a browser and download the full version of the images.
- Return to https://www.instagram.com/p/BK1KWEthQkb.
- Scroll through the comments, and expand any, if necessary, by clicking "+".
- Click the summary of likes below the heart icon.
- Scroll until all are loaded, select all with "Ctrl" + "A" or "command" + "A" on keyboard.
- Open a new spreadsheet through Microsoft Office or LibreOffice and paste the results.
- Rename this tab "BK1KWEthQkb" and open a new tab.
- Repeat the process with any other posts of interest.
- Return to the account (https://www.instagram.com/hak5darren) and click "Followers".
- Load the data, copy it, and paste it into the new sheet titled "Followers".

- Repeat the process for "Following".
- On the target profile, scroll down until all posts are loaded and select/highlight all.
- Right-click, choose "Copy selected links", and paste data into a new sheet titled "Posts".

You now have a spreadsheet with multiple pages inside it. The first provides the comments, details, and likes of a specific post. The second displays all of the target's followers. The third shows his friends, and the fourth provides the direct URL of every post for later analysis or extraction. Figure 12.05 displays a partial view of my example, highlighting the final sheet created. This was a lot of work, and I do not replicate these steps for every Instagram investigation. You must decide when the extra work is warranted. Unfortunately, there is no reliable automated solution which extracts all data.

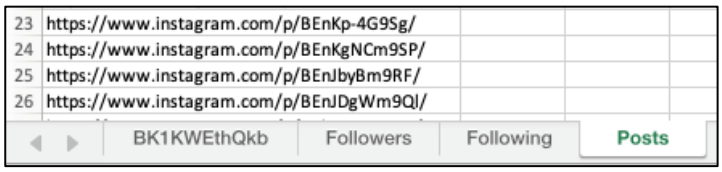


Figure 12.05: A spreadsheet created from a target Instagram post.

Channels

While Instagram was originally introduced as a photo-sharing platform, it is quickly becoming dominated by user-submitted videos. Many profiles now include a video channel page titled "IGTV". This can also be accessed directly via URL with the following structure.

https://www.instagram.com/ambermac/channel/

Opening a video page allows you to play the content and see any details such as likes and comments. Capturing the text data can be done using the techniques previously explained. However, saving the video introduces new challenges. While the video URL is embedded into the source code of the page, it requires heavy modification in order to replicate the content in a format which can be archived. Instead, we will rely on our browser to help us locate the stream. While viewing any Instagram video page, launch the "Inspector" tool within your Firefox or Chrome browser. You should be able to simply right-click on the page and select "Inspect Element". If this is not visible, you can launch Inspector from the Developer Tools within the browser menu.

Within Inspector, click the "Network" tab and play the video. If the video was already playing, reload the page. The inspector will display a lot of information which can seem overwhelming. Instead of manually filtering through these network connections, simply type "mp4" within the search field directly below the "Network" tab. This should eliminate all of the data and leave you with one URL. Right-click this URL and select "Open in New Tab". Figure 12.06 displays these options. The new tab within your browser should present the same video in full-size resolution. You can now right-click on the video and select the option to save it.

This may all seem tedious and complicated. While there are browser extensions which simplify these processes, they are replicating the manual methods presented here. Knowing these manual functions allows you to explain the capture process when mandated. When I testified in criminal trials about my online evidence, I proudly explained the ways I captured evidence from the source rather than relying on third-party tools which do not disclose their exact methods.

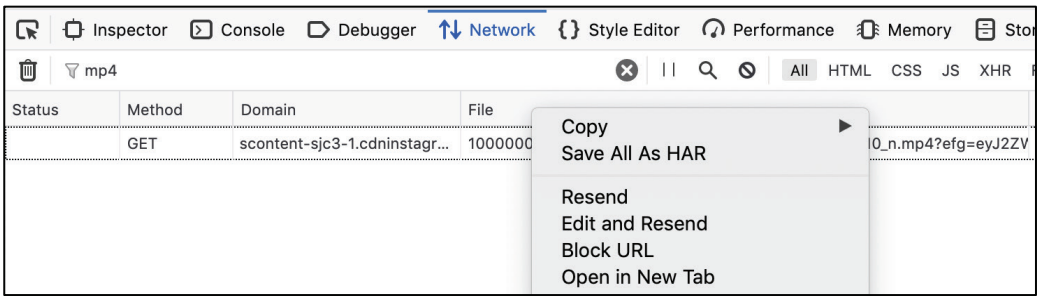


Figure 12.06: The "Inspector" utility within Firefox identifying an Instagram video MP4 file.

Tagged Users

Instagram allows users to be tagged within posts. On any target profile, you may see the "Tagged" link directly above the images. You can also navigate directly with the following structure.

https://www.instagram.com/hak5darren/tagged/

Google Instagram Search

We have exhausted all useful aspects of collecting data from within Instagram.com. The search functions are extremely limited there, so we must utilize a third-party search engine to find further data and connections. In my experience, Google is our best option, and you should be familiar with the queries conducted at the beginning of the chapter. Let's conduct a demonstration. Earlier, we searched site:instagram.com "hak5darren" and received 165 results. Browsing through these three pages reveals no evidence of a post including the word "Pager". However, I know he posted a photo with that term in the title. Searching the following reveals the target post.

site:instagram.com "hak5darren" "pager"

This confirms that Google indexed the post, but did not present it when I searched the suspect's Instagram username. This means that we should also query search terms when we conduct searches associated with a target. The search tools presented in a moment simplifies this, but let's conduct one more manual attempt. I see that "hak5darren" posts comments to "snubs" within his posts. I would like to search for any Instagram posts, within any accounts, which contain either a mention of or comment by both "hak5darren" and "snubs". The following is appropriate.

site:instagram.com "hak5darren" "snubs"

Twitter Instagram Search

Many people who post to Instagram broadcast the links on Twitter. Following this data can reveal unique comments from Twitter users. In rare scenarios, it can reveal Instagram posts from private accounts. Posts themselves are not private, only ACCOUNTS are private. If I know the exact URL of your "private" post, I should be able to see the content. If you post an Instagram link on your public Twitter account, that post is not private. If you post images to Instagram and have no followers, you may post the same link to Twitter where you have an audience.

Because of these reasons, we should always check Twitter for Instagram posts. The user "hak5darren" on Instagram is also "hak5darren" on Twitter. The following Google search identifies posts on Twitter (site:twitter.com) which mention "hack5darren", and possess an Instagram image URL (instagram.com/p).

site:twitter.com "hak5darren" "instagram.com/p"

Third-Party Tools

Overall, the various Instagram search websites do nothing more than what we replicated in the previous instruction. However, these sites could identify that one piece of vital evidence which we missed. Therefore, it is always best to have as many resources as possible. I have found the following third-party resources beneficial to my investigations. Third-party Instagram tools appear and disappear rapidly. After publication, I will continue to update the Instagram search tool with new features, as explained soon.

Instagram Stories

Instagram Stories are vertical photos or videos, up to 15 seconds in length, which disappear 24 hours after posting. Instead of being displayed in the user's feed, they are displayed at the top of an active user's app when they are logged in. Since Stories disappear quickly, they are often used much more casually than the public Instagram feed. I have found the following utilities helpful for displaying and downloading this temporary content. I demonstrate each.

InstaFollowers (instafollowers.co/download-instagram-stories)

Of these, Instafollowers has been the most beneficial. It was the only option which displayed "Stories Highlights" of previously expired content and easily allowed download of all stories directly through Instagram. Consider the following example. Using their search tool, I conducted a query for my target. Several "Stories" were presented with a "Download" button below each. That button forwarded me directly to the Instagram host of each video. Each link is quite long, but should load the video directly in your browser. Right-clicking should present the option to download the file.

StoriesDown (https://storiesdown.com/)

This service is very similar to the previous and should present most of the same data.

Downloader for Instagram (downloader-for-ig.com)

I realize this chapter presents many technical ways of acquiring Instagram user data which can be simplified with browser extensions and Python tools. Searching the Chrome store for Instagram extensions will provide many options. While I do not rely on these for my daily investigations, I understand the allure for easier solutions. If you want to install a browser extension which simplifies downloads of data, my preference is "Downloader for Instagram". After installation, you should see new download options within Instagram profiles and posts. Figure 12.07 displays our target's profile with a new green download option in the upper right. Clicking this presents a menu which allows download of all photos and videos from his account.

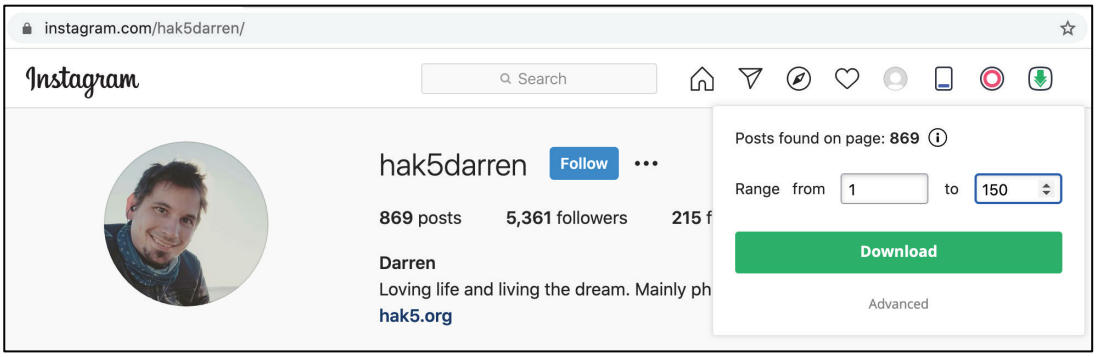


Figure 12.07: An Instagram profile with download options.

Clicking "Download" prepares an archive of the data which is presented as a single zip file. This file contains all of the images and videos within the account.

While this solution seems ideal for daily use, I have some bad news. I have witnessed suspension of Instagram accounts when using automated extensions such as this. Always log in to a "junk" account and expect that you may be blocked. Logging back in to the account and completing a Captcha usually reverses the suspension.

Dumpor (dumpor.com)

From an evidence perspective, I believe it is best to obtain your Instagram data from the official website. However, account requirements might hinder your investigation. Some organizations prevent employees from using any covert account to access data about a target. If logging in to Instagram is prohibited, consider Dumpor. This service allows you to navigate through a person's Instagram posts without the need for an account. You can search a username or tag from the main site, or submit a direct URL query as follows.

Username: https://dumpor.com/v/ambermac
Tag: https://dumpor.com/t/osint

Toolzu (toolzu.com)

In 2022, I found this site to be quite beneficial to Instagram investigations. It relies on the Instagram API, so you will not find any content which is not already public. However, the layout and analysis of profile data is unique. It also allows direct URL input, so it will play well with our tools. Consider the following.

https://toolzu.com/profile-analyzer/instagram/?username=ambermac

Toolzu creates a summary of the account, but then extracts all profile data and analyzes if to provide details which are not available directly from Instagram. In this example, I now know that my target:

Posted over 2000 Instagram uploads
Has 35,600 followers
Posts 1.29 times weekly
Averages 5.54 posts per month
Is most likely to post on a Tuesday or Wednesday
Most commonly tags with Partner, TheFeed, CanadasSeniorCare, Seniors, and AmberMacMedia
Most commonly captions with newsletter, Canada, Tuesday, and technology
Interests include business (15,38%), travel (3,85%), marketing (3,85%), and art (3,85%)
Most commented post is https://www.instagram.com/p/CiSniiBgqKH/
Most liked post is https://www.instagram.com/p/CiSniiBgqKH/

IntelTechniques Instagram Tool

If this all seems like a lot of work for minimal information, consider using my custom Instagram Tool, which includes many of the search options previously discussed. It cannot replicate the source code techniques, but may save you time with username and keyword searching. The online version is available on my website at **https://inteltechniques.com/tools/Instagram.html**, and an offline copy is presented within your downloads portal.

CHAPTER THIRTEEN
SOCIAL NETWORKS: GENERAL

Facebook, Twitter, and Instagram will each provide more data than all other social networks combined. However, there are many other options, and all should be researched. While we see high usage on the most popular sites, we may not see much incriminating evidence. Instead of finding your suspect confessing to your investigation on Facebook or Twitter, you may be more likely to find his grandmother testifying to his innocence. Smaller networks often provide more intimate details. Your suspect may feel exposed on the bigger networks, but a bit more private on smaller websites.

I divide general social networks and online communities over the next two separate chapters. For OSINT purposes, I believe social networks are defined as generic areas which cater to a wide audience with various interests. We see much more general engagement between members within these platforms. People from all walks of life are on LinkedIn and Snapchat, regardless of their profession or age.

On the contrary, I believe online communities cater to a niche audience and are likely ignored by the masses which do not have a similar interest. I place TikTok and dating services in this category. Most people on Tinder are single and TikTok primarily consists of a young audience. These are specialty sites with which my mother is unfamiliar (I hope).

It is impossible to mention even a fraction of potential networks which may contain online evidence about your target. Instead, I focus on those which have large numbers of members and are common to most internet users. My goal was to present various investigation tactics within each popular community which can be carried over to networks which are not discussed.

As an example, the Google "site" queries which will be used to identify LinkedIn content can also be used for hundreds of other networks. Please watch for techniques instead of specific resources or links. I promise that the following methods will need to be adjusted over time, but the overall strategies should prove to be useful when we see changes in the search infrastructure from the networks. Let's start with the most popular options and work our way down to the niche sites.

LinkedIn (linkedin.com)

When it comes to business-related social networking sites, LinkedIn is the most popular. It is owned by Microsoft and currently has more than 675 million subscribers internationally. The site requires searchers to create a free profile before accessing any data. As with any social network, I recommend creating a basic account with minimal details. The search field of any page offers a search using a real name, company, location, or title. These searches will often lead to multiple results which identify several subjects. The site was redesigned in 2020, which provides new options. The upper center portion of any search result page will offer some basic refinements to the search to filter by People, Posts, Companies, Groups, Jobs, Schools, and Events. These can also be queried by direct URL. The following includes a summary of each filter and the direct search URL which we will add to our custom search tools.

People: (https://www.linkedin.com/search/results/people/?keywords=john wilson)
This is the most common filter which presents profiles of people. Further filtering includes location and employer. This URL would find people named John Wilson.

Posts: (https://www.linkedin.com/search/results/content/?keywords=john wilson)
This option is similar to Facebook and presents posts including the provided search terms. This helps find content about (or by) our target. Further filters include search by date, author, and industry.

Companies: (https://www.linkedin.com/search/results/companies/?keywords=john wilson)
This query strictly identifies companies which include the searched keywords. Further filters include location, industry, and company size.

Groups: (https://www.linkedin.com/search/results/groups/?keywords=john wilson)
This option presents LinkedIn groups which contain your keyword within the title or description. It does not search for member names.

Jobs: (https://www.linkedin.com/jobs/index/?keywords=john%20wilson)
This presents current job openings and includes numerous additional filters. While beneficial for employment-seekers, I find this less useful than the other queries when my target is a person. However, it can be quite useful when investigating a company. Network penetration testers can use this to identify software applications used by the client, which could lead to identification of vulnerabilities.

Schools: (https://www.linkedin.com/search/results/schools/?keywords=john wilson)
This queries schools with the keywords in the title.

Events: (https://www.linkedin.com/search/results/events/?keywords=john wilson)
This queries events with the keywords in the title.

Knowing the real name of your target will be most beneficial. The results page should include the target's employer, location, industry, and possibly a photo. After identifying the appropriate target, clicking the name will open that user's profile. If searching a common name, the filters will help limit options.

Profiles

The profiles on LinkedIn often contain an abundance of information. Since this network is used primarily for business networking, an accelerated level of trust is usually present. Many of the people on this network use it to make business connections. Some of the profiles will contain full contact information including cellular telephone numbers. This site should be one of the first stops when conducting a background check on a target for employment purposes. The target profile often contains previous employment information, alumni details, and work associates. Aside from searching names and businesses, you can search any keywords that may appear within someone's profile. Since many people include their phone numbers or email addresses in their profile, this can be an easy way to identify the user of that specific data. Visiting this profile identifies further information as well as confirmation of the target number. You can also search LinkedIn for a specific username, which may be directly associated with a profile. As an example, the following URL connects directly to a target.

https://www.linkedin.com/in/ambermac/

Posts

In years prior, LinkedIn was a place to create a profile and communicate directly with another person. Today, it is a true social network with Posts, Likes, and Comments. Conducting a keyword search through any LinkedIn page or my own custom tool will present anything applicable. From there, consider the following.

- Clicking the three dots within a post allows you to copy a static URL of the content, which is beneficial during documentation.
- Expanding all comments before generating a screen capture presents additional evidence within your documentation.
- Many posts are redacted to save space. Click "...see more" to enable this hidden content before generating a screen capture.

- Clicking the number next to the "clapping hands" icon presents a list of people interested in the post, which looks almost identical to the Instagram options.
- The Instagram capture techniques previously presented work the same for LinkedIn.

Timestamps

Much like other social networks, LinkedIn hides the dates and times behind generic announcements such as "posted 2 weeks ago". As online investigators, we need something more specific. Let's conduct a detailed example, which can be replicated across all of LinkedIn. Consider the post in Figure 13.01. The date of posting is "1w" in the upper-right corner, insinuating the post was published one week prior to today's date. In that image, I have already clicked the three dots in the upper-right which presented a new menu. I clicked "Copy link to comment", which was the following.

https://www.linkedin.com/feed/update/urn:li:activity:6996561887907692544?commentUrn=urn%3Ali%3Acomment%3A%28activity%3A6996561887907692544%2C6996572489061474307%29

Loading that URL in the browser presented the same comment, but it was technically a unique page from the original post which generated this response. Next, I right-clicked on the page and selected the option to "View Page Source", much like we did previously with Instagram. I searched through this code for the message, which is "Congratulations!" in this scenario. I then saw the following text, and I have bolded the important areas.

Congratulations!","$type":"com.linkedin.voyager.common.TextViewModel"}, "createdTime":**1668112871423**

The numbers represent Unix time, the same as with Instagram. We can visit the Epoch Converter site at epochconverter.com again to convert this number to GMT time, which happens to be Thursday, November 10, 2022 at 20:41:11. We now have a specific date and time of the post.

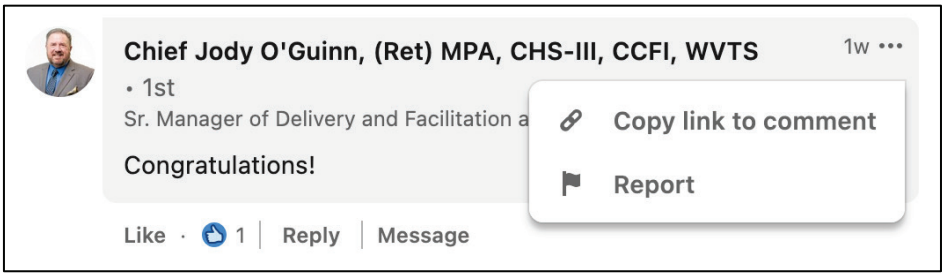


Figure 13.01: A generic timestamp on a LinkedIn post.

I believe this is the most reliable way to investigate this, but we have one additional option which does not require viewing any source code. Let's revisit the URL above, but break down the pieces. The first numbers (6996561887907692544) represented the original poster's activity number for the post. The second numbers (6996561887907692544) repeated this activity number again, but the third numbers (6996572489061474307) presented a unique activity number for the "Congratulations" response, which is our target. We can now create two unique addresses. The first URL below would present the original post. The second URL would present an error, but contains the timestamp of the "Congratulations!" response in the source code.

https://www.linkedin.com/feed/update/urn:li:activity:6996561887907692544
https://www.linkedin.com/feed/update/urn:li:activity:6996572489061474307

You could paste both of these URLs, or any other LinkedIn URL, into my LinkedIn timestamp tool, which is explained in a moment, to immediately see the timestamp of each post. My tool relies on JavaScript originally created at github.com/Ollie-Boyd. Figure 13.06 displays the result from this demonstration.

Searching by Personal Details

You might get lucky and find your target with a simple name search. Unfortunately, this is rarely the case. With hundreds of millions of profiles, LinkedIn must make some assumptions when choosing the profiles to display after a search. This is especially true with common names. Let's conduct several queries as part of a demonstration of LinkedIn's URL structure. Searching "John Smith" produces practically useless results at the following URL.

https://www.linkedin.com/search/results/people/?**keywords**=john%20smith

Instead of the default "keywords" parameter, let's force a change with the following URL.

https://www.linkedin.com/search/results/people/?**firstName**=john

Changing to "firstName" displays content associated with people named John. This is still fairly unhelpful and includes a lot of content which is not applicable to our target. Now, let's specify the full name of our target in the following URL.

https://www.linkedin.com/search/results/people/?**firstName**=john&**lastName**=smith

The results now only include links to profiles of people with the real name of John Smith. This is much more useful and may be all you need to identify your target. With a name such as John Smith, we need to go a few steps further. The following URL adds his employer (Microsoft).

https://www.linkedin.com/search/results/people/?**firstName**=john&**lastName**=smith
&**company**=microsoft

If you wanted to go further, we could specify his title (Manager) and school (Oklahoma) in the following URL. If you know these details about your target, you could start with this, but I find that providing too many details can work against you. Figure 13.02 displays the result of this URL, which identified the only person on LinkedIn which fit the criteria.

https://www.linkedin.com/search/results/people/?**firstName**=john&**lastName**=smith
&**company**=microsoft&**title**=manager&**school**=Oklahoma

The previous URL query is the most precise option we have. This has been the most beneficial structure I have found to navigate directly to my target. However, it can fail. If your suspect did not provide the school attended or current employer to his profile, you will not receive any leniency from LinkedIn within this search. However, we can rely on Google to help us. Your target may have mentioned an employer somewhere else within the profile or listed a school within the "Interests" area. The following reveals many profiles associated with the target.

site:www.linkedin.com john smith Microsoft manager Oklahoma

Searching by Company

If you are searching for employees of a specific company, searching the company name often provides numerous profiles. Unfortunately, clicking on any of these profiles presents a very limited view with the name and details redacted. The name of the employee is usually not available, but the photo and job description are usually visible. You are now required to upgrade to a premium account, or be in the same circles as the target, in order to get further information. Instead, consider the following technique. Search for the business name of your target company, or the employer of your target individual. I typed "Uber" into the search bar and received the official business page on LinkedIn. Clicking the "See all 88,788 employees on LinkedIn" link presented me with

numerous employee profiles such as those visible in Figure 13.03. Notice the names are redacted and only "LinkedIn Member" is available. Clicking this first result prompts me with "Profiles out of your network have limited visibility. To see more profiles, build your network with valuable connections". We struck out, but there are ways that you can proceed in order to unmask these details. First, copy the entire job description under the "LinkedIn Member" title. In this example, it is "Account Executive at Uber". Use this in a custom Google search similar to the following.

site:linkedin.com "Account Executive at Uber"

The results listed will vary from personal profiles to useless directories. Since Uber is such a large company, I had to view many pages of results until I identified my target. When I opened the 24th search result, the LinkedIn page loaded, and her photo confirmed it was the correct target. The easier way would have been to search the images presented by Google. After the above search is conducted, click on the Images option within Google and view the results. Figure 13.04 (left) displays a section, which easily identifies the same image as the LinkedIn target. Clicking this will load the profile page with full name and details.

Another way to accomplish this is to navigate through the profiles in the "People also viewed" column. These pages include other profiles viewed that are associated with whichever person you are currently analyzing. These people may not be friends or co-workers with your target, but there is a connection through the visitors of their pages. As an example, I returned to the Google search at the top of this page. I clicked on the first search result, which was not my target. However, in the "People also viewed" area to the right, I saw my target, including her full name and a link to her complete profile. Figure 13.04 (right) displays this result.

Finally, the last option is to conduct a reverse image search on the photo associated with the target's profile. Full details of this type of search will be presented later. For this demonstration, I will right-click on her photo and choose "Copy image location". On the Google Images page, I can click the camera icon and submit this URL. While the first result is not the target, clicking the page does present a link to the target's unmasked page. I will later explain many detailed ways to fully query reverse image search options within the upcoming Images chapter.

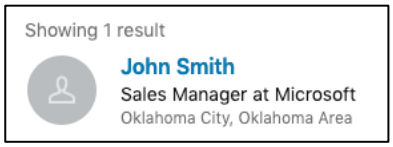


Figure 13.02: A LinkedIn result via direct URL.

Figure 13.03: Redacted employee results from a business search.

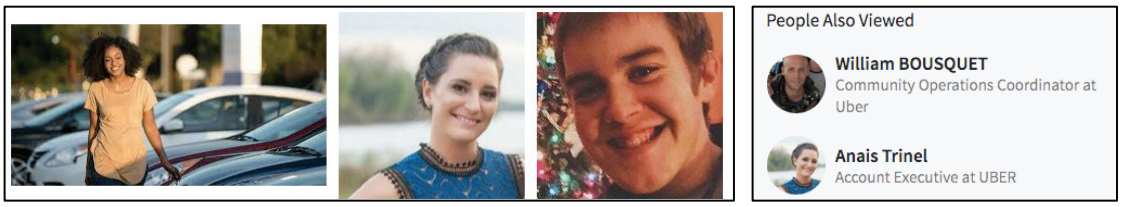


Figure 13.04: Google Images results (left) and un-redacted LinkedIn results (right).

Searching by Country

While LinkedIn is an American company, it is a global social network. If you know that your target is in a specific country, you can filter your search accordingly. This can be done manually by navigating to a foreign subdirectory such as uk.linkedin.com (UK), ca.linkedin.com (Canada), or br.linkedin.com (Brazil). Each of these pages query the entire LinkedIn network, but each places emphasis on local individuals.

PDF Profile View

You may want a quick way to collect the publicly available details from the profiles that you find. One option is to have LinkedIn generate a PDF of the information. While on any profile, click the "More…" button and choose "Save to PDF". This will not extract any private details, but will make data collection fast.

Google Search

When all else fails, go to Google. It scrapes and indexes most of LinkedIn's profiles and pages. The following search would identify profiles of our target, followed by the direct URL.

site:www.linkedin.com john smith microsoft
https://www.google.com/search?q=site%3Awww.linkedin.com+john+smith+microsoft

Google Images

The results may be overwhelming. Often, I know the face of my target and I simply want to browse images from LinkedIn. The following URL queries Google for any images (&tbm=isch) associated with my target's name (john+smith) on LinkedIn (site:linkedin.com). In a moment, we will replicate all of this with my tools.

https://www.google.com/search?q=site:linkedin.com+john+smith&tbm=isch

Google Videos

Many LinkedIn posts contain embedded videos. While a keyword search directly on LinkedIn may not find them, a query on Google should. The following URL replicates our image search, but focuses only on videos (&tbm=vid)

https://www.google.com/search?q=site:linkedin.com+john+smith&tbm=vid

Bing Search

While Google is typically our most powerful search engine, I prefer Bing for LinkedIn queries. Microsoft owns both Bing and LinkedIn, and indexes LinkedIn data well. The following search on Bing replicates our Google attempt, followed by the direct URL.

site:linkedin.com john smith microsoft
https://www.bing.com/search?q=site%3Alinkedin.com+john+smith+microsoft

Yandex Search

Finally, we should always consider Yandex as a LinkedIn search engine. The following search on Yandex replicates our Google attempt, followed by the direct URL.

site:linkedin.com john smith microsoft
https://www.yandex.com/search/?text=site%3Alinkedin.com+john+smith+microsoft

Viewing Without Account

Every LinkedIn method presented until now requires you to possess a valid LinkedIn account and to be logged in to see any content. Without an active account, all LinkedIn pages display a prompt to log in. This can be an issue if you do not have an account or your departmental policies prohibit the use of any credentials during an investigation. We can solve this with Google's **Mobile Friendly Test** (search.google.com/test/mobile-friendly).

Enter any LinkedIn URL, such as the profile located at http://linkedin.com/in/ambermac. The result appears as the mobile view of this profile including the target's name, image, and bio. Figure 13.05 (left) displays the results from the target LinkedIn URL without an account while Figure 13.05 (middle) displays the results of the same URL within Google's test page. We can also submit a direct URL of search.google.com/test/mobile-friendly?url=http://linkedin.com/in/ambermac within our browser to replicate this process. This is a great start, but the results are limited. The preview only displays a portion of the LinkedIn profile. Instead, consider the following.

- Click on the "HTML" tab above the profile on the test page.
- Click the "Copy" icon above the source code.
- Navigate to **CodeBeautify** (codebeautify.org/htmlviewer) and paste the HTML code.
- Click "Run" and view the entire LinkedIn profile without requiring an account.

Partial results are displayed in the highly compressed image in Figure 13.05 (right). The entire page is visible. You should be able to replicate this for any public LinkedIn page. This works because Google's servers have the authorization to bypass LinkedIn's credentialing requirements during their indexing of LinkedIn. Now, we do too.

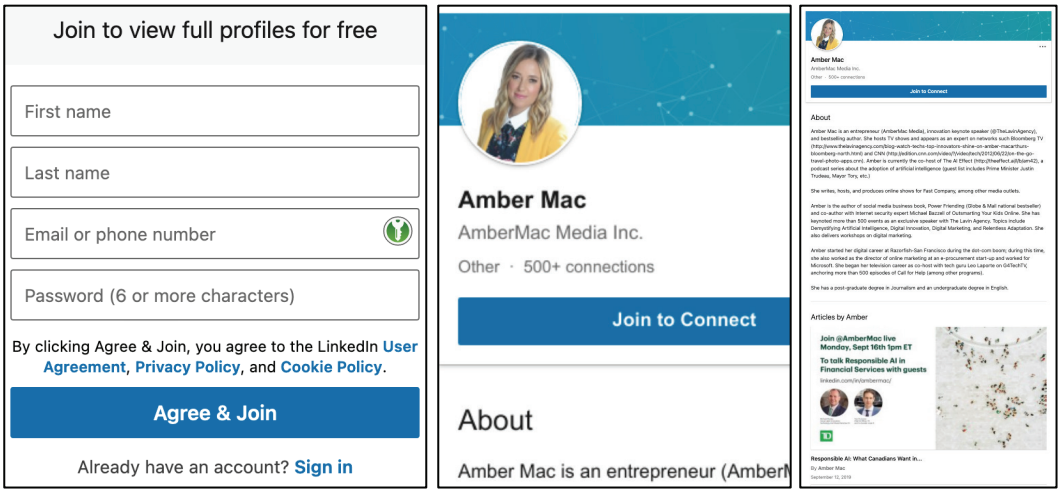


Figure 13.05: A LinkedIn profile login, mobile version bypass, and full profile view.

IntelTechniques LinkedIn Tool

These search techniques may seem complicated, but we can simplify them with our custom search tool. The "LinkedIn" section of the offline tool, which you previously downloaded, possesses several advanced search features. Most techniques mentioned in this section are available as automated queries within this tool. The online version is available at **https://inteltechniques.com/tools/Linkedin.html**.

Snapchat (snapchat.com)

Overall, Snapchat is a difficult OSINT resource. It is available officially as a mobile application, and there is minimal native web search. The majority of the content is set to auto-destruct after a specific amount of time, and is privately delivered from one user to another. In 2018, we started to see much more public content and in 2020 we saw better search options. While there are some extended search options within the app itself, I will only focus on traditional web resources here. Let's start with official options within the Snapchat website.

Keyword Search: (https://story.snapchat.com/search?q=osint) This queries the search term throughout Snapchat. I find it to fail often, but should be attempted.

Story Search: (https://story.snapchat.com/s/inteltechniques) This option loads the most recent Snapchat "Story" from the specified user. It then cycles through older stories from that user.

User Search: (https://www.snapchat.com/s/inteltechniques) This option may forward to the "Story" page, which can also serve as the landing page for that profile.

Snap Map: (map.snapchat.com) This interactive map allows you to query public "Stories" by location. In Figure 13.06, I have zoomed into LAX and I can click on any of the heat maps to load a post. You will see the approximate time in the upper left, but not much additional information. Viewing the source code of any story page will display a static URL to the post, but no additional details. I only find this service valuable when monitoring a high-profile event at a specific location. The search feature allows entry of a location in order to quickly zoom to an area of interest.

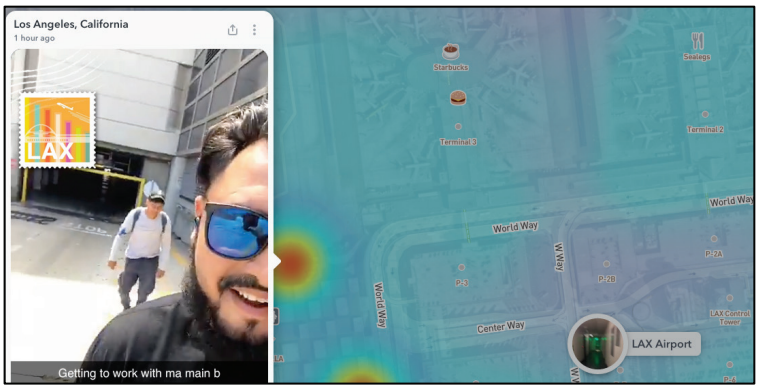


Figure 13.06: A Snap Map result from an airport.

Third-Party Snapchat Search

Once you have exhausted official search options through Snapchat, you should consider third-party options. These will never include all Snapchat data and will usually focus on the most popular accounts. Any non-official Snapchat service relies on its own successful scraping of Snapchat, and no third-party options possess the entire Snapchat database. Therefore, we must always include numerous search options. These services appear and disappear rapidly, but an online search for "Snapchat search users" should identify any replacements.

Ghostdex (ghostdex.app)

Enter a username within the search option or navigate directly to snapdex.com/celebrity/mike and view the bio, a profile picture, location data, and public "snaps". This will only be useful if you know your target's Snapchat name, as any additional search features appear missing. Note that creating a Snapchat username is optional, and many profiles cannot be searched this way.

Google/Bing/Yandex

You should expect by now that we can use traditional search engines to query specific sites and services. A query of site:snapchat.com "inteltechniques" would search for any mention of "inteltechniques" within Snapchat across the three major search engines. Similar to Tumblr, I include Snapchat query options within the Username Tool, which is explained later.

Google Networks

Every Google account has been issued a specific numeric ID which associates the account with any Google service used. This includes Gmail addresses, YouTube accounts, photo albums, and other areas. My goal is always to identify the Google ID and then research any areas which may be associated with my target. First, we must find the ID itself. There are several ways, let's start with the manual option.

- Log in to any covert Google account and navigate to mail.google.com/chat.
- Right-click on the page and select "Inspect".
- Click "Find a Chat" and enter your target's email address.
- Strike enter but do not send any communication.
- In the "SearchHTML" field, enter the email address of your target.

Striking enter or return on your keyboard should cycle through the results. One of the results should contain data such as "data-member-id="user/human/100202552162672367520". The numbers after "human/" is the Google User ID of your target. I typically document this within my report for future reference. You can also use the automated service at **Epieos** (tools.epieos.com/email.php) to obtain this information. The following URLs would display any public photo albums and map contributions made by this target.

https://get.google.com/albumarchive/100202552162672367520
https://www.google.com/maps/contrib/100202552162672367520

Let's conduct a real example using my friend Justin Seitz as the target. We know his personal email address is justin.seitz@gmail.com (he is the creator of Hunchly and provided permission to do this). This means that he has a valid Google account, so he should also have a Google ID. I used the Epieos tool to search his email. The result appears in Figure 13.07 (left). This identifies his Google ID as "100202552162672367520". The following URL displays his Google map contributions. Clicking "Reviews" presents the content in Figure 13.07 (right).

https://www.google.com/maps/contrib/100202552162672367520

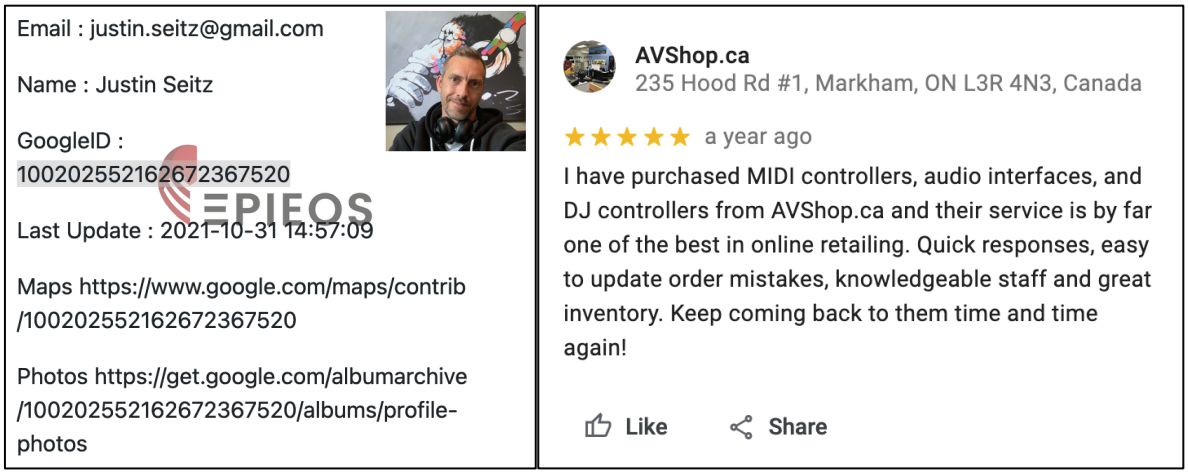


Figure 13.07: Google ID number (left) associated with a map review (right).

We now know his interests and potential area of residence. We can also use this technique to confirm association from our target to malicious negative reviews. My success rate of viewing public photo albums with this method is low, but the availability of mapping data has been high. This will immediately display reviews of businesses and photos of locations which have been shared by your target with the Google Maps service.

Tumblr (tumblr.com)

Tumblr was purchased by Yahoo in 2013, Verizon became the owner when it acquired Yahoo in 2017, but then sold Tumblr to WordPress owner Automatic in 2019. While I believe neither Yahoo or Verizon took advantage of the reach of this network, I suspect we will see Tumblr continue to thrive over the years under control of Automatic. Tumblr is half social network and half blog service. At the time of this writing, there were hundreds of millions of blogs and hundreds of billions of posts. These posts can include text, photos, videos, and links to other networks. The search method and layout are no longer user-friendly. The search feature will only identify blogs that were specifically tagged by the creator with the provided search terms. I suggest using a custom Google search. As an example, I conducted a search of "osint" within the official Tumblr search and received three results. I then conducted the following search in Google and received the appropriate 611 results.

site:tumblr.com "osint"

Once you find a blog associated with your target, consider the Photo Gallery Tool explained in Chapter Four. It will retrieve all of the images and videos. Opening each post within a browser allows you to see and capture all "likes" and "reblogs". Similar to other networks such as Instagram, people can "tag" items with specific terms. The following URL presents any posts tagged "osint". At the time of this writing, the tag URL presented many more items than a keyword search on Tumblr.

https://www.tumblr.com/tagged/osint

Usernames cannot be queried, but a direct URL may locate the content. If your target uses "inteltechniques" on other sites, you may find applicable content at the following URL.

https://inteltechniques.tumblr.com

Most profiles contain a search field dedicated to the author's posts. It can also be queried with a direct URL. If you wanted to search for any posts by "inteltechniques" which contain the term "osint", the following URL would apply.

https://inteltechniques.tumblr.com/search/osint

Many Tumblr blogs display a large layout which hides many of the posts. Therefore, I always prefer to browse a user's posts with the "Archive" display.

https://inteltechniques.tumblr.com/archive

I have incorporated some of these queries into the Username Tool, which is explained in a later chapter.

Telegram (telegram.org)

Much of the content on Telegram is encrypted private communications between individuals. However, "Channels" were added in 2015 and have become quite popular. These are publicly visible and often include content replicated from other social networks and websites. There is no official search option on telegram.org, but options exist based on the public data. I have found site searches on Google to be the most helpful. The following examples may help explain. I searched "osint" within different official Telegram domains and received very unique results from each.

site:telegram.me "osint" (2620 results)
site:t.me "osint" (22 results)
site:telegra.ph "osint" (255 results)

Limited results will likely open within browser windows if you do not have Telegram installed, and may launch the Telegram desktop application if it is available. **Note that I will dive much deeper into Telegram usage within the data breaches chapter.**

Telegram Analytics (tgstat.ru - telemetr.io)

I believe these are the most robust Telegram Channel search options currently available. A keyword search presents channels which are associated with the terms and immediately displays analytics including subscribers, growth, and post reach. Clicking on the channel name presents further details about post behaviors and history.

Access to Private Profiles

There have been several "hacks" in the past that would allow a type of "back door entry" into a profile that is marked as private. By the time these methods become known, the vulnerability is usually corrected by the social network host. Websites or applications that publicly claim to be able to access this secured data are most often scams or attempts to steal your own passwords. In my experience, it is best to avoid these traps and focus on finding all available public information. At the time of this writing, an application had recently surfaced that claimed to be able to obtain all information from within private Facebook accounts. The program did not work, but installed a malicious virus instead. If it seems too good to be true, it probably is.

Contact Exploitation

Previously, I explained how to add cell phone numbers and email addresses as contacts to an Android virtual machine in order to supply them to various apps. When programs received the numbers directly from the contacts list, it believed the contacts were "friends"; therefore, they often identify the names and accounts associated with each number or email. I refer to this technique as contact exploitation, and the Android technique is not the only option for this type of activity. This technique works throughout several social networking environments. I keep covert Gmail accounts solely for adding my target's contact information and asking networks to find friends based on this data. I am often presented with profiles in my target's true name as well as alias accounts.

We now know that locating someone's social network profile can reveal quite a lot about them. Just knowing a target name can make use of the people search engines that will identify places to seek more information. Unfortunately, sometimes the investigator does not know the target's name. A common scenario is an investigation about some type of event. This could be a specific violent crime, a bomb threat on a campus, or inappropriate chatter about a particular business. All of these scenarios require search engines that monitor social network traffic. There is an abundance of these types of services. Some will work better on reactive investigations after an incident while others show their strength during proactive investigations while monitoring conversations. Many of the sites mentioned here will find the same results as each other.

Overall, some of the strongest methods of searching social network traffic have already been discussed in the Facebook and Twitter chapters. Searching for traffic at the source, such as on twitter.com, will usually provide more accurate and updated content than on an aggregated website of multiple sources. Furthermore, searching specific services through Google or Bing may sometimes quickly locate results that would be difficult to obtain anywhere else. The use of the site operator previously explained will take you far. Aside from direct searches on social networks and targeted search engine queries, there are other options. The accuracy of the services mentioned in the rest of this chapter varies monthly. Hopefully, you will find some of these websites to have value in your investigations. I believe that the options in this chapter should be used to supplement, not replace, the results obtained from previous methods.

Social Searcher (social-searcher.com)

I had previously discouraged users from attempting searches on the first version of Social Searcher. Since then, I have begun to rely on their free service to digest data located on the main social networks. You can provide any keywords, usernames, or terms and receive the most recent results from Facebook, Twitter, Instagram, and the overall web. It allows email alerts to be created for notification of new content matching your query. One of the unique features of this website is the free ability to export search results into CSV format. This output contains the username, date & time, and entire message among other information. Having this in a spreadsheet format can be incredibly beneficial. This document also included dozens of Reddit and other network posts. The document could be imported into any other collection system.

International Social Networks

While this book is heavily focused on social networks popular in the United States, they tend to be fairly global with an international presence. This is especially true for Facebook and Twitter. However, there are many social networks which are not popular within the United States that are the primary networks to local residents abroad. This section attempts to identify and explain the most popular foreign networks that may be used by your international targets.

Russia: VK (vk.com)

VK is basically a Russian version of Facebook. You can create a new free account or log in using your existing Facebook credentials. Most search options function without logging in to an account. The page at vk.com/people offers advanced search options which allow filtering by location, school, age, gender, and interests. Most profiles publicly display a user's birthday, location, and full photo collection.

Russia: Odnoklassniki (ok.ru)

Odnoklassniki works similar to most other social media platforms. It is intended to be a way to communicate with friends, as well as an opportunity to network with other people with similar interests. The service is concentrated on classmates and old friends, and translates to "Classmates" in Russian. The official search page is located at ok.ru/search, but you will need to create an account to take full advantage of the options. I have found a targeted site search on Google to be most effective. Searching for Michael Smith could be conducted with "site:ok.ru michael smith". The links connect directly to profiles, which can be browsed as normal. These will appear very similar to Facebook profiles. The upper-right portion of a profile will announce the date of the user's last login. Most of the profile details are public, and do not require any type of URL trickery in order to expose the details.

China: Qzone (qq.com)

Qzone is typically used as a blogging and diary platform. Most of the loyalty to the platform is due to the popularity of the instant messaging tool "QQ". This is a one-to-one messaging platform, so opportunities for public search are not present. The search options on qq.com pages provide results similar to Chinese search engines such as Baidu. The searches are not restricted to the social network profiles. I have found searching on Google or Baidu to work best for English queries. The format of "site:user.qzone.qq.com michael smith" should begin your query.

China: Renren (renren.com)

Literally translated as "Everyone's Website", Renren is a Chinese remake of Facebook. It is one of the most popular Chinese social networks. Users earn points for activities such as logging in, posting status messages, commenting, and receiving comments. As users earn points, their level on the site increases, which unlocks special emoticons, profile skins, and the ability to go "invisible" and view other users' profiles without their

knowledge. The home page does not allow profile search, but browse.renren.com does. Clicking on any profiles from this query will prompt you to sign in to an account. However, a targeted site search should eliminate this. The following Google search identified several various pages that contained Michael Smith.

site:renren.com "michael smith"

If the results are too overwhelming, you can use the following structure to filter the content.

site:blog.renren.com "michael smith" (Filter for blog results only)
site:page.renren.com "michael smith" (Filter for profile results only)
site:zhan.renren.com "michael smith" (Filter for news results only)

Latin America: Taringa (taringa.net)

Taringa has a presence in every country in the Spanish-speaking world. Its main markets are Argentina, Spain, Colombia, Chile, Peru, and the U.S. Latino community. The search functionality is fairly straightforward, but the following URLs may produce more efficient results.

taringa.net/buscar/posts/?q=OSINT (Searches OSINT within post comments)
taringa.net/buscar/comunidades/?q=OSINT (Searches OSINT within community posts)
taringa.net/buscar/shouts/?q=OSINT (Searches OSINT within "Shout" posts)
taringa.net/buscar/imagenes/?q=OSINT (Searches OSINT within images)

Newer Social Networks

These social networks represent only a portion of the available options. Overall, resort to custom Google searches when new services appear. As I write this, I am monitoring newer sites such as **Parler** (parler.com), **Gab** (gab.com), **Gettr** (gettr.com), and **Mastodon** (mastodon.social), all of which claim to be the next generation of social networks. The search techniques previously presented throughout this book also apply to these new platforms. Let's briefly dissect each.

Parler is indexed by Google, so the following queries should identify data of interest.

site:parler.com "keyword"
site:parler.com/post/ "keyword"
site:parler.com/profile/ "keyword"

Once you identify a target profile, such as "inteltechniques", or a topic, such as "bomb", the following should disclose details.

https://parler.com/inteltechniques
https://parler.com/search?type=user&s=inteltechniques
https://parler.com/search?type=hashtag&s=bomb

Clicking through options such as "Media", "Following", and comment notifications should present data similar to Twitter and Instagram. There is no real magic here, what you see is what you get.

Gab has been widely described as a haven for extremists including neo-Nazis and white supremacists, and has attracted users who have been banned from other social networks. Gab is not indexed well by Google, but we have the following direct query URLs assuming your target is "inteltechniques" and your keyword of interest is "bomb". Note that an account is required to view most data.

https://gab.com/inteltechniques
https://gab.com/inteltechniques/followers
https://gab.com/search/bomb/top
https://gab.com/search/bomb/latest
https://gab.com/search/bomb/users
https://gab.com/search/bomb/groups
https://gab.com/search/bomb/topics
https://gab.com/hash/bomb
https://gab.com/hash/bomb/date
https://gab.com/hash/bomb/score
https://gab.com/groups/bomb

Gettr is a Twitter clone targeted toward far-right political beliefs. The structure of all accounts and feeds follows the Twitter examples previously presented. The following would apply if I were a member.

https://www.gettr.com/user/inteltechniques
https://www.gettr.com/user/inteltechniques/followers
https://www.gettr.com/user/inteltechniques/following
https://www.gettr.com/search?q=to:inteltechniques
https://www.gettr.com/search?q=from:inteltechniques
https://www.gettr.com/user/inteltechniques/comments
https://www.gettr.com/user/inteltechniques/medias
https://www.gettr.com/user/inteltechniques/likes

Mastodon received a lot of attention in late 2022. As Elon Musk began taking over Twitter, many people exited and sought alternative options. Mastodon is a very popular Twitter clone, so many people ended up there. While search engines will push you toward joinmastodon.org, that domain will not offer many search possibilities. You might find mastodon.social, but it is only one of many communities which allow connection through the Mastodon network. This is because Mastodon is a ***decentralized*** social network which can be self-hosted and is available on numerous servers. Services such as Twitter are ***centralized***. You must connect through twitter.com and Twitter has the power to prevent your access. You can join and participate in Mastodon from a number of servers, each independent of any Mastodon domains. This can cause a lot of confusion. Let's start with mastodon.social, but we will rely on official APIs for our OSINT research. Assuming your target was "alexwinter" and he was present on the "mastodon.social" instance, the following would apply.

https://mastodon.social/@alexwinter
https://mastodon.social/@alexwinter/with_replies
https://mastodon.social/@alexwinter/following
https://mastodon.social/@alexwinter/followers
https://mastodon.social/@alexwinter/media

These URLs would display his profile and all messages; messages with replies; the people following him; the people he follows; and posts which contain media. This should seem very familiar after the Twitter chapter. While his username on "mastodon.social" is simply "alexwinter", his full Mastodon username, including the community where he joined, is "@alexwinter@mastodon.social". This means that someone could join Mastodon through another server with his exact username. The following are examples of multiple accounts with a username of "JohnDoe".

https://social.tchncs.de/@JohnDoe
https://mastodon.iriseden.eu/@JohnDoe
https://mastodon.top/@JohnDoe

This is why you should never rely on one site, such as mastodon.social. While it may connect you to everything on the Mastodon network, your results could be filtered without you knowing. Next, let's compare a typical graphical view of a profile with the API option. I will return to https://mastodon.social/@alexwinter. I can see the date he joined, an image, and general biographical details. Let's compare that to the following URL.

https://mastodon.social/api/v2/search?q=alexwinter&limit=40&type=accounts

This URL reveals the following text-only result.

id	"109292970397906870"
username	"alexwinter"
acct	"alexwinter@mastodon.social"
display_name	"Alex Winter"
locked	false
created_at	"2022-04-25T00:00:00.000Z"
note	"Boring, suburban dad, mostly making things at Trouper Productions.
url	"https://mastodon.social/@alexwinter"
followers_count	6229
following_count	468
statuses_count	138
last_status_at	"2022-11-12"
verified_at	null

I believe this provides more explicit details, and can be easily copied into a report. The "limit=40" is the maximum number of results which can be returned. Next, let's repeat this with a different server, specifying something different than the server which the user joined.

https://infosec.exchange/api/v2/search?q=alexwinter&limit=40&type=accounts

The results are identical since each server is relying on the overall Mastodon API for this data. We can also use the API to query hashtags. This is where things get interesting. When I searched OSINT within mastodon.social, I received only five results. However, the following API revealed 30 results.

https://mastodon.social/api/v2/search?q=osint&limit=40&type=hashtags

These are only the hashtags which include OSINT on mastodon.social. Repeating the request with the following URL presented 15 additional tags, all on that server.

https://infosec.exchange/api/v2/search?q=osint&limit=40&type=hashtags

Similarly, the following two URLs display very different results.

https://mastodon.social/tags/osint4good
https://infosec.exchange/tags/osint4good

What a mess. I am told that future releases of the server software will correct this issue and display all hashtags across all servers. We will see. Because of these issues, I do not see Mastodon ever replacing Twitter, and it may be nothing more than a niche network with little activity. Or, by the time you read this, it could be the biggest social network in the world and I will have to eat my words. These new search options are included within the Communities Tool, which is discussed within the next chapter. There is no way to predict which social networks will be the next big thing. However, possessing a sharp set of Google and OSINT skills will take you far. If desired, you could add these options to the custom search tools explained in Section One. Until I see wide-spread usage of these platforms, I will wait.

CHAPTER FOURTEEN
ONLINE COMMUNITIES

Online communities are very similar to social networks. The thin line which has separated the two is slowly disappearing. While social networks cater to a broad audience with many interests, these communities usually relate to a specific service or lifestyle. Some online communities do not get indexed by search engines; therefore, the presence of a target's participation will not always be found through Google or Bing. Any time that a target's interests or hobbies are located through the previous search techniques, you should also seek the online communities that cater to that topic. This can often identify information that is very personal and private to the target. Many people post to these communities without any regard to privacy. Some communities require registration to see the content, which is usually free. Occasionally, a cached version of the pages on the site is available without registering. This chapter will provide methods of infiltrating these communities to maximize the intelligence obtained.

Reddit (reddit.com)

Contrary to some previous editions, I now start this chapter with Reddit. This social news aggregation, web content rating, and discussion website went from a place for those "in-the-know" to a resource often cited on mainstream media. More users than ever post, reply to, and read the user-submitted content in the form of either a link or text, each submitted to a specific category known as a Subreddit. Other users then vote the submission up or down, which is used to rank the post and determine its position on the website's pages. The submissions are then discussed on the "comments" page of every entry. The Subreddits cover practically any topic you can imagine. If your target has the slightest interest in the internet, he or she has probably been to Reddit. As of 2020, there were over 1,200,000 Subreddits and over 350 million registered users.

Reddit Search

The official search option on Reddit has been plagued with problems since inception. The search field in the upper right of every page will allow you to query any terms you desire, but the results have not always been optimal. In 2016, I saw the search function improve drastically, and even with some added new features. When typing terms into the search field on any Reddit page, the results will be from all pages, including thousands of Subreddits. While you should understand this option, and even execute target searches from the home page on occasion, we should consider some advanced search options. We can replicate that standard keyword search within a URL. This is beneficial for bookmarking searches of interest that need checked often. The format for a search about OSINT is as follows.

https://www.reddit.com/search?q=OSINT

The results from such a generic search can be quite overwhelming. With the following URL, we can force Reddit to only deliver results if our search term is within the title of a post, and not simply present within comments.

https://www.reddit.com/search?q=title:OSINT

If you know the name of the Subreddit, you can navigate directly with the following structure.

https://www.reddit.com/r/OSINT/

If you locate a username of interest while searching Reddit, you can load all of that user's posts and comments by clicking on the name. Alternatively, the following URL can be used.

https://www.reddit.com/user/inteltechniques

If you want to see a summary of all original posts, the following URL can be used.

https://www.reddit.com/user/inteltechniques/posts/

If you want to see a summary of all original comments, the following URL can be used.

https://www.reddit.com/user/inteltechniques/comments/

If you have a target website, and you want to know if the URL has ever been posted as a submission link, the following URL will display all results.

https://www.reddit.com/search?q=site:inteltechniques.com

If Reddit is not providing the results you think you should be receiving, you should return to our previous instruction on Google searching. The following query would identify any posts, categories, or users that included the word "surveillance".

site:reddit.com "surveillance"

If you wanted to force Google to restrict its searching to a specific Subreddit, such as OSINT, you would add "/r/osint" after the first portion. If you wanted to restrict the searching to a specific user, you would add "/user/inteltechniques" to the end. The following are two examples.

site:reddit.com/r/osint "surveillance"
site:reddit.com/user/inteltechniques "surveillance"

Each of these queries could be replicated within both Bing and Yandex. While Google is good at indexing new content within Reddit, I have witnessed better performance from Yandex when the content was associated with pornography and other adult content.

"Old" View

Since 2018, Reddit has been changing the layout and function of their website. This also includes many additional trackers which collect information about your usage. I prefer the "old" view of Reddit, which can be achieved by replacing "www" with "old" within any Reddit link. This is personal preference, and I believe the old view provides a more compressed experience which displays more content within each page.

Deleted Content

If you have identified any Reddit content of interest, you should consider checking any online third-party archives. These historic representations of an account will often disclose previously deleted or modified content. It is extremely common for Reddit users to edit or delete a comment entirely, especially if it was controversial. I have investigated numerous Reddit accounts where the evidence I expected to find was not present. First, I always search the standard archive options which were previously explained. The following direct URLs would attempt to display historic versions of a Reddit user's profile. You could replace the Reddit user URL within each of these with a Subreddit address or Reddit post URL.

webcache.googleusercontent.com/search?q=cache:https://www.reddit.com/user/mikeb
web.archive.org/web/*/https://www.reddit.com/user/mikeb

You may get lucky with these queries, but the results are often limited. These will display the historic view of a Reddit account at a specific moment in time. While this may provide evidence for your investigation, you should also identify any further deleted content. In order to dig much deeper, we will rely on Pushshift.

Pushshift (files.pushshift.io)

This archive contains most of the content which was publicly posted to Reddit since 2005. This provides an amazing collection of data, including most deleted posts. The next time your target wipes out various Reddit posts before you can collect the evidence, Pushshift may reveal the deleted content. This site allows you to download the entire archive, but that may be overkill for most users. Instead, we can take advantage of their robust application programming interface (API). First, let's assume that you are only interested in a specific user that has deleted all content. Figure 14.01 displays a partial post which I created on Friday, July 19, 2019. On September 1, 2019, I deleted all of my posts from Reddit, including that initial content. Figure 14.02 displays the confirmation my account was empty. As I wrote this in 2020, the following URL queried the data set for posts that have been archived by Pushshift for user inteltechniques. In 2022, I requested removal from Pushshift, so replace inteltechniques with the deleted account palmdraws if you would like a live demo.

https://api.pushshift.io/reddit/search/comment/?author=inteltechniques

This URL presented a lot of information about my deleted posts, but I was most interested in the data visible in Figure 14.03. It displayed the entire deleted comment and the subreddit location. Furthermore, it provided a Unix time of 1563580052, which converts to Friday, July 19, 2019 5:47:32 PM GMT.

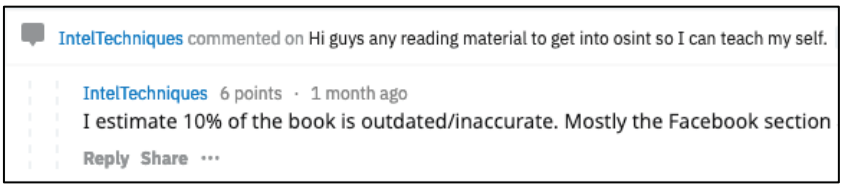


Figure 14.01: A Reddit post prior to deletion.

Figure 14.02: A Reddit post after deletion.

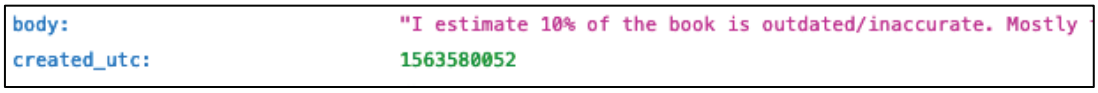


Figure 14.03: A Pushshift result of a deleted post.

This URL will display the most recent 25 posts, regardless of whether they are still on Reddit or have been removed. This is a great start, but our target may have thousands of posts. The following URL adds two options at the end to force sorting in ascending format and display 1000 comments within a single page.

https://api.pushshift.io/reddit/search/comment/?author=inteltechniques&sort=asc&size=1000

If you are seeking a specific post with unique wording, you can accomplish this with the following URL. This example would identify public and deleted posts mentioning my username.

https://api.pushshift.io/reddit/search/comment/?q=inteltechniques

Each of these searches may present too much content and may not be easy to digest. We can filter unwanted content in order to produce less results. The following would repeat our previous search, but only display content from the Subreddit Privacy.

https://api.pushshift.io/reddit/search/comment/?q=inteltechniques&subreddit=privacy

If you wanted to limit results to a single user with a timeframe between 5 days prior to your search and 30 days prior to your search, you would navigate directly to the following URL.

https://api.pushshift.io/reddit/search/comment/?author=inteltechniques&after=30d&before=5d

In order to demonstrate the value of this, consider the following random example. On November 8, 2020, I searched on Twitter for "reddit account" "deleted". The first result was the Tweet seen in Figure 14.04. Reddit user Defaultyboi6829 posted on Twitter "Good thing I deleted my Reddit account". On November 10, 2020, I navigated to the following URL, which displayed 25 of the most recently deleted comments. His latest post was a disagreement about Minecraft with another user. Evidence of this interaction is not present anywhere on the live view of Reddit.

https://api.pushshift.io/reddit/search/comment/?author=Defaultyboi6829&sort=asc

Note that all of these searches only identify results which are comments and not user submissions. A submission is a new topic, and a comment is a post within a specific submission. In order to replicate all of these queries for user submissions, simply replace "comment" in each example to "submission". The following URL only displays my deleted submissions.

https://api.pushshift.io/reddit/search/submission/?author=inteltechniques

In a moment, I will demonstrate my offline search tools which you can use to simplify this entire process. You can also experiment with an online tool called **RedditSeacrh** (redditsearch.io). The ability to extract deleted content from a community as large as Reddit is a substantial OSINT technique. I encourage you to monitor Reddit for new features and changes in order to update your own tools as needed.

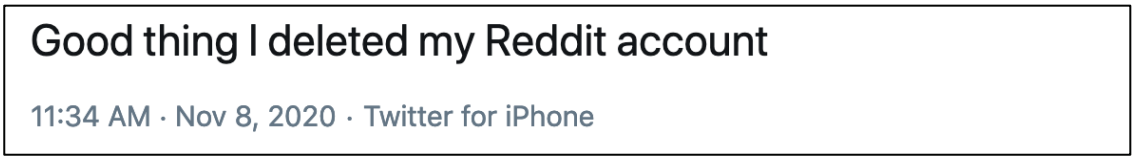


Figure 14.04: A Twitter post announcing deletion of a Reddit account.

Images

Reddit is well-known for hosting entertaining images and memes. The majority of linked images on Reddit are now self-hosted, but I still see a lot of images hosted on a photo-sharing site called **Imgur** (imgur.com). This can be very beneficial when you are investigating an image post that has been removed from Reddit. If a user posted a photo to Imgur, then linked it to a Reddit post, and then deleted the post, the image is still online. You will no longer have a link to the image, and randomly searching Imgur will be unproductive. Instead, we can browse all of the Reddit images on Imgur with a direct URL. The following address will display the current images, in reverse chronological order, associated with the Subreddit titled NetSec. Scrolling will continuously load older images.

https://imgur.com/r/netsec

If you find an image of interest, you should consider searching the name of the image within Pushshift. Let's run through an example. Assume that you suspected your target was posting images of his antique vehicle on the Subreddit /r/projectcar, but he deleted the posts before you could find them. You should first navigate to the following page of related images on Imgur.

https://imgur.com/r/projectcar

Assume you then located a potential suspected vehicle image at the following address.

https://imgur.com/J0C7Mi9

You should right-click on the image and select "View Image" to open view the full-size version in a new tab.

https://i.imgur.com/J0C7Mi9.jpg

The filename of the image is J0C7Mi9. The following two URLs search this image within both submissions and comments within all of Reddit, including deleted posts.

https://api.pushshift.io/reddit/search/submission/?q=J0C7Mi9
https://api.pushshift.io/reddit/search/comment/?q=J0C7Mi9

The second URL provides the following data within the result.

Author: mrmoto1998
body: [Obligatory pic of the moldsmobile](https://imgur.com/J0C7Mi9.jpg)
created_utc: 1568129504
link: /r/projectcar/comments/d296fl/tore_up_some_carpet_in_the_moldsmobile/eztg3jk/

You now know the author, date, and original link of the post on Reddit. The previous example may be an extreme case scenario, but the possibilities are endless. The important message is to search any keyword data through Pushshift when your investigation is associated with Reddit. Next, you should consider a reverse image search. This will be explained in detail later, but you should know now that you have a Reddit-specific reverse image search option called Karma Decay. Assume that you located an image on Imgur at the following URL.

https://imgur.com/r/funny/0DnE1aB

You can navigate to karmadecay.com, supply this address, and immediately see if that image has been posted to any other locations within Reddit. If you wanted to bookmark a direct URL for future checking, you could use the following to obtain the same result.

http://karmadecay.com/imgur.com/r/funny/0DnE1aB

Note that Karma Decay blocks pornographic images. If your investigation involves any adult photos, you will need to use a different service called NSFW Reddit Reverse Image Search (i.rarchives.com). Enter any image here and it will search for other copies within Reddit. You can also submit directly via URL as follows.

http://i.rarchives.com/?url=http://i.imgur.com/mhvSa.jpg

Investigation Subreddits

There are many Subreddits that can provide a unique benefit to an investigator, three of which are outlined here. There are several versions of each of these, but those that I present here have the most history of being helpful. You will find additional options with a bit of searching.

Reddit Bureau of Investigation (reddit.com/r/rbi)

This active community helps other Reddit users solve crimes and other problems. Internet gurus will help find deadbeat parents; computer specialists will aid in tracking stolen devices; and private investigators will assist with investigation techniques. I have used this option several times during my career. The most successful cases involved hit and run traffic crashes. In 2013, I assisted a northern Illinois police department with the

investigation of a fatal car crash. The offender fled the area and an elderly woman died. Three small pieces of the offending vehicle were left at the scene. After posting this information to local media outlets, I submitted it to RBI. Within minutes, several vehicle body shop employees were tracking down the parts and eventually tied them to a specific year and model of a 10-year-old vehicle. This information led to the arrest of the subject. Another victim of an unrelated hit and run traffic crash posted a blurry photo of the suspect vehicle and asked for assistance. Within hours, a Reddit user identified the license plate through digital correction techniques.

Pic Requests (reddit.com/r/picrequests)

A constant frustration in my work is blurry, out of focus, or grainy digital images. Commonly, I will receive surveillance photos that are too dark or light to identify anything of value in the image. Occasionally, I will find images on social networks that could be beneficial if they were just a touch clearer. Pic Requests saves the day. This Subreddit consists of digital photo experts that can perform Photoshop magic on practically any image. Many Reddit users will request old photos colorized, torn photos digitally repaired, or unwanted subjects removed from an image. I have uploaded several surveillance images to this group with a request for assistance. The users have been incredibly helpful by identifying digits in blurred license plates and turning dark surveillance footage into useful evidence.

What Is This Thing? (reddit.com/r/whatisthisthing)

I am consistently amazed at the results from this Subreddit. What Is This Thing is a place where you can post a digital photo of practically anything, and someone will know exactly what it is while providing detailed and cited additional information. Many users post images of old antiques and intricate items hoping to identify something valuable in their collection. I use it to identify tattoo meanings, graffiti, suspicious items mailed to politicians, vehicle parts, and just about anything else that is presented to me during my investigations.

Real World Application: In 2012, I was asked to assist with a death investigation of a "Jane Doe". I submitted a sanitized version of a tattoo on her back that appeared to be Chinese symbols. Within five minutes, a Reddit user identified the symbols, their meaning, and references to the region of China that would probably be related to my investigation. A reverse image search of his examples led to information about a human trafficking ring with which the victim was associated. This all occurred over a period of one hour.

If you plan to use these techniques on Reddit, please consider a few things. You should create a free account now and hold on to it. Creating a new account and asking for help minutes later can be viewed as rude. I like to use accounts that appear to have been established a long time ago. If you are visible as an active member of Reddit with a history of comments, this might encourage other active members to assist you. You should never be demanding in your requests. Remember, these people are volunteering to help you. Many of them possess a skill set that cannot be found elsewhere. I also never upload any content that is not already publicly available. If digital images were released to the press, I have no problem releasing them to Reddit. If my target image is already on a public social network, I see no reason it cannot be linked through Reddit.

While Reddit seems to get most of the attention in this type of community, there are alternative options that are rapidly growing. These include 4chan, Hacker News, and others. I will briefly discuss the most common search options, which can be replicated with my Custom Communities Tool that is explained later.

4chan (4chan.org)

4chan is a mess. It is an image-board website and users generally post anonymously, with the most recent posts appearing above the rest. 4chan is split into various boards with their own specific content and guidelines, modeled from Japanese image-boards. The site has been linked to internet subcultures and activism groups, most notably Anonymous. The site's "Random" board, also known as "/b/", was the first board to be created, and is the one that receives the most traffic. This site is full of bullying, pornography, threats, and general illicit behavior. It has also been the focus of numerous investigations. There is no search feature. In this scenario, we

will use 4chansearch.com. The following examples are direct URLs that take advantage of this third-party search option, each using "OSINT" as a search term.

Active Search: http://4chansearch.com/?q=OSINT&s=4
Archives Search: http://4chansearch.com/?q=OSINT&s=7
Archives Alternative: https://archive.4plebs.org/_/search/text/OSINT/order/asc/
Google Search: https://www.google.com/search?q=site:4chan.org%20OSINT

Hacker News (news.ycombinator.com)

While this site is targeted toward a tech-savvy community, general discussion topics are followed by millions of users daily. Fortunately, we have a lot of control with searching specific posts, keywords, users, and favorites. The following searches locate data based on a keyword (OSINT) and user (inteltechniques).

Text Search: https://hn.algolia.com/?query=OSINT&type=all
Username Search: https://news.ycombinator.com/user?id=inteltechniques
User Posts: https://news.ycombinator.com/submitted?id=inteltechniques
User Comments: https://news.ycombinator.com/threads?id=inteltechniques
User Favorites: https://news.ycombinator.com/favorites?id=inteltechniques
Google Search: https://www.google.com/search?q=site:news.ycombinator.com+OSINT

TikTok (tiktok.com)

TikTok is a Chinese video-sharing social networking service which has become viral globally. It is used to create short music, lip-sync, dance, comedy, and talent videos of 3 to 15 seconds, and short looping videos of 3 to 60 seconds. Posts accept and display comments, both of which require login for access. There is a limited native search within the website, but we can formulate URLs to gain direct access to targeted details. The following URL structure displays the user profile for "inteltechniques".

https://www.tiktok.com/@inteltechniques

If you suspect your target possesses a username which includes "osint", but do not know the exact profile name, the following URL should assist. It displays every username which includes "osint" anywhere in the name.

https://www.tiktok.com/search/user?q=osint

Many users "tag" their posts with a keyword. The following displays posts which include the tag "osint".

https://www.tiktok.com/tag/osint

The search field within TikTok pages can be unreliable. Instead, we can formulate a URL with search terms. The following presents posts which include "osint" within the original post.

https://www.tiktok.com/search?q=osint

If you want to isolate this search to only display videos which include "osint", the following URL would apply.

https://www.tiktok.com/search/video?q=osint

Google indexes individual TikTok posts, so the following search query would present results which include the term "osint" within any post or author profile. This can be beneficial to discover deleted posts.

site:tiktok.com osint

Some TikTok users will broadcast live. Any current live streams can be found with the following URL.

https://www.tiktok.com/live

Once you find your target, navigating through the posts is similar to the techniques previously mentioned with Instagram. However, we do have a few unique offerings. Each profile possesses three identifiers within the source code view of the page. In the following example, I searched the following text within the source code view of the TikTok page of @willsmith, with the result in parentheses.

"authorId" (6727327145951183878)
"uniqueId" (willsmith)
"nickname" (Will Smith)

Note that nicknames are not unique. They are vanity names which can be reused. We can now search the user number within search engines in order to identify additional posts which may no longer be present within the TikTok website. Once you identify an individual post, it may appear as follows.

https://www.tiktok.com/@willsmith/video/7032216839878905134

This will usually include a video which can be played in full screen by clicking on it. Once in the player view, you can right-click the video and save it natively as an MP4 file. Most videos display an upload date within the content, but I always prefer to obtain a full date and time of upload. To do this, right-click a post and choose to view the page source. When this new tab of text opens, search (ctrl-f or cmd-f) for "~tplv". This should present a result similar to the following.

1637781925~tplv

The numbers directly before this search term represent a Unix timestamp, which was explained in previous chapters (Instagram). We can convert this number at unixtimestamp.com, which produces the following result.

Wed Nov 24 2021 19:25:25 GMT+0000

We now know the exact date and time of the post. Everything presented about TikTok up to this point should not require an account. However, accessing comments within a post requires you to be logged in. I always recommend a "burner" account which is associated with a Gmail address or social network connection. When you attempt to create an account, you will see a list of accepted associations. Once you are logged in, you should see any comments included with the target post. Each post presents a username with hyperlink; the comment from that user; the date of the comment; the number of "likes"; and an option to expand further comments to that comment. Screen captures may work well for unpopular posts, but posts with many views may be a problem. **Export Comments** (exportcomments.com) will extract the first 100 comments from any post. A premium account is required to download larger content. There are several "scraper" applications and Chrome extensions which claim to aid in comment exportation, but I have found them all to be unreliable.

There are numerous TikTok services which provide analytics on target profiles. I prefer those which allow submission within the URL, such as the following, which display results for "vancityreynolds" (Ryan Reynolds, who is an owner of Mint Mobile, which may be of interest to listeners of my show).

https://tokcount.com/tiktok-analytics/vancityreynolds
https://tokcounter.com/tiktok-analytics/vancityreynolds
https://exolyt.com/user/vancityreynolds/full
https://analisa.io/profile-tiktok/vancityreynolds

These four sites confirm the following details about my target.

6.8M Followers
3 Following
18M Likes
43.7k Average Post Comments
39.4M Average Plays
108k Average Shares
13% Engagement Rate

Latest Bio Details
Profile Image
All Recent Posts
Daily Engagement Rate Changes
Follower Growth Chart
Daily Performance Score

While most of these details will not be helpful to an investigation, they should all be documented. I have included most of the TikTok search features within the Communities Search Tool presented in a moment. We currently see TikTok as part of every investigation which is associated with a target under the age of 20. It is important to understand this platform and the many ways of exploring the content.

Nextdoor (nextdoor.com)

This popular online community allows for people within a specific neighborhood or geographical area to communicate privately within a controlled space. People within a neighborhood in Texas cannot see posts from a neighborhood in Illinois. In order to join a specific neighborhood, one must either receive an invite from another neighbor or request a physical invite be sent via postal mail to an address within range. This presents problems for OSINT. If you are investigating posts within your county, there are a few things you can do to extend your range.

While logged in to a Nextdoor account, navigate to "Settings" then "Neighborhoods". By default, you can only see activity within your specific neighborhood. However, clicking the "Explore Neighborhoods" button should present surrounding areas which you can join. In my experience, you can usually see the majority of your county by joining all available groups.

Meetup (meetup.com)

Meetup consists of users and groups, with all communication related to events where people actually meet in real life. Each user creates a profile that includes the person's interests, photos, and username. A group is created by one or more users and is focused on a general interest in a specific location. An example would be the "Houston Dog Park Lovers", which is a Houston-based group of dog owners that meet at dog parks. Each group will post events that the members can attend. The majority of the events posted on Meetup are visible to the public and can be attended by anyone. Some groups choose to mark the details of the event as private and you must be a member to see the location. Membership is free and personal information is not required.

You can search Meetup by interest or location on practically any page. Once you locate a group page, you can browse the members of the group. This group page will also identify any future and past events sponsored by the group. These past events will identify the users that attended the event as well as feedback about the event. This site no longer offers the option to search by username. In order to do this, you will need to use a search engine as described in a moment. A user profile will often include links to social networks and messages from associates on the website. Additionally, these profiles will identify any future Meetup events that the user plans on attending. Because of this, the site has been used in the past by civil process servers, detectives, and the news media to locate people that had been avoiding them. The following Google search structures have been most helpful in my experience.

Name Search (John Morrison): site:meetup.com inurl:member john morrison
Event Search (Protest): site:meetup.com inurl:events Protest
Post Search (Protest): site:meetup.com inurl:discussions Protest
Google Keyword Search (OSINT): site:meetup.com OSINT

Dating Websites

When investigating cheating spouses, background information, personal character, or applicant details, dating sites can lead to interesting evidence. The presence of a dating profile does not mean anything by itself. Millions of people successfully use these services to find mates. When a target's profile is located, it will usually lead to personal information that cannot be found anywhere else. While many people may restrict personal details on social networks such as Facebook, they tend to let down their guard on these intimate dating websites. In my experience, the following will apply to practically every dating website.

- You must have an account to browse profiles, which is usually free.
- You must have a premium (paid) account to contact anyone.
- If a target uses one dating service, he or she likely uses others.

Instead of explaining each of the dating services, I will focus on methodology of searching all of them. While each website is unique and possesses a specific way of searching, they are all very similar. Overall, there are three standard search techniques that I have found useful, and they are each identified below.

Username: Every dating website requires a username to be associated with the profile, and this data is searchable. Surprisingly, most users choose a username that has been used somewhere else. I have seen many dating profiles that hide a person's real name and location, but possess the same username as a Twitter account. The Twitter account then identifies name, location, and friends. Additional username tools are presented later.

Text Search: This is a technique that is often overlooked. Most dating network profiles include an area where the users can describe themselves in their own words. This freeform area often includes misspellings and obvious grammatical errors. These can be searched to identify additional dating networks since many users simply copy and paste their biography from one site to another. In 2013, I was teaching an OSINT course in Canada. During a break, one of the attendees asked for assistance with a sexual assault case that involved the dating website Plenty Of Fish. The unknown suspect would meet women through the online service and assault them. All of the information on his profile was fake, and the photos were of poor quality and unhelpful. Together, we copied and pasted each sentence that he had written in his bio for the profile. Eventually, we found one that was very unique and grammatically worded poorly. A quoted Google search of this sentence provided only one result. It was the real profile of the suspect on Match.com, under his real name, that contained the same sentence describing himself. The high-quality photos on this legitimate page were used to verify that he was the suspect.

Photo Search: In later chapters, I explain how to conduct reverse-image searching across multiple websites. This technique can compare an image you find on a dating network with images across all social networks, identifying any matches. This will often convert an "anonymous" dating profile into a fully-identifiable social network page. This applies to any dating networks, and photos will be your best way of identifying your target.

The list of popular dating websites grows monthly. The following are the current most popular services.

Match (match.com)
Plenty of Fish (pof.com)
eHarmony (eharmony.com)
OK Cupid (okcupid.com)
Christian Mingle (christianmingle.com)
Ashley Madison (ashleymadison.com)
Adult Friend Finder (adultfriendfinder.com)
Farmers Only (farmersonly.com)
Elite Singles (elitesingles.com)
Zoosk (zoosk.com)
Friendfinder-X (friendfinder-x.com)
Badoo (badoo.com)

Tinder (tinder.com)

A section about online dating would not be complete without a reference to Tinder. The simplest explanation of Tinder is that it connects you with people in your immediate geographical area who are also using the service. Some call it a dating app, some refer to it as a "hook-up" app. Either way, it is probably the most popular dating service available today.

While this service was once natively available only through a mobile app, they have recently begun allowing account access via their website. However, I find this to be full of frustration. In order to access Tinder via their web interface, you must provide either Facebook account information or a mobile telephone number. I never advise connecting any third-party service to any covert Facebook account, so that option is out. Instead, you can provide a Google Voice number, which is a bit more acceptable. Supply a covert Google Voice telephone number and prepare for the issues.

First, Tinder will send you a text message with a code to verify your number. Supplying this code to Tinder passes the first hurdle. If you are using a VPN, you will immediately be sent to a series of tests to verify you are human. If you have been annoyed by Google Captcha pop-ups in the past, Tinder's options take frustration to a new level. In most scenarios, you will be asked to complete a series of 20 small tasks. In my experience, completing them perfectly results in a failure report 99% of the time. Tinder simply does not like a combination of a VOIP number and a VPN. Providing a real cellular number seems to pacify Tinder somewhat. Providing a true number and internet connection without a VPN seems to make it happy. However, you sacrifice privacy and security in order to do so. If you accept these risks, you can proceed with the web-based access to Tinder. After I explain the process, I will present my chosen solution.

In order to access Tinder from your web browser, several things must be perfectly aligned in order to prevent account blocking. Tinder gets bombarded with fraudulent accounts, and their radar for investigative use is very sensitive. The following instructions assume that you do not have an existing Tinder account.

- Connect to a public Wi-Fi location, without a VPN, near your target.
- Click the Login button at tinder.com and choose "Log In With Phone Number".
- Supply a Google Voice number.
- Confirm the text message received on Google Voice number.
- Complete registration with alias name and photo.

These instructions may seem simple, and too good to be true. They are. Tinder has begun blocking any type of GPS spoofing, even if done manually through the browser inspector. They focus much more on the networks through which you are connected. The previous edition explained ways to spoof GPS within your browser and pretend to be at another location. In my experience, these tricks simply do not work anymore. If you are able to connect through the web version of Tinder, it is unlikely to be of any use. Furthermore, your "matches" will likely be far away from your actual location. Personally, I no longer try to make this work.

Instead, I keep an old Android phone ready for any Tinder investigations. I have the Tinder app installed along with the "Fake GPS Location" app. I keep the Tinder app logged in using a covert Google Voice number. Before I open Tinder, I set my desired location through the Fake GPS Location application. Upon loading Tinder, I can control the search settings through the app. I usually choose to limit the search to a few miles away from my target's location. In my experience, this will not work on an iPhone due to GPS spoofing restrictions. Since Tinder actively blocks emulators, connecting through VirtualBox or Genymotion does not work. This will simply require a dedicated Android device.

Real World Application: I have two recent experiences with covert Tinder accounts to share. The first is a human trafficking investigation with which I was asked to assist. The investigation unit received information that a well-known pimp had recruited several children to enter his world of prostitution. He was not promoting them on websites, as he believed it was too risky. Instead, he posted profiles to Tinder while located at a run-

down motel. From several states away, I spoofed my GPS on my Android device to the motel of interest. I set my search settings for females within five miles aged 18-19. I immediately observed two images of what appeared to be young girls in a shady motel room. I "swiped right" to indicate interest, and was immediately contacted by the pimp pretending to be one of the girls. We agreed on a price and he disclosed the room number. The local agency working the case began immediate surveillance while a search warrant was obtained. While waiting, they arrested two men before the "date" could begin, and also arrested the pimp after a search warrant was obtained for the room.

The other example I have presents a much different view of Tinder usage. An attorney reached out requesting assistance with a cheating spouse investigation. He was looking for evidence which confirmed the husband was involved with other women. He provided several images of the man and common locations which he was known to frequent. After many failures, I had Tinder launched with the GPS spoofed to the suspect's office. I claimed to be a woman looking for a man his age, and I was eventually presented an image of my target. I swiped right, as did he, and we began a conversation. The evidence piled up immediately.

What photo should you use? First, uploading one photo and vague information to your account looks suspicious. Providing images of other people without their consent is wrong. Using stock photography from the internet will quickly get you banned from Tinder. I rely heavily on **Fiverr** (fiverr.com). I search for people willing to send unique photos of themselves for a few dollars. I once paid a 21-year-old woman $30 for five "selfies" while dressed in different outfits. I received a signed consent form allowing me to upload them to Tinder with the intent of luring cheating spouses. At first, she assumed I was a pervert with a unique fetish. After my explanation, she loved the idea and was eager to assist.

Tinder Profiles

Tinder users can optionally create a username within the network. Instead of being limited to the identity of "Tom, 30, NYC", a user can claim a specific username such as Tom911. This generates a web profile which can be seen publicly (often unknowingly to the user). The following format would display a user with that username. Both should connect to active profiles with multiple photos of each target.

https://www.gotinder.com/@Tom911
https://www.gotinder.com/@MikeB

If you open the source code view of these profiles, we can dig deeper. Searching "birth_date" for user MikeB reveals "1996-11-24" while ""_id" displays "5571c5edb47cba401e0cf68b". We now know the date of birth and Tinder user ID of our target. If he should change his name or username, we still have a unique identifier which could be checked within a new profile.

Discord (discord.com)

Discord is a free voice, video, and text chat application and digital distribution platform which was originally designed for the video gaming community. Today, it is heavily used within various hacking, doxing, and other communities associated with cybercrime. Some call it a modern-day IRC replacement. Discord allows users to create virtual servers which further divide into text and voice channels. Discord stores unlimited logs of all channels within every server. Anybody who joins a Discord server has full access to all server history. Access to a Discord server is granted through invites in the form of a URL. Discord is classified as a "deep web" resource, as Discord servers are unable to be indexed by conventional search engines such as Google. I present two initial thoughts on Discord investigative techniques.

- You must receive some type of invite in order to join a server. Once you have an invite, joining is easy and covert details are accepted. You can often find generic invite links within related forums or simply by asking a member of the server. Administrators will know that you joined, but will only see the details you provided during registration.

- Once you are in the server, you should have access to complete chat history since inception. In 2018, I was asked to assist the podcast Reply All with an investigation into the OG Users Discord. Members of this group stole social network accounts from random people and sold them to each other and the public. The primary avenue for communication was through a designated Discord server. The episode is titled ***130-The Snapchat Thief*** if you want to hear more.

Let's conduct a demonstration of finding, joining, and archiving a Discord server. First, I navigated to **Disboard** (disboard.org). This free service indexes numerous Discord servers which have open invitations. I conducted a search for the term "osint" and received one result of "Team Omega Cybersecurity and Analysis". The static Disboard link was the following.

https://disboard.org/server/join/605819996546924544

This immediately forwarded to the Discord link at the following address. This is the official invitation link which could be shared by members of the channel. I found it through Disboard because someone from the group likely posted the details with the intent of increasing usage. If you do not find any servers of interest on Disboard, try **Discord Me** (discord.me).

https://discord.com/invite/DbtGker

I was greeted with a login window asking for the name I wished to use in the channel. I provided OSINTAuthor and completed a Captcha. I was immediately given an error which demanded a cellular telephone number in order to join this server. This is common when using a VPN, hardened browser, and guest login. Therefore, I never recommend this route. Instead, register for a Discord account at https://discord.com/register, but take a few precautions. In my experience, creating the account from within Chrome appears less "suspicious" than Firefox. Connecting from a network without a VPN seems to allow registration while an IP address from a VPN results in another telephone demand. Therefore, I create a handful of accounts any time I am at a hotel or library. I create them from within my Windows VM using Chrome on the public Wi-Fi without a VPN. These accounts can be stored until needed.

I prefer to conduct all Discord investigations within their official Windows application while inside a virtual machine. Conduct the following steps to replicate my Discord machine.

- Clone your Original Windows 10 virtual machine previously created (Section One).
- Title your new VM "Discord" and conduct the following inside the Windows VM.
- Download and install the Discord app from https://discord.com/download.
- Download the first file titled "DiscordChatExporter.zip" from the website located at https://github.com/Tyrrrz/DiscordChatExporter/releases.
- Extract the contents of the zip file to your Windows VM Desktop.
- Launch DiscordChatExporter.exe from within the new folder.
- Launch the Discord app, provide your account credentials, and connect to the target Discord server (example: https://discord.com/invite/DbtGker).
- Press "Ctrl" + "Shift" + "I" on the keyboard to launch the Discord developer options.
- Click the arrows in the upper right and select "Application".
- Double-click "Local Storage" and select "https://discord.com".
- Press "Ctrl" + "R" on the keyboard and look for "Token" in the right menu.
- Select and copy the entire token key (without the quotes).
- Paste the token into the DiscordChatExporter program and click the arrow.
- Select the desired target server on the left and the target channel on the right.
- Choose the "CSV" export format and leave the dates empty.
- Choose the save location and click "Export".

The result is a text-only CSV file with the entire contents of the exported server and channel. You should repeat this process for each channel of your target server. In 2019, I located a Discord server used to share file-sharing links to recent data breaches. I exported all of the content, which included thousands of links to mega.nz, Google Drive, Dropbox, and other hosts. The day after I created the export, the Discord server was removed due to policy violations. However, I already had the full export and could investigate the links at my own pace. If desired, you could export an HTML report which would be much more graphically pleasing. I prefer the CSV option because I can import and manipulate the text easier than an HTML web page.

Discord is not the only platform for this type of communication, but I find it to be the most popular with amateur cyber criminals. **Slack** (slack.com) appears very similar, but it is targeted more toward professionals. **Riot** (about.riot.im) and **Tox** (tox.chat) each possess encrypted communications and better overall security, but adoption is lower than Discord. I believe you should be familiar with all of these environments, and be ready for an investigation within any of them. I keep an investigation VM with all four applications installed and configured. This can be a huge time-saver when the need arises.

Board Reader (boardreader.com)

Online forums provide a unique place for discussion about any topic. If you can think of the subject, an entire site full of people is probably hosting a discussion about the topic. These are usually referred to as user forums. Sometimes, these sites are excluded from being indexed by search engines. This can make locating them difficult. A new wave of forum search sites fills this void. Board Reader queries many forum communities, message boards, discussion threads, and other general interest groups which post messages back and forth. It also offers an advanced search which allows you to choose keywords, language, date range, and specific domain. If you have trouble filtering results on other forum search sites, this can be useful.

Craigslist Forums (forums.craigslist.org)

This forum is categorized by topic instead of location, but location filtering is supported. These forums are not indexed by most search engines, so a manual search is the only way to see the content. In order to search these areas, you may be required to create a free user account. As usual, you can use fictitious information in your profile. After logging in, you can search by keyword on this main page, but not by screen name. This option will identify posts matching the search terms within any of the topics.

The "Handle" option would search by username, but the search field for this query was removed in 2020. Fortunately, we can replicate it with a URL. The following URL displays all posts within the past 31 days by Craigslist user "honeygarlic".

https://forums.craigslist.org/?act=su&handle=honeygarlic

This page also provides two valuable pieces of information. The first line identifies the date the account was created and number of days since inception. Our example appears as follows.

since: 2004-06-16 06:40 (5992 days)

Next, we are notified if the user has deleted any posts in the past 31 days, as follows.

0% of this handle's posts in the last 31 days have been deleted

As a general rule, most people will use the same username across several sites. Craigslist is no exception. If you have identified a username of a target, a search on the Craigslist forums is worth a look. Although you will not get a result every time you search, the commentary is usually colorful when you do. When you locate a username of a target on the Craigslist forums, searching that username on other sites could provide an abundance of information.

Online Prostitution

Craigslist was once used by many prostitutes nationwide as an avenue to meeting "Johns". Likewise, many people used the site to locate a prostitute. In 2009, Craigslist was forced to remove the "Erotic Services" section that hosted these posts announcing this activity. In 2018, Backpage was forced offline and seized by various government agencies. Today, it is difficult to find a post offering prostitution on Craigslist and impossible on Backpage. This does not mean that the prostitutes and their clients simply stopped the potentially illegal behavior. Instead, they found new resources. There are many sites online that aid in prostitution and human trafficking. A few of the big players are listed here. I encourage you to investigate which services are applicable to your cities of interest.

https://5escorts.com
https://www.bedpage.com
https://cityoflove.com
https://cityxguide.com
http://craigslistgirls.com
https://www.eros.com
https://www.escort-ads.com
https://escortfish.ch
https://www.humaniplex.com
http://ibackpage.com
https://escortindex.com
https://onebackpage.com
https://openadultdirectory.com
https://preferred411.com
https://sipsap.com
http://skipthegames.com
https://stripperweb.com
https://www.slixa.com
http://theotherboard.com
https://www.tsescorts.com
https://www.tnaboard.com

Escort Review Websites

These types of services may be difficult for some readers to understand. I was also surprised when I first found them. This is where prostitution clients communicate with each other and leave reviews of their experiences with the prostitutes. These "Johns" document the slightest details of the experience including price, cleanliness, and accuracy of the photograph in the ad. Furthermore, this is the first location that will announce an undercover operation by the police. This is important for law enforcement, as this could create an officer safety issue. It is also how the police can determine when a new name or photo should be used in future covert ads. Another purpose for this data is to create documentation of the reviews of an arrested prostitute. This can prove to be valuable in court for the prosecution of offenses. There are several of these services, and every metropolitan area will have a preferred website by the customers. A Google search of "Escort reviews Anaheim" will get you to the popular options. Of course, replace Anaheim with your city of interest.

The Erotic Review (theeroticreview.com)

If you do not know of any individual services that prostitution clients are using in your area, The Erotic Review is a safe bet. Practically every metropolitan area has a presence here. Much of this site will not be available unless you join as a premium member. However, there should be plenty of visible free content for basic investigations. Most of the posts are unsuitable for this book. At the time of this writing, this site was blocking U.S. IP addresses. Switching my VPN to Canada or any other country bypassed the restriction.

Real World Application: While participating in an FBI Operation, I focused on locating juvenile prostitutes and women forced into the sex industry by pimps. One easy way to determine if a sex worker was traveling extensively was to search her number through various escort websites. If it returned numerous cities with postings, that was a strong indication that she was a full-time sex worker and was likely not traveling alone. Every contact that we made with traveling prostitutes resulted in the identification of the pimps that transported them to the stings.

Online Newspaper Comments

Practically every newspaper now has some sort of online presence. Most digital editions allow readers to leave comments about individual articles. These comments can usually be seen at the bottom of each web page. While the value that these comments add to the newsworthiness of each piece of news is debatable, the content can be important to an investigation. In years past, most newspapers hosted their own digital comment delivery

system within their website. This often resulted in a large headache while trying to maintain order, prevent feuds between readers, and delete direct threats. Today, most news websites use third-party services to host these comments. The most popular are Facebook and Disqus. When Facebook is utilized, most people use their real names and behave better than when using only a username on Disqus. Any complaints about the comment activity can be referred to Facebook since they technically store the content. Searching Facebook comments can be conducted through the technique previously explained.

In order to search for content within the Disqus comment system, you can conduct a custom Google search. First, it is important to understand how the Disqus system is recognized by Google. There is an option to log in to a Disqus account and you can "upvote" or "downvote" each comment to show your approval. The words visible on this page that were provided by Disqus are important for the search. The word "comments" will be visible on every Disqus provided environment and there will also be a link to disqus.com. Therefore, the following search on Google should provide any websites that have the Disqus comment delivery system and also have a reference to OSINT.

"osint" "disqus" "comments"

This may produce some non-Disqus results that happen to possess all three words, but those should be rare. This will also identify many pages that do not contain any comments whatsoever. In order to only receive results that actually have comments, alter your search to the following.

"osint" "disqus" "1..999 comments"

This instructs Google to only display results that contain the keywords "OSINT" and "Disqus" and also contain the exact phrase of any number between 1 and 999 followed immediately by the term "comments". This would provide results that contain any number of comments with the exception of "0" or over "1000". The "1..999" portion is the Google range operator that will display any number within the specified range.

Craigslist Auctions (craigslist.org)

Craigslist is one big online classified ad for every area of the world. The site can ease the pain of finding an apartment; provide numerous options for buying a vehicle locally; or assist in locating just about any item or service that you can imagine that is within driving distance of your home. It is also a landing spot for stolen goods, illegal services, and illicit affairs. While Craigslist offers a search option, the results are limited to active posts only. You can also only search within one category at a time. You can browse through the posts individually, but this will be overwhelming.

Government and private investigators have found much success in locating stolen goods within this site. To start, you must find the Craigslist site for your area. Often, simply visiting craigslist.org will direct you to the landing page for your geographical area. If this does not happen, navigate through your country, then your state, then your metropolitan area to see listings around you. If the theft occurred recently, a live search in the "for sale" section may produce results. I do not recommend searching from the main page, as there are no advanced options. Instead, click on any section title. For example, clicking on the "for sale" section will take us to that area. The top of the page will have a search field that will search all of the categories in this section. Additionally, we can filter by price range, posts that contain images, or terms that only appear in the title of the post.

Craigslist also has features that allow you to view results by list view, gallery view, or map view. These locations will only refer to the city of the item, and not exact GPS location. The gallery view can be used as a "photo lineup" to identify a stolen item. The map view can be beneficial when only looking for items within surrounding areas. Four new options on the upper right of every result page allow you to sort the items by newest listings (default), relevance, lowest price, and highest price. Most pages with items for sale will also allow you to filter the results so that only items being sold by individuals are listed. This would eliminate businesses and dealers. The default is to show both, and I recommend it unless you are overwhelmed by the number of results.

If a thief sells the item on Craigslist, he or she will usually delete the post after the transaction is complete. If the post is deleted, it will not be listed in the results of a search on Craigslist. This is where Google and Bing come into play. Both Google and Bing collect information from Craigslist posts to include in their search results. This collection can never be complete, but a large archive of posts is available. Searching Google or Bing with "site:craigslist.org" (without quotes) will search through archived posts on Craigslist that are both active and removed. Similar to the previous example, you can search "site:craigslist.org laptop Edwardsville" (without the quotes). This search produced 572 results that match these criteria on Google. These include the current posts that were available with the live search on craigslist.org as well as posts that have been recently deleted from Craigslist. If you wanted to focus only on a specific regional area of Craigslist, changing the search to "site:stlouis.craigslist.org laptop Edwardsville" would filter results. This example would only show listings from the St. Louis section of Craigslist. You can use any region in your custom searches.

The results that are still current will link to the actual post and display all content of the post. If a search result links to a post that has been deleted from Craigslist, a standard "page not found" error will be returned. You can still get additional information from this deleted post by looking through the text supplied on this search page. The brief description will often disclose an email address or telephone number. Some listings may have a cached view, but lately this has been rare. In a scenario where thousands of search results are presented by Google or Bing, you can add search terms to filter to a more manageable number of posts. Adding the make or model number of the product may quickly identify the stolen property.

You can also search by terms other than the product of interest. Many people that use Craigslist do not want to communicate through email sent from the website. Most users will include a telephone number in the post as a preferred method of communication. The overwhelming majority of these telephone numbers belong to the cellular telephone of the user submitting the post. This can be a huge piece of intelligence for an investigator attempting to identify a person associated with a telephone number. It is common that a criminal will purchase a cellular telephone with cash and add minutes to it as needed. This makes it difficult for someone to identify the criminal from the phone number. Ironically, the same criminal will post the telephone number as well as a name on a public internet site for the world to see. Sometimes, a person will post both a cellular and a landline telephone number on the same post. This allows an investigator to associate these two numbers, and a quick internet search should identify the owner of the landline telephone number.

Another way to search Craigslist posts is to identify screen names. Craigslist discourages inserting a screen name or email address within a post; however, most people have figured out how to bypass this limitation. Instead of someone typing their email address within their posts, they will insert spaces between the first portion of the email address (username) and the second portion of the email address (domain name). For example, instead of the user typing their email address as JohnDoe911@gmail.com, he or she may identify the account as "JohnDoe911 at gmail com". This would be enough to prevent Craigslist's servers from identifying the text as an email address and prohibiting the post. Fortunately for the investigator, this information is indexed by Craigslist and other search engines to be retrieved.

You can search any keyword in either the official Craigslist site or on Google and Bing using the "site" operator. In my experience, Bing offers more results of archived Craigslist posts than Google. If you do not have success with Bing, Google should be searched as well. Many private investigators find the "personals" section of interest. The "Casual encounters" area is well known for extramarital affairs. If you want to only search all live Craigslist posts, regardless of which geographical area it exists, you can use sites such as totalcraigsearch.com, adhuntr.com, and searchalljunk.com.

Craigslist has a few advanced search operators that may be of interest. It supports a phrase search with quotation marks such as "low miles". It accepts the hyphen (-) operator to exclude terms such as honda black -red. This search finds postings that have 'honda' and 'black' but not 'red'. A pipe symbol (|) provides "OR" searches such as honda | toyota. This search finds postings that have 'honda' or 'toyota' (or both). You can group terms together in parentheses when queries are complicated. A search of red (toyota | honda) -2000 -2001 finds listings that have 'red' and either 'honda' or 'toyota' (or both) but do not have 2000 or 2001. Wildcards are as follows.

Hond* civ* (match "honda civic", "honda civil", etc.)
wood floo* (matches "wood floors", "wood flooring", etc.)
iphone* (matches "iphone", "iphones", "iphone5", etc.)

Craigslist's email alert feature has made third-party tools for this purpose unnecessary. After logging in to your account, you can customize alerts to send an email to you when specific search terms are located.

Real World Application: Many thieves will turn to the internet to unload stolen items. While eBay requires banking information or a credit card to use their services, most thieves prefer Craigslist's offer of anonymity. My local police department successfully located a valuable stolen instrument this way and set up a sting to arrest the thief. Often, the thief will be willing to bring the item to you in order to get some quick cash. Another tip that has helped me during investigations is to look for similar backgrounds. When I had a group of gang members stealing iPhones from vehicles and pockets, they would sell them right away on Craigslist. Since there were hundreds of legitimate iPhones listed, identifying the stolen units can be difficult. By looking for similarities in the backgrounds, I could filter the list into interesting candidates. Finding unique backgrounds, such as tables or flooring, within several posts can be suspicious. Additionally, I have found posts that include "hurry", "must sell today", and "I will come to you" to be indicators of illegal activity.

eBay Auctions (ebay.com)

eBay is an online auction site. Since the site requires a user's financial information or valid credit card to post items for sale, many thieves have moved to Craigslist to unload stolen goods. eBay offers an advanced search that will allow filters that limit to auctions from a specific location, or specified distance from the location. On any search page, there is an "Advanced" button that will display new options. Of these options, there is a category titled "show results". The last option in this category is titled "items near me". Here, you can select a zip code and filter results to a minimum of 10 miles from the zip code selected. This will now allow you to search for any item and the results will all be from sellers near a specific zip code. This location option will remain active as you search for different keywords. These searches will only search current auctions that have not expired. In order to search past auctions, select the "Completed listings" option under the category of "Search including". If you want to conduct your searches directly from a URL, or if you want to bookmark queries that will be repeated often, use the following structure. Replace TERMS with your search keywords and USER with your target's username.

Keyword: ebay.com/dsc/i.html?&LH_TitleDesc=1&_nkw=TERMS
Sold: ebay.com/sch/i.html?_from=R40&_nkw=TERMS&LH_Sold=1&LH_Complete=1
Complete: https://www.ebay.com/sch/i.html?_from=R40&_nkw=TERMS&LH_Complete=1
Username: https://www.ebay.com/usr/USER
User Feedback: https://feedback.ebay.com/ws/eBayISAPI.dll?ViewFeedback2&userid=USER
User Items: https://www.ebay.com/sch/USER/m.html
User: http://www.ebay.com/sch/ebayadvsearch/?_ec=104&_sofindtype=25&_userid=USER
User Followers: https://www.ebay.com/usr/USER/followers#followers
User Following: https://www.ebay.com/usr/USER/all-follows?priflwtype=people#people

Flippity (flippity.com)

An alternative to the location feature on the official eBay site is Flippity. This site performs the same function as mentioned previously, but with less work on the user's part. The results of your search will appear on a map with the ability to minimize and expand the radius as desired. This is a quick way to monitor any type of items being sold in a specific community.

GoofBid (goofbid.com)

Not everyone uses spellcheck. Some people, especially criminals, will rush to list an item to sell without ensuring that the spelling and grammar are correct. You could conduct numerous searches using various misspelled words, or you can use GoofBid. This site will take your correctly spelled keyword search and attempt the same search with the most commonly misspelled variations of the search terms. Another alternative to this service is **Fat Fingers** (fatfingers.com).

Search Tempest (searchtempest.com)

If you find yourself searching multiple geographical areas of Craigslist and eBay, you may desire an automated solution. Search Tempest will allow you to specify the location and perimeter for your search. It will fetch items from Craigslist, eBay, and Amazon. You can specify keywords in order to narrow your search to a specific area. Advanced features allow search of items listed within the previous 24 hours, reduction of duplicates, and filtering by categories. While I encourage the use of these types of services, I always warn people about becoming too reliant on them. These tools could disappear. It is good to understand the manual way of obtaining data.

OfferUp (offerup.com)

This service is steadily stealing the audience currently dominated by Craigslist. OfferUp claims to be the simplest way to buy and sell products locally. A search on their main page allows you to specify a keyword and location. The results identify the usual information including item description and approximate location. OfferUp follows the eBay model of including the seller's username and rating. The unique option with OfferUp is the ability to locate the actual GPS coordinates associated with a post instead of a vague city and state. This information is not obvious, but can be quickly obtained. While on any post, right-click and choose to view the page source. Inside this new tab of text should be two properties titled place:location:latitude and place:location:longitude. You can search for these in your browser by pressing "Ctrl" + "F" (Windows) or "command" + "F" (Mac). Next to these fields should display GPS coordinates. In my experience, these precise identifiers will either identify the exact location of the target, or a location in the neighborhood of the suspect. I would never rely on this all the time, but I have had great success getting close to my targets through this technique.

Amazon (amazon.com)

Amazon is the largest online retailer. Users flock to the site to make purchases of anything imaginable. After the receipt of the items ordered, Amazon often generates an email requesting the user to rate the items. This review can only be created if the user is logged in to an account. This review is now associated with the user in the user profile. An overwhelming number of users create these product reviews and provide their real information on the profile for their Amazon account. While Amazon does not have an area to search for this information by username, you can do it with a search engine. A search on Google of site:amazon.com followed by any target name may link to an Amazon profile and several item reviews. The first link displays the user profile including photo, location, and the user's review of products purchased.

This technique of using Google or Bing to search for profiles on websites that do not allow such a search can be applied practically everywhere. Many sites discourage the searching of profiles, but a search on Google such as "site:targetwebsite.com John Doe" would provide links to content matching the criteria. The difficulty arises in locating all of the sites where a person may have a profile. By now, you can search the major communities, but it is difficult to keep up with all of the lesser-known networks.

FakeSpot (fakespot.com)

There is an abundance of fake reviews on Amazon, which can make it difficult to determine which reviews accurately describe a product and which are provided by employees associated with the seller. FakeSpot attempts to identify products that are likely misrepresented by the review community. During a search for a remote-

controlled drone, I found that the Amazon "Best Seller" possesses over 53% fake reviews, and top reviewers "tri nguyen" and "EUN SUN LEE" appear to be automated reviewers based on other products. This service also supports analysis of reviewers on Yelp and Trip Advisor. FakeSpot now requires you to install a browser extension. If this service is valuable to your investigations, I find this service effective. If you rarely need this type of data, I would avoid any unnecessary extensions.

Pinterest (pinterest.com)

Pinterest is an online "pinboard" where users can share photos, links, and content located anywhere on the internet. It is a way to rebroadcast items of interest to a user. People that follow that user on Pinterest can keep updated on things that the user is searching and reading. The search feature on the main website is useful for keyword searches only. It will search for any term and identify posts that include those words within the description. A search of my last name displayed several photos of people. Clicking each of these links will present the full-page view of the photo and any associated comments. This page will also identify the full name of the person that uploaded the content and the original online source. Clicking on the full name of the user will open the user profile which should include all "pinned" content. Unfortunately, you cannot search a person's full name or username on Pinterest and receive a link to their profile page. To do this, you must use Google. The following direct search URLs will identify the usernames (BILL) and keywords (CRAFTS) present on Pinterest.

Username: https://www.pinterest.com/BILL/
User Pins: https://www.pinterest.com/BILL/pins
User Boards: https://www.pinterest.com/BILL/boards
User Followers: https://www.pinterest.com/BILL/followers/
User Following: https://www.pinterest.com/BILL/following
Pins Search: https://www.pinterest.com/search/pins/?q=CRAFTS
Boards Search: https://www.pinterest.com/search/boards/?q=CRAFTS
Google Search: https://www.google.com/search?q=site:pinterest.com+CRAFTS

BugMeNot (bugmenot.com)

This utility may violate your departmental policies about online research. BugMeNot allows users to share their logins for various websites with the world. Technically, users of this service are giving the public consent to use their credentials in order to access data behind a login. However, this could violate the terms of service for a specific website. I once needed to access a private web forum which was hidden behind a login portal. This forum was not accepting new members. Searching the URL on BugMeNot revealed a username and password shared by a member of the forum for public use. This allowed me to access the site and find my desired data.

IntelTechniques Communities Tool

Similar to the previous search tools, this option attempts to simplify the various search techniques presented within this chapter. The entire tool is quite large, and could not fit into a graphical representation on a single page here. This tool should replicate all of the specific URLs cited within this topic. While the chances of your target appearing here are lower than large social networks, this resource should not be ignored. In my experience, the details obtained about a target from online communities are usually much more intimate and personal than the public blasts on the larger sites. The online version is updated often and can always be accessed on my site at **https://inteltechniques.com/tools/Communities.html.**

CHAPTER FIFTEEN
EMAIL ADDRESSES

Searching by a person's real name can be frustrating. If your target has a common name, it is easy to get lost in the results. Even a fairly unique name like mine produces almost 20 people's addresses, profiles, and telephone numbers. If your target is named John Smith, you have a problem. This is why I always prefer to search by email address when available. If you have your target's email address, you will achieve much better results at a faster pace. There may be thousands of John Wilsons, but there would be only one john.wilson.77089@yahoo.com. Searching this address within quotation marks on the major search engines is my first preference. This should identify web pages which include the exact details within either the content or the source code. This is the "easy stuff" which may present false positives, but should provide immediate evidence to the exposure of the target account. I then typically search the username portion of the email address by itself in case it is in use within other providers, such as Gmail, Hotmail, Twitter, LinkedIn, etc. Let's conduct an example assuming that "john.wilson.77089@yahoo.com" is your target. You could place this email within quotes and execute through each service manually, or use the following direct search URLs to expedite the process.

Google Email: https://google.com/search?q="john.wilson.77089@yahoo.com"
Google Username: https://google.com/search?q="john.wilson.77089"
Bing Email: https://bing.com/search?q="john.wilson.77089@yahoo.com"
Bing Username: https://bing.com/search?q="john.wilson.77089"
Yandex Email: https://yandex.com/search/?text="john.wilson.77089@yahoo.com"
Yandex Username: https://yandex.com/search/?text="john.wilson.77089"

The custom search tools presented at the end of this chapter and the next will automate this entire process. Next, you should verify the target email address.

Emailrep.io (emailrep.io)

This is an email verification service with many additional features. Below is an actual search result, and I provide an explanation after each detail [within brackets]. As you will see, this is an impressive free search, and at the top of my list for every investigation.

"email": "redacted@gmail.com",	[Email address provided]
"reputation": "high",	[Likelihood to be a real email address]
"suspicious": false,	[Indications of spam or malicious use]
"references": 20,	[Number of online references]
"blacklisted": false,	[Blocked by spam lists]
"malicious_activity": false,	[Known phishing activity]
"malicious_activity_recent": false,	[Known recent phishing activity]
"credentials_leaked": true,	[Password present within data leaks]
"credentials_leaked_recent": false,	[Password present within recent leaks]
"data_breach": true,	[Address present within known breaches]
"first_seen": "07/01/2008",	[Date first seen online]
"last_seen": "02/25/2019",	[Date last seen online]
"domain_exists": true,	[Whether domain is valid]
"domain_reputation": "n/a",	[Reputation of domain]
"new_domain": false,	[Domain recently registered]
"days_since_domain_creation": 8795,	[Days since the domain was registered]
"spam": false,	[Marked as spam]
"free_provider": true,	[Free provider such as Gmail]
"disposable": false,	[Disposable provider such as Mailinator]

"deliverable": true,	[Inbox able to receive mail]
"accept_all": false,	[Address is a catch-all]
"valid_mx": true,	[Domain possesses an email server]
"spf_strict": true,	[Domain email security enabled]
"dmarc_enforced": false,	[Dmarc email security enabled]
"profiles": "youtube","google","github","twitter"	[Profiles associated with email]

This provides detail we cannot find anywhere else. In this example, I now know that my target email address has been in use for at least twelve years and appears within at least one data breach. This encourages me to explore the breach search methods explained in a moment. The following "explainer" which was included at the end of my result provides even more value.

"Not suspicious. This email address has been seen in 3 reputable sources on the internet, including Twitter. It has been seen in data breaches or credential leaks dating back to 07/18/2017, but not since 02/25/2019. SPF and DMARC are strictly enforced, so this email address is not spoofable. We've observed no malicious or suspicious activity from this address. "

An additional email verification option is **Email Hippo** (tools.verifyemailaddress.io). This service provides minimal data, but may identify something missing from the previous superior option. When I searched each of my Gmail accounts, I received "OK" as a response. When I changed one character, I immediately received "BAD" as a result. As you validate your target addresses through Email Hippo, the responses appear at the bottom as a collection. Choosing the Export option allows you to download all results. This service limits free daily usage by IP address and browser cookies.

Email Assumptions

You may know about one address, but not others. It can be productive to make assumptions of possible email addresses and use the verifiers to see if they exist. For example, if your target's name is Jay Stewart and he has an email address of jay112003@yahoo.com, you should conduct additional searches for the addresses of jay112003@gmail.com, jay112003@hotmail.com, jay112003@live.com, and others. If you already know your target's username, such as a Twitter handle, you should create a list of potential email addresses. If I had no email address or username for my target (Jay Stewart), but I knew that he worked at the Illinois Medical District Commission (medicaldistrict.org), I may start searching for the following email addresses.

jstewart@medicaldistrict.org
jay.stewart@medicaldistrict.org
j.stewart@medicaldistrict.org
stewartj@medicaldistrict.org

These are merely assumptions of potential addresses. Most, if not all, of them do not exist and will provide nothing for me. However, if I do identify an existing address, I now have a new piece of the puzzle to search. These addresses can be verified against the previous three tools.

Email Format (email-format.com)

If the previous email assumption techniques were unproductive or overkill for your needs, you may want to consider Email Format. This website searches a provided domain name and attempts to identify the email structure of employee addresses. When searching medicaldistrict.org, it provided several confirmed email accounts under that domain and made the assumption that employee emails are formatted as first initial then last name. Our target would have an email address of jstewart@medicaldistrict.org according to the rules. I use this service to help create potential email lists from names collected from Facebook, Twitter, and other social networks. I can then verify my list with the services previously mentioned.

Gravatar (gravatar.com)

This service is responsible for many of the small image icons that you see next to a contact in your email client. You may notice that some incoming emails display a square image of the sender. This is configured by the sender, and this image is associated with any email address connected. You do not need to wait for an incoming message in order to see this image. While the Gravatar home page does not offer an email address search option, we can conduct a query directly from the following URL, replacing the email address with your target information. This image can then be searched with a reverse image query as explained later.

https://en.gravatar.com/site/check/test@gmail.com

Compromised Accounts

Email addresses are compromised regularly. Hacker groups often publicly post databases of email addresses and corresponding passwords on websites such as Pastebin. Manually searching for evidence of compromised accounts can get complicated, but we will tackle this later in the book. Several online services now aid this type of investigation. These services provide one minimal piece of information about any email address entered. They disclose whether that address appears within any publicly known hacked email databases. While most will never disclose the owner, any email content, or passwords, they will confirm that the target's email account was compromised at some point. They will also identify the service which was targeted during the breach.

This helps us in two ways. First, it confirms an email address as valid. If your suspect account is todd007@gmail.com, and that address was compromised in a breach in 2015, you know the address is valid, it was active, and is at least a few years of age. Second, you know the services which need to be investigated. If todd007@gmail.com was included in the Dropbox and LinkedIn breaches, you should attempt to locate those profiles. Compromised account data is absolutely the most beneficial technique I have used within my investigations in the past five years. We will dive much deeper into data sets later, but let's focus on online services first. These are also good websites to check your own address in order to identify vulnerabilities.

Have I Been Pwned (haveibeenpwned.com)

This is a staple in the data breach community. This site allows entry of either a username or email address, but I find only the email option to be reliable. The result is a list and description of any public breaches which contain the provided email address. The descriptions are very helpful, as they explain the type of service associated and any details about the number of users compromised. In a test, I searched an old email of mine that was used to create several covert accounts many years prior. The result was notification that the email address had been compromised on six websites, including Bitly, Dropbox, LinkedIn, Myspace, River City Media, and Tumblr. The data present here is voluntarily contributed to the owner of the site. Cyber thieves steal various credential databases and allow Have I Been Pwned (HIBP) to confirm the legitimacy. HIBP then makes the content searchable and credits the thief by name. The thief can then charge more money for the stolen goods as HIBP has vetted the content. It's a weird game.

Searching content within the HIBP website is straight-forward. You may recall the method from the chapter about search tools which explained a way to query an email address through HIBP directly from a URL which presents text-only results. Let's revisit that technique. The following URL queries test@test.com against the HIBP database.

https://haveibeenpwned.com/unifiedsearch/test@test.com

Now, let's compare results. Figure 15.01 displays the output from the official website. Immediately after, I present the text provided with this custom URL.

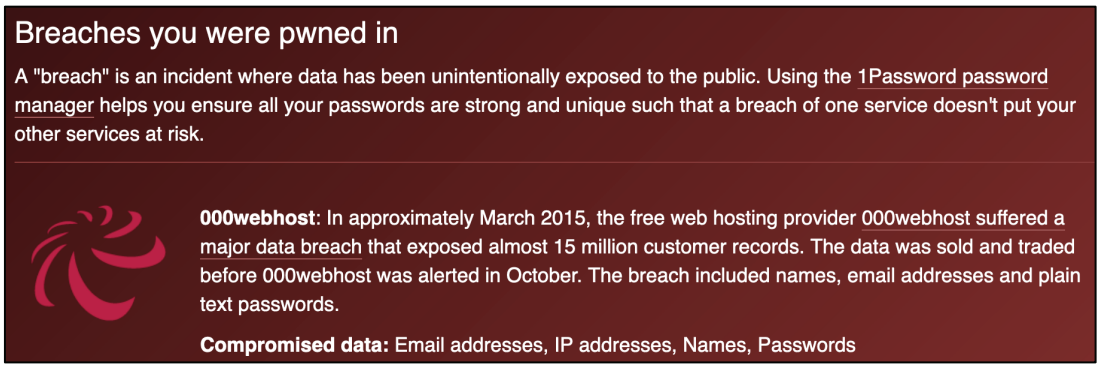


Figure 15.01: An email search result from Have I Been Pwned.

"Name": "000webhost",
"Title": "000webhost",
"Domain": "000webhost.com",
"BreachDate": "2015-03-01",
"AddedDate": "2015-10-26T23:35:45Z",
"ModifiedDate": "2017-12-10T21:44:27Z",
"PwnCount": 14936670,
"Description": "In approximately March 2015, the free web hosting provider 000webhost suffered a major data breach that exposed almost 15 million customer records. The data was sold and traded before 000webhost was alerted in October. The breach included names, email addresses and plain text passwords.",
"DataClasses": "Email addresses","IP addresses","Names","Passwords"
"IsVerified": true,
"IsFabricated": false,
"IsSensitive": false,
"IsRetired": false,
"IsSpamList": false

As you can see, the text version contains more details and can be easily copied and pasted into a report. Have I Been Pwned is an amazing tool, but it does not contain all known breaches. Some "white hat" hackers eagerly share the data leaks and breaches which they discover in order to receive acknowledgement from the site. Once HIBP has verified the legitimacy and source of the data, it becomes much more valuable in the black markets. Many researchers have accused this website of encouraging data theft, as it can be sold for a higher amount once the owner has vetted the content. Many criminals who deal with stolen credentials dislike this site and its owner. They do not share their goods and try to keep them off of the radar of the security community. Therefore, we must always utilize every resource possible.

Dehashed (dehashed.com)

While Have I Been Pwned is often considered the gold standard in regard to breached account details, we cannot ignore Dehashed. It takes a more aggressive approach and seeks breached databases for their own collection. Using the same email address provided during the previous test, I received two additional breach notifications. These included lesser-known breaches which had not yet publicly surfaced on HIBP. When combining results from both of these services, you would now know that this target email address was likely a real account (you received results); it had been used online since 2012 (the date of the oldest breach according to HIBP); it is connected to an employment-minded individual (LinkedIn); and it exists in spam databases as a U.S. consumer (River City Media).

I believe that Have I Been Pwned and Dehashed complement each other, and one should never be searched without the other. This search alone often tells me more about a target email address, even if it never identifies

the owner. Dehashed allows unlimited search for free, but will not disclose passwords without a premium account. They advertise the ability to see all associated passwords for a small fee. I have tried this service in the past, and it worked well. However, I believe that paying a company to disclose stolen passwords might exceed the scope of OSINT. It may also violate your own employer's policies. Please use caution. Unfortunately, Dehashed now requires you to be logged into a free account in order to conduct any queries, even at a free access level.

Spycloud (spycloud.com)

This service is a bit different. They are extremely aggressive in regard to obtaining fresh database breaches. They possess many data sets which are not present in HIBP. However, they do not display details about accounts which you do not own. Our only option is general details through their free API. The following URL submits a query for test@email.com.

https://portal.spycloud.com/endpoint/enriched-stats/test@email.com

The results are in JSON format, which may appear like pure text within your browser. The following result was presented to me when I searched an email address within my own domain. They basically tell you that the email address queried is present within multiple database breaches, but the identity of each is not available. I use this service to simply verify an email address.


```
  "you": {
     "discovered": 7,
     "records": 8,
     "discovered_unit": "Months"
  "company": {
     "discovered": 1,
     "records": 11,
     "discovered_unit": "Month",
     "name": "email.com"
```


Hudson Rock (hudsonrock.com)

When searching an email address, Hudson Rock identifies whether it has been seen within stealer log data. Stealer logs are explained much later in the book, but presence within them indicates that the email owner's computer has been compromised by a virus. We can query their API with the following URL.

https://cavalier.hudsonrock.com/api/json/v2/preview/search-by-login/osint-tools?email=test@email.com

Cybernews (cybernews.com/personal-data-leak-check)

This service only provides a "true" or "false" identifying the presence of the email within a breach. The following URL submits a text-only query. This could verify an email address exists.

https://check.cybernews.com/chk/?lang=en_US&e=test@test.com

Passwords

The previous options only display the services which are associated with a target email account due to that data being stolen and publicly released. Passwords are not visible unless you pay a premium to Dehashed or Spycloud. Other sites present more robust options for displaying redacted and unredacted passwords for billions of accounts without any fees. I believe this is OSINT data, but your employer's policies may disagree. Let's approach cautiously with some demonstrations.

Leak Peek (leakpeek.com)

The benefit of this service is that it displays a partial view of passwords associated with email addresses within a breach. It possesses a decent data set of 8 billion credentials. This consists of a "Combo List", which will be acquired later. The following examples identify the masking protocol used to block portions of each password.

test@email.com:55Ji*******
test@email.com:Big6****

Note that Leak Peek allows query by email address, username, password, keyword, and domain. We will use this resource again within upcoming chapters.

Breach Directory (breachdirectory.org)

The results here seem very similar to Leak Peek, but redundancy is always valuable. However, full passwords are available if you are willing to do some work. Consider the following results for test@email.com.

Ex7l****	dc3245ecdcf2e40b140e121a014cc37f70239f41
Exig***	7e39ea215613a23a6b33f1dabc340ccf65dfa243

The left column displays the typical redacted passwords and the right column displays SHA-1 hash values of the full passwords. I will explain much more detail about hash values later in the book. For now, this is a representation of the entire password, but we must decrypt the has to reveal the true password. There are severa online options, and I will explain my recommended approach with the data breaches chapter. For demonstration, I will use **MD5 Decrypt** (md5decrypt.net/en/Sha1). I entered dc3245ecdcf2e40b140e121a014cc37f70239f41 into this website and received the following response.

dc3245ecdcf2e40b140e121a014cc37f70239f41 : **Ex7layer**

I now know the full password without creating any account or downloading any breach data. These services are only an introduction into breach data. We have a lot more to discuss later in the book.

PSBDMP (psbdmp.ws)

Up to this point, all of these services display content received directly from known data breaches. PSBDMP takes a different approach. It monitors Pastebin for any posts including email addresses and/or passwords. There is no search field, but their free API presents detailed results with the following URL structure.

https://psbdmp.ws/api/search/test@test.com

The following is an actual result.

"id": "69yKWrZE",
"tags": "none",
"length": 25004,
"time": "2019-07-05 14:30",
"text": "$contact->setEmail('test@test.com');

The "id" is the Pastebin identifier. We can access the original file with the following URL.

https://pastebin.com/69yKWrZE

By visiting the Pastebin URL, we can see the full data, including the following excerpt.

```
$contact->setFirstName('John');
$contact->setLastName('Doe');
$contact->setEmail('test@test.com');
$contact->setEmail('example@mail.com');
$contact->setPhoneCountryCode('123');
$contact->setPhone('12345678');
```

If this had not been intentional test data, the results could have been very valuable for an investigation. We can also use this service to search for exposed password details. If I wanted to know if "password1234" had ever appeared within current or deleted Pastebin files, I would navigate to the following.

https://psbdmp.ws/api/search/password1234

The following is an actual result, displaying the Pastebin file, date, and credentials.

```
"id": "fEV4hrbe",
"tags": "none",
"length": 189357,
"time": "2016-12-12 19:34",
"text": "rd123 sketches5973:password1234 sketches5973:passw"
```

We can also sort through their archives with the following Google query.

site:psbdmp.ws "test@gmail.com"

The first result is https://psbdmp.ws/WjwCkNL4. Clicking that link presents a dead page. However, modifying the URL as follows presents the original content.

https://pastebin.com/WjwCkNL4

I once had an investigation where Google had indexed a PSBDMP Pastebin hit, but searching the Pastebin site through Google did not reveal the data. I also once investigated a Pastebin link to a client doxing at https://pastebin.com/qbkWTxCW. Pastebin had already removed the data and it was no longer accessible. A URL of https://psbdmp.ws/api/search/qbkWTxCW contained the full text file archived from Pastebin by PSBDMP. I believe the free version of PSBDMP provides great value for online investigators. Their paid API can be more beneficial if you have a need for that service. I have only used it a few times, and still have a few free trial credits remaining. More details can be found at https://psbdmp.ws.

IntelligenceX (intelx.io)

You can search IntelligenceX for free and receive partial results, or create a free trial to see everything. In my experience, you are only limited to the number of burner "trial" email addresses with which you have access. While I have used this site for the free tier and an occasional free trial, I do not recommend any of the paid services. We will replicate the same results for free throughout the book and within our own tools. In fact, many of the tools on their site were copied verbatim without attribution from the tools which are included free here.

Avast Hack Check (avast.com/hackcheck)

Avast offers a free search engine which identifies email addresses which appear within known data breaches and leaks. However, it should be avoided. If you enter any email address within this page, the account immediately receives an email from Avast about the search. Furthermore, Avast enrolls the email address into their invasive email newsletter database. Instead consider the "Friends Check" option at the following URL.

https://www.avast.com/hackcheck/friends-check

This website queries the same database maintained by Avast, but does not add the email address to their marketing campaigns or notify the account owner. This service will not identify any specific breach, but could be used as another email address verification option. If you receive a positive result, you know the email address has been used in the past and appears within a breach.

LeakIX (leakix.net)

This service indexes various areas of the internet looking for data leaks which may contain sensitive information. We can query through the site or directly via the following URL structure, which identifies an email address associated with a website owner's WordPress account.

https://leakix.net/search?scope=leak&q=%22test@test.com%22

The following is a partial result.

Japan GMO Internet,Inc 2022-06-21 01:45 2022-11-06 01:55 https://www.houyoukai.sai-net.work
ASN: 7506, Open ports: 443, Certificate domains: www.houyoukai.sai-net.work, houyoukai.sai-net.work
Found Wordpress users (CVE-2017-5487):
User #1 testtest-com
Name: test@test.com

Email Reputation

When I am investigating an email address, I want to know its reputation or spam score. If the address has been blacklisted or reported as spam, I won't waste too much time attempting to identify the true owner. There are two reliable services which we can query with the following URLs.

https://cleantalk.org/email-checker/michael@inteltechniques.com
https://spamdb.org/blacklists?q=michael@inteltechniques.com

An example record may appear as follows.

Record	Sites	Blacklisted	Real	Disposable	Updated
spam@gmail.com	14	True	Fake	True	Nov 14, 2022 22:52:42

Hunter (hunter.io/email-verifier)

This service advertises the ability to display email addresses associated with a specific domain. However, the "Verifier" tool can also be valuable. This URL allows query of an individual email address. It immediately provides details similar to the first two services mentioned in this chapter, such as the validity of the address. From there, it displays any internet links which contained the email address within the content at some point. This appears to be sourced from their own database, as many links are no longer present on the internet, and not indexed by Google. I once searched my target email address here and confirmed that it was present within a forum associated with terrorism. The forum had recently been removed, but this evidence was permanent.

OCCRP (data.occrp.org)

Self-identified as "The global archive of research material for investigative reporting", this service possesses an amazing amount of data. We will dig into this resource within a later chapter, but we should acknowledge the email options here. A query of any email address immediately displays documents associated with the account. Additionally, any keyword search which provides results can be filtered by email address. Let's conduct an

example. When I search for "inteltechniques", I receive two results, both of which appear to be PDF files from a journalism conference in 2017. I could download and read the documents, or click the filters to the left. When I click on the "Email" option, I am notified that the following email addresses are included within these documents. This is great way to find additional email addresses which may be associated with your target.

jaimi.dowdell@gmail.com
james.grimaldi@wsj.com
mark.maremont@wsj.com

Spytox (spytox.com)

This is a "White Pages" style of query and you will potentially see a name, city, and telephone number associated with the account. The paid options are never worth the money.

XLEK (xlek.com)

I often find email addresses of my targets within this service, but it does not allow you to query by email address. We will need to rely on Google to assist. Assume your target is "mike@gmail.com". The following Google query produces over 200 results.

site:xlek.com "mike@gmail.com"

These results include the actual target email address queried and anything else which also matches, such as "johnson.mike@gmail.com" and "smith.mike@gmail.com". This overall strategy should apply toward any people search websites which display email addresses but do not provide a specific email search field.

That's Them (thatsthem.com)

The majority of email addresses and usernames I have searched through this service returned no results. However, on occasion I received detailed results such as full name, address, phone number, and vehicle information. Although this is rare, I believe That's Them should be on your list of email and username search resources.

Search People Free (searchpeoplefree.com/cyberbackgroundchecks.com)

This data set is suspected to be generated from various marketing database leaks. Because of this, we usually receive much more complete results. Most results include full name, age, current home address, previous addresses, telephone numbers, family members, and business associations.

Proton Mail (protonmail.com)

Proton Mail provides arguably the most popular secure and encrypted email service. Because of this, many criminals flock to it. In 2020, at least half of my criminal email investigations were associated with a Proton Mail account. When this happens, I want to know the date of account creation. The following steps should be conducted while logged in to a free Proton Mail account.

- Create a new "Contact", add the target email address, then save it.
- Access the contact and click the "Email Settings" icon.
- Click the "Show advanced PGP settings" link.

The result should display the creation date of the "Public key". This is usually the creation date of the email account, but not always. If your target generated new security keys for an account, you will see that date instead. Fortunately, this is very rare. Figure 15.02 displays the result when searching the email address of a colleague. I

now know he has had this account since at least March 10, 2016. If the result displayed a recent date, I would suspect it was a new "burner" account created for a malicious purpose.

FINGERPR...	CREATED	EXPIRES	TYPE	STATUS
900a5c16b...	Mar 10, 2016	-	RSA (2048)	PRIMARY

Figure 15.02: A Proton Mail public key announcing account creation date.

This technique requires you to possess a free or paid Proton Mail account. It provides the easiest way to generate, view, and document the details. However, we can replicate this entire process without a requirement to log in to an account, but the results are not as reliable. Assume your target email is notmyemail@protonmail.com. First, we must query it to determine if it is present within the Proton Mail system with the following URL.

https://api.protonmail.ch/pks/lookup?op=get&search=notmyemail@protonmail.com

If your browser prompts you to download a file titled "pubkey.asc", this indicates that the address exists. If you receive a message of "No Key Found", then it does not. If the address exists, navigate to the following URL.

https://api.protonmail.ch/pks/lookup?op=index&search=notmyemail@protonmail.com

That URL currently provides the following response.

```
info:1:1
pub:74ecf3959bac5eba2bd636e204fac101b757d18f:1:2048:1623879788::
uid:notmyemail@protonmail.com <notmyemail@protonmail.com>:1623879788::
```

The last set of digits (1623879788) represents an Epoch Unix timestamp, which was explained in Chapter Twelve during the source code review of Instagram profiles. We can convert that number into a date and time at https://www.unixtimestamp.com/index.php. This tells me that the account was created on or before Wednesday, June 16, 2021 at 21:43:08 GMT. Be warned that I have experienced false positives with both of these URLs if you are connected to a VPN server which is abusing this process or you conduct multiple queries within a few seconds. Technically, they should obtain the same data from the same source, but I have had many failures. This is why I prefer the manual method explained at the beginning of this section.

Domain Connections

Every domain name registration includes an email address associated with the site. While many people will use a privacy service to hide personal details, this is not always the case. Fortunately, many free services have been collecting domain registration details and offer queries of current and archived domain registration data. They will identify domain names that were registered with the target email address entered. This is beneficial when you have a tech-savvy target that may have registered websites of which you are not aware. This also works on domains that no longer exist. As a test, I provided an email address of brad@notla.com. The following results identify the details found by each service. The use of affiliate IDs obtained by services such as Analyze ID will be explained during the domain instruction presented later.

Whoxy (whoxy.com/reverse-whois)
Full Name & Mailing Address
Telephone Number
Nine Owned Domain Names
Registrar and Host Details

Whoisology (whoisology.com)
Full Name & Mailing Address
Telephone Number
Twenty Owned Domain Names
Registrar and Host Details

AnalyzeID (analyzeid.com)
Amazon Affiliate ID
AdSense Affiliate ID
Five Owned Domain Names

ScamSearch (scamsearch.io)

This free service allows query of an email address to identify any association with reported online scams. The standard search box allows for an exact search of any email address, but we can submit a wild card search via URL. The following displays any Proton Mail email addresses in the scam database.

https://scamsearch.io/searchadvanced?_emailwild=email&search=protonmail.com

This query reveals "mamoru.uchida@protonmail.com" as a scammer from the website HackForums.net. I always recommend a wild card search over a standard query, as it may catch misspellings or similar accounts associated with your target. While this site offers username and telephone search, I have found the results to be minimal. I only use it for email and Bitcoin queries.

Imitation

The assumptions of email addresses can be valuable in discovering potential valid accounts, as mentioned earlier. Imitation of any target email addresses can reveal more details, and confirm association with online activities. Consider the following example. Your target email address is bill@microsoft.com, and you want to know if he is a Mac or Windows user. You could first navigate to apple.com/account and attempt to make an Apple account with that address. If allowed to proceed past the first screen, then that user does not already have an account associated with the target address. If you are informed that the email address is already in use, then you know that your target is a Mac user, and that specific address controls the account. You could navigate to signup.live.com and attempt to create an account with the address. If denied, you know that your target is already a Windows user and that address controls the account.

This method can be replicated across practically all websites and services. I have used this technique to confirm that target email addresses are associated with services from Yahoo, Gmail, Facebook, Twitter, and many others. I have also identified real-world services in use by my target by attempting to create accounts with local cable, power, and water companies, supplying the target email account and being notified that the address was "already in use". Knowing the local cable provider of a suspect can seriously limit the geographical area where he or she could be residing. Be sure not to fully execute any account creations with your target email address. This method should only be used on an initial account creation screen, and never submitted.

Email Provider

If your target's email address ends in gmail.com or yahoo.com, the identity of the email provider is quite obvious. However, business addresses and those with custom domain names do not notify you of the service that hosts the email. A domain's email provider is the company listed in the domain's MX record. The email provider may be the same as the domain's hosting company, but could also be a separate company. You may need to know the email provider in order to issue a court order for content or subscriber data. You may simply want to document this information within your investigation for potential future use. Regardless of your needs, the following will obtain the email provider from almost any address.

Navigate to **MX Toolbox** (mxtoolbox.com) and enter the domain of the email address, such as phonelosers.org. The result should include a hostname and IP address. These identify the email provider for the target domain. In this example, the host is mx1.mailchannels.net. This tells me that Mail Channels is likely the email host. This technique helps me identify the email providers or hosts behind business email accounts. If I am trying to connect an individual to a shell company, this may associate the same small provider with each target. I believe every thorough OSINT report should include a brief mention about the domain email provider. This should be checked as the investigation continues. Changing providers could be a sign of paranoia or intent to conceal evidence. Law enforcement can use this information in order to secure a proper search warrant.

IntelTechniques Email Addresses Tool

Similar to the previous IntelTechniques search tools, I have created a custom email address search tool as previously explained in Section I. This is my most used tool, and the option which requires the most updates. As you find new or updated resources, you will likely want to keep this tool functioning 100 percent. Figure 15.03 displays the online version available at the time of this writing.

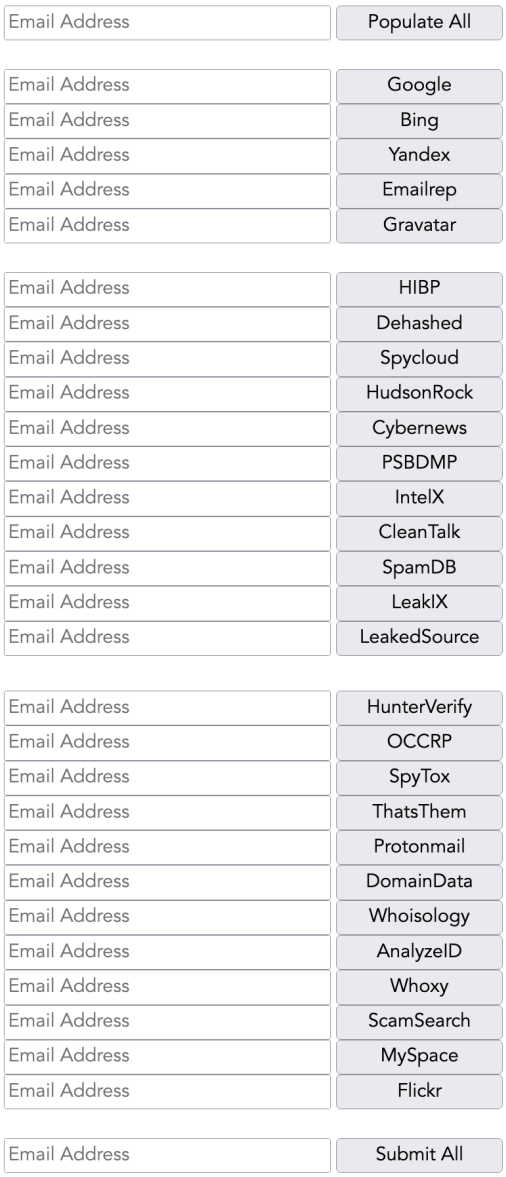


Figure 15.03: The IntelTechniques Email Addresses Tool.

CHAPTER SIXTEEN
USERNAMES

Once you have identified a username for an online service, this information may lead to much more data. Active internet users often use the same username across many sites. For example, the user "amanda62002" on Myspace may be the same "amanda62002" on Twitter and an unknown number of other sites. When you identify an email address, you may now have the username of the target. If a subject uses mpulido007@gmail.com as an email address, there is a good chance that he or she may use mpulido007 as a screen name on a number of sites. If the target has been an internet user for several years, this Gmail account was probably not the first email address used by the target. Searches for potential addresses of mpulido007@yahoo.com, mpulido007@hotmail.com, and mpulido007@aol.com may discover new information. Manual searching of this new username information is a good start. Keeping up with the hundreds of social websites available is impossible. Visiting the following services will allow you to search usernames across several websites, and will report links to profiles that you may have missed.

KnowEm (knowem.com)

KnowEm is one of the most comprehensive search websites for usernames. The main page provides a single search field which will immediately check for the presence of the supplied username on the most popular social network sites. A search for the username "inteltechniques" provides information about the availability of that username on the top 25 networks. If the network name is slightly transparent and the word "available" is stricken, that means there is a subject with a profile on that website using the supplied username. When the website is not transparent and the word "available" is orange and underlined, there is not a user profile on that site with the supplied username. For an online researcher, these "unavailable" indications suggest a visit to the site to locate that user's profile. The results could indicate that the target username is being used on Delicious and Twitter, but not Flickr or Tumblr. The link in the lower left corner of the result will open a new page that will search over 500 social networks for the presence of the supplied username. These searches are completed by category, and the "blogging" category is searched automatically. Scrolling down this page will present 14 additional categories with a button next to each category title stating "check this category". This search can take some time. In a scenario involving a unique username, the search is well worth the time. Fortunately for us, and our tools, the following direct URLs can be used to save the steps for conducting a basic and advanced search.

https://knowem.com/checkusernames.php?u=inteltechniques
https://knowem.com/checksocialnames.php?u=inteltechniques

CheckUserNames (checkusernames.com)

This site searches approximately 1/3 of the sites on KnowEm, but it links directly to target profiles.

Name Checkr (namecheckr.com)

This service appeared in late 2014 and conducts the same type of search as the previous competitors. The only slight advantage here is that the search is conducted faster than other sites. Additionally, you have a live hyperlink that will navigate to any identified accounts of the target.

User Search (usersearch.org)

This service stands out a bit from the others in that it only provides actual profile results. It searches the supplied username for a presence on over 100 of the most popular websites and returns a list of identified profiles matching the target. While this service is the slowest of all options, this could be an indication of account verification for more accurate results. My frustration with this site is the requirements to conduct a full search.

You must re-enter the username several times and process through the various query submissions. Some pages do not present all options, so you must often return to the homepage to start over. Fortunately, we can submit our queries directly via URL as follows, which allows us to add these to our custom search tools presented at the end of the chapter.

https://usersearch.org/results_normal.php?URL_username=inteltechniques
https://usersearch.org/results_advanced.php?URL_username=inteltechniques
https://usersearch.org/results_advanced1.php?URL_username=inteltechniques
https://usersearch.org/results_advanced2.php?URL_username=inteltechniques
https://usersearch.org/results_advanced4.php?URL_username=inteltechniques
https://usersearch.org/results_advanced5.php?URL_username=inteltechniques
https://usersearch.org/results_advanced6.php?URL_username=inteltechniques
https://usersearch.org/results_advanced7.php?URL_username=inteltechniques
https://usersearch.org/results_dating.php?URL_username=inteltechniques
https://usersearch.org/results_forums.php?URL_username=inteltechniques
https://usersearch.org/results_crypto.php?URL_username=inteltechniques

NameVine (namevine.com)

This username search service provides a unique feature missing in the rest. It allows you to begin typing any partial username and it will immediately identify registered accounts within the top ten social networks. This could be beneficial when you are not sure of the exact name that your target is using. If your target has a Twitter username of "Bazzell", the previous services will easily identify additional accounts that also possess this name. If you think that your target may be adding a number at the end of the username, it could take some time to search all of the possibilities. With NameVine, you can quickly change the number at the end of the username and get immediate results. It will search Twitter, Facebook, Pinterest, YouTube, Instagram, Tumblr, Wordpress, Blogger, and Github. It will also check for any ".com" domains that match. The benefit of this service is the speed of multiple searches. URL submission is as follows.

https://namevine.com/#/inteltechniques

Social Searcher (social-searcher.com)

This option provides a unique response. It is not a traditional username search site which displays accounts OWNED by the target. Instead, it queries for social network posts which MENTION the target. While searching my own username, I did not locate any profiles associated with me. Instead, I received numerous LinkedIn posts and profiles which mentioned my username. The static URL is as follows.

https://www.social-searcher.com/search-users/?ntw=&q6=inteltechniques

Whats My Name (whatsmyname.app)

This resource appeared in 2020 and replicates many of the features previously presented. However, it is always good to have redundant options and this service provides a unique feature. You can export your results to clipboard, XLSX, CSV, or PDF. A search of my own username revealed the following for easy documentation.

GitHub	coding	https://api.github.com/users/inteltechniques
Gravatar	images	http://en.gravatar.com/profiles/inteltechniques.json
InternetArchive	misc	https://archive.org/search.php?query=inteltechniques
Medium	news	https://medium.com/@inteltechniques
Reddit	news	https://www.reddit.com/user/inteltechniques
Spotify	music	https://open.spotify.com/user/inteltechniques
Twitter	social	https://shadowban.eu/.api/inteltechniques

Manual Query

You may desire a quick peek into the profile associated with a username across the most common networks. I added the following options to the Username Tool, assuming "inteltechniques" is the target username.

https://twitter.com/inteltechniques
https://facebook.com/inteltechniques
https://instagram.com/inteltechniques
https://www.tiktok.com/@inteltechniques
https://tinder.com/@inteltechniques
https://inteltechniques.tumblr.com
https://www.snapchat.com/s/inteltechniques
https://medium.com/@inteltechniques
https://youtube.com/inteltechniques
https://www.reddit.com/user/inteltechniques

Compromised Accounts

In the previous chapter, I explained how I use Have I Been Pwned (haveibeenpwned.com) and Dehashed (dehashed.com) in order to identify online accounts associated with a target email address. While HIBP does not always work with usernames, Dehashed performs well. Similar to the previous instruction, enter your target's username with the intent to discover any data breaches which include these details. The custom search tool contains multiple breach queries, as I find that information to be the most valuable at the beginning of an investigation. If my target username is "inteltechniques", I find the following direct query URL to be most beneficial for username search. This should look familiar to the options presented in the previous chapter.

https://dehashed.com/search?query="inteltechniques"

While we cannot submit queries via direct URL through all data breach sites, the following allows manual entry of usernames. All of these were explained in the previous chapter.

https://haveibeenpwned.com
https://leakpeek.com/
https://breachdirectory.org
https://psbdmp.ws

Gravatar (gravatar.com)

This service was explained in the previous chapter in regard to email addresses, but can also apply to usernames. The following URL would search my own information and display any email account images associated with my username.

https://en.gravatar.com/inteltechniques

Link Tree (linktr.ee)

This popular link aggregation service allows members to announce all of their networks on one page. We can also take advantage of this via direct URL, such as the following.

https://linktr.ee/ambermac

Email Assumptions

We can also make assumptions in order to identify target accounts. Assume that your suspect is IntelTechniques on Twitter. A search of the email addresses of IntelTechniques@gmail.com, IntelTechniques@hotmail.com, and IntelTechniques@yahoo.com at HIBP or Dehashed might reveal an active account that appears within a compromised database. Unfortunately, this manual search is time consuming and takes a lot of redundant effort. Therefore, consider using this option within the custom search tools.

These tools contain two search fields near the bottom of the page. Each queries a provided username through HIBP and Dehashed. The code appends the most popular email domains after the supplied username, and conducts a search within a new tab for each. If you provide IntelTechniques as the target username, the following email addresses are queried within independent browser tabs.

IntelTechniques@gmail.com
IntelTechniques@yahoo.com
IntelTechniques@hotmail.com
IntelTechniques@protonmail.com
IntelTechniques@live.com
IntelTechniques@icloud.com
IntelTechniques@yandex.com
IntelTechniques@gmx.com
IntelTechniques@mail.com
IntelTechniques@mac.com
IntelTechniques@me.com

Any positive results indicate that an address exists matching the search criteria, and that address was present within a breach. If you look at the code behind the tools, it appears as follows.

```
setTimeout(function(){window.open('https://dehashed.com/search?query=%22' + all3 + '@gmail.com%22',
'1leak3window' );},1000);
```

The Timeout function places a short pause in between searches. This ensures that we do not upset the server by conducting too many automated queries at once. If desired, you could replace "gmail.com", or any other options, with an email provider more appropriate for your investigations. If you are outside the U.S., you may want to replace several of these. You could also add more at the end. If I were to add "qq.com" after the last option, it would appear as follows. The 35000 is the next Timeout option to ensure that we have 3000 milliseconds in between each search execution.

```
setTimeout(function(){window.open('https://dehashed.com/search?query=%22' + all3 + '@qq.com%22',
'13leak3window');},35000);
```

Similar to how compromised database searches are the most powerful email search option that I use, querying usernames in this manner can be equally important. This search tool eliminates any laborious process and removes any excuses not to conduct this type of search every time.

The previous option made assumptions about usernames within email addresses which may appear on breach search sites. We can replicate that method within a standard search engine. The following URL populates a search on Google of the target username with the addition of the most popular email address domains, including quotation marks and the OR operator, as previously discussed. If your target username is IntelTechniques, you could manually conduct the following Google search.

"IntelTechniques@gmail.com"OR"IntelTechniques@yahoo.com"OR"IntelTechniques@hotmail.com"OR"IntelTechniques@protonmail.com"OR"IntelTechniques@live.com"OR"IntelTechniques@icloud.com"OR"IntelTechniques@yandex.com"OR"IntelTechniques@gmx.com"OR"IntelTechniques@mail.com"OR"IntelTechniques@mac.com"OR"IntelTechniques@me.com"

If desired, you could copy this query and paste it into Bing and Yandex, but I have found the results to be very unreliable. Your custom search tool makes this much easier. Enter your username in the "Email Search" option at the end in order to replicate this search. It will usually generate the most results the fastest.

Skype Username (web.skype.com)

Identifying a Skype username can be an important lead. It could direct you toward additional searches of the newly found data. Unfortunately, usernames are not obviously available when researching a real name on Skype. However, a quick method will reveal any Skype username when searching by name or email address. While logged in to a Skype account within the website or application, navigate to the search area. This section allows you to enter a real name or email address, and then conducts a search of the Skype user directory. Any results will appear immediately below. Clicking on these results displays the user's basic profile details including a photo. If the user chose not to include a photo, a silhouette graphic appears. Right-click on either image format and choose to "Open image in a new tab" (Chrome) or "View image" (Firefox). The new tab will contain the image that was already available. However, the address in the URL will reveal the Skype username of the target.

University Homepages

These automated searches for usernames can be very productive. However, they will not locate accounts within all online communities. One large untapped resource is the massive presence of university personal web pages and usernames. Most universities issue each student a university email address. These usually follow a standard naming convention such as the following.

lastname.firstname@university.edu

If you can identify the convention that the school uses and know the full name of the target, you can often determine the email address of the student. This address can be searched in order to identify any social networks that may have been missed. Furthermore, the first part of that address is usually the username that would have been issued to the student for a homepage. The target may have never used this email address online and a search result may appear empty. That does not mean there is not data available. The chance that the target created some type of homepage while attending the university is high. Finding the content is easy.

Hopefully, the searches explained earlier have helped in identifying a university that the target attended. A search for the university's website will reveal the domain name that the university uses. For example, Southern Illinois University at Edwardsville's website is siue.edu. We can now take that information and conduct a specific search for any personal pages on their domain. The search should similar to **site:siue.edu laura**.

I picked the name of Laura at random just to identify any student or employee personal website on the SIUE domain. One of the results was a link to a personal website belonging to "Laura Swanson". The link was similar to **www.siue.edu/~lswanso/**.

This indicates that the naming convention for personal websites is a tilde (~) followed by the first initial and then the first six letters of the last name. If the target of interest was "Scott Golike", the personal website would probably be at **www.siue.edu/~sgolike/.**

We can also assume the possibility of his school issued email account to be sgolike@siue.edu. A few searches using previously discussed techniques should confirm if this address belongs to the target. A search using the email to Facebook profile technique may identify an abandoned profile. We can now navigate to this personal

school page and see if there is any content. If there is, we can collect the data and conduct an analysis for intelligence and further research leads. If there is no page at this address, it does not mean that there has never been data there. This only indicates that there is no current content on this website. When students graduate, universities will usually remove all of the personal content from the servers. As discussed previously, this is never an excuse to stop looking. You can now take the URL of a target and conduct a search on The Wayback Machine (wayback.archive.org).

As an example, I can navigate to the first personal link for "Laura Swanson". Figure 16.01 (first) displays a portion of the live page at www.siue.edu/~lswanso/. If this page did not exist and the site contained no content, you could check on The Wayback Machine. Figure 16.01 (second) shows the search results for this personal page and identifies numerous archives dating back to 1997 for this site. Checking all of these options presents the many different versions of the site including one from 2005, as seen in Figure 16.01 (third), and the first capture from 1997, as seen in Figure 16.01 (fourth). This presents new data that would not have been uncovered with conventional searches. When a personal website is located, earlier versions should be archived.

Real World Application: While assisting another agency, information had developed in regard to a suspect in a priority investigation. After all online search attempts revealed nothing of value in locating the subject, a deleted student personal page was located using this method. It contained a list of friends, roommates, family members, and interests that were not previously known. This information helped locate the individual within hours.

It should be noted that some institutions will not follow a standard naming convention for all students and faculty. Additionally, there will be occasions when two or more students will have a name similar enough to create the same username. Usually, there is a plan in place to thwart these duplications. Sometimes it is as simple as adding the number "2" after the username. Universities are not the only places that create personal web pages based on a member name. Several internet service providers allow each subscriber a personal space online as part of the provider's main website. Comcast provides 25 MB of storage in a folder with the title of the subscriber's username. For example, if the email address of the customer was crazycheetah70@comcast.net, the username for the service would be crazycheetah70. The URL to view the personal web page would be as follows.

home.comcast.net/crazycheetah70

The following is a sample list of personal web page addresses from additional internet providers, using "crazycheetah70" as a username example. You should also search for internet providers in the target's area and attempt to find deleted pages on The Wayback Machine, Google Cache, Bing Cache, Yandex Cache, and the other services discussed in Chapter Nine.

360.yahoo.com/crazycheetah70
crazycheetah70.webs.com
crazycheetah70.weebly.com
webpages.charter.net/crazycheetah70
sites.google.com/ crazycheetah70
about.me/ crazycheetah70
angelfire.com/ crazycheetah70
geocities.com/ crazycheetah70
reocities.com/crazycheetah70
crazycheetah70.tripod.com
home.earthlink.net/~crazycheetah70
home.comcast.net/~crazycheetah70

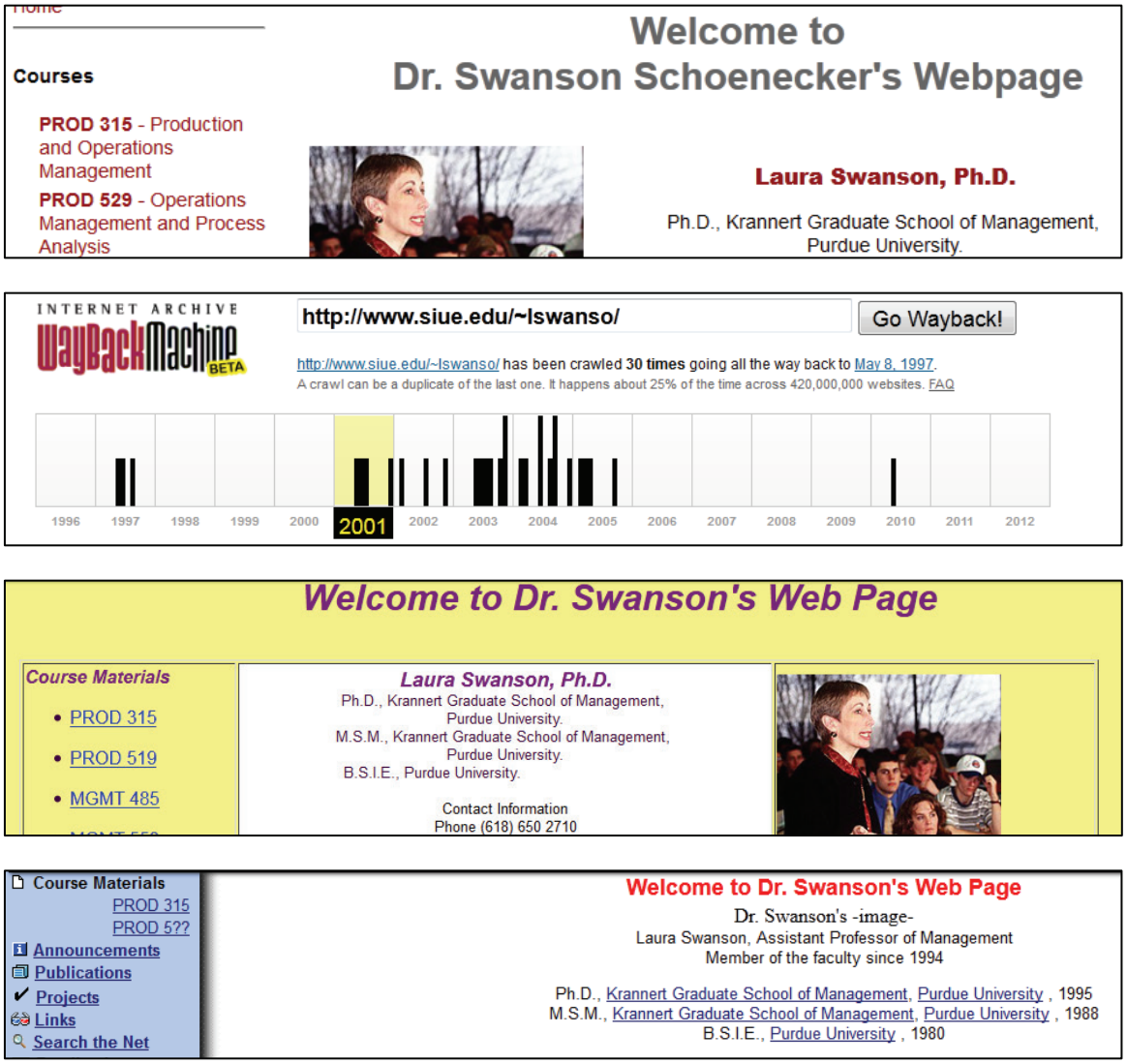


Figure 16.01: Historic versions of university pages.

IntelTechniques Usernames Tool

Similar to the custom email search tool, this page assists with an automated search of some of the techniques mentioned previously. This page allows you to execute manual queries or use the option to attempt all search possibilities. As a reminder, you must allow pop-ups for these pages if you want to use the "Submit All". An online version exists at **https://inteltechniques.com/tools/Username.html**.

CHAPTER SEVENTEEN
PEOPLE SEARCH ENGINES

Just as Google and Bing specialize in searching content on the internet, people search engines specialize in finding content about a particular person. Many of these sites utilize search engines such as Google and Bing to help compile the data, and then present a summary style interface that is easy to consume. The sites listed here each have their own strengths and weaknesses. Standard searches are free on all of them; however, each site generates revenue in some form. Usually, this is by displaying advertisements that often appear to be a report function within the site. I do not recommend purchasing any of the premium paid services until all free options have been exhausted.

These details are often focused on targets in the United States, but many services are starting to reach past North America. Searching a target's real name often leads to the discovery of home addresses, telephone numbers, email accounts, and usernames. The following resources are presented in order of most beneficial within my investigations to least. Please note that many of these sites are now blocking access from IP addresses associated with VPNs. This is due to malicious "scraping" behavior by competing sites. If you receive an "access denied" error, it is likely your VPN is being blocked by the service.

True People Search (truepeoplesearch.com)

This service has provided the most comprehensive and consistent free report based on real name. The results usually include current address, previous addresses, telephone numbers (including cellular), email addresses, relatives, and associates. There are no blatant ads, but the "Background Reports" links forward you to a paid service which will likely return the same details. This is my first stop when trying to locate an individual in the U.S. The direct search URL is as follows.

https://www.truepeoplesearch.com/results?name=michael%20bazzell

Fast People Search (fastpeoplesearch.com)

This service appears to rely on the exact same database as True People Search. However, it possesses one huge advantage. True People Search gained a lot of media attention as a "stalking" assistant, which resulted in numerous articles instructing individuals to remove their profiles with an opt-out link. Many people have removed their information, but practically none of them replicated the procedure on Fast People Search. If your target removed his or her profile from the previous option, it might still be present here. The direct search URL is as follows.

https://www.fastpeoplesearch.com/name/michael-bazzell

Nuwber (nuwber.com)

A newcomer in 2017 was Nuwber. I first learned about this service from various members of the privacy forum at my website. They were discussing the importance of removing their own personal details from this site through various opt-out procedures available at the time. I found my own information to be quite accurate. Therefore, this makes for a great OSINT resource. The default landing page allows search of a first and last name. The results are presented by location, and each profile often includes full name, age range, home address, telephone number, and neighbors. The direct search URL is as follows.

https://nuwber.com/search?name=michael%20bazzell

XLEK (xlek.com)

This was another service that surprised many privacy-conscious people. A typical entry contains a target's full name, current home address, current home telephone number, email addresses, previous home addresses, additional telephone numbers, possible relatives, and age. I recently located a target's email address from this service, which was not present on any other site. Using the techniques mentioned in previous chapters, I was able to create a full dossier. The default page does not present a direct URL, but the inspector presents following.

https://xlek.com/search_results.php?fname=michael&lname=bazzell&locations:all

Family Tree Now (familytreenow.com)

In 2016, this website emerged and launched an uproar online. This site is targeted toward those who want to conduct family history research, and its specialty is connecting a person to his or her relatives. The results do not display home addresses, but simply the age of the target and a large list of family members. After gaining a lot of online popularity, many people started complaining about the availability of this sensitive information. While this type of service is nothing new, people were outraged at this violation of their privacy. Since Family Tree Now sources all of their data from public databases, they defended their product, which is still available today. This attention may have been the inspiration for this company to take things to another level with True People Search. The direct search URL is as follows.

https://www.familytreenow.com/search/genealogy/results?first=Michael&last=Bazzell

Cyber Background Check (cyberbackgroundchecks.com)

I suspect this site is associated to the first two entries in this chapter. However, I have located unique data. Therefore, it should be included in our tools. The direct search URL follows.

https://www.cyberbackgroundchecks.com/people/michael-bazzell

Intelius (intelius.com)

Intelius is a premium service that provides reports about people for a fee. Most of the information is from public sources, but some of it appears to come from private databases. Searching for any information on the main website will always link you to a menu of pricing options. The information will never be displayed for free. However, the page that lists the report options does possess some interesting information. This free preview identifies an exact age, possible aliases, cities lived in, previous employers, universities attended, and relatives. If the subject is married, this will usually identify the spouse. In most situations, it will identify the maiden name of the person's wife. Anything that you do not see on this main screen, you must pay a fee. I never recommend purchasing any of this data. Users are usually disappointed with the results. The direct search URL is as follows.

https://www.intelius.com/people-search/Michael-Bazzell

Radaris (radaris.com)

This service has many similarities to Intelius. However, the data set is unique. The business model is to entice you into purchasing an entire profile. I only use the service for the limited free content available in the preview. After searching a name, select the most appropriate target and choose "Full Profile" in the lower right of the result. This will open the full view of any free information. This will often include the target's middle name, age, current address, previous address, landline telephone number, and links to social networks. The Background Check options forward you to a premium website which I do not recommend. The direct URL follows.

https://radaris.com/p/Michael/Bazzell/

Spytox (spytox.com)

This service provides limited data during a free search, but the records are often updated from premium resources. In my experience, the free option only provides city, state, and phone. The direct URL follows.

https://www.spytox.com/michael-bazzell

Search People Free (searchpeoplefree.com)

This database looks awfully familiar to True People Search and Fast People Search, but I occasionally locate an individual present here and nowhere else. One benefit here is that the email addresses identified within the results are hyperlinks. They connect to additional names on occasion. The direct search URL is as follows.

https://www.searchpeoplefree.com/find/michael-bazzell

That's Them (thatsthem.com)

In late 2014, a new website quietly entered the crowded scene of people search services. On the surface, it was just another service that aggregated publicly available information. Consequently, a closer examination revealed That's Them to contain information that is not available anywhere else for free. This service has many options, and most will be discussed in this book. For the purposes of this chapter, I will focus on the "Name and Address" search option in the top menu of the website. Entering a full name with city and state is preferred, but not required. Results often display the person's age range, cell phone number, landline number, full address, religion, financial details, home IP address and any associated email addresses. I searched my own name to test the accuracy of the results. My profile correctly identified similar information as well as the exact VIN number of a previous vehicle. This type of data is impressive without any fees. I have found their details of religion and financial information to be unreliable. Note that the options to purchase additional information are advertisements from third-party companies, and should be avoided. The direct search URL is as follows.

https://thatsthem.com/name/Michael-Bazzell

Spokeo (spokeo.com)

Spokeo is probably the most well-known of all the people search engines. There are two very distinct versions of this service, free and premium. The premium service will provide a large amount of accurate data, but at a cost. The free version provides an interface that is easy to navigate. The results from a target name search will be presented after choosing a state and city. Only the states and cities where the target name has a presence will be shown. Choosing this target will display a profile with various information. Within this data will be several attempts to encourage you to purchase a premium account. Basically, anything that you do not see within this profile will cost you money. Any links from the profile will present a membership plan with pricing. The profile will often display full name, gender, age, and previous cities and states of residency. However, it no longer presents the actual current address. Spokeo is one of many sites which present an animate gif file while you wait on the results. This is unnecessary and is in place to make you believe data is being queried in real time. The following static search URL bypasses this annoyance.

https://www.spokeo.com/Michael-Bazzell?loaded=1

Advanced Background Checks (advancedbackgroundchecks.com)

This is another service which includes advertisements for premium options. It appears to present data very similar to Intelius. Surprisingly, the majority of their data archive is free without any payment. The main search results appear redacted and entire addresses and telephone numbers are masked. With many other services, clicking these details prompt the user for payment. Instead, this service opens a new page revealing the entire

record. This often includes home address, home landline telephone number, age, and relatives. Clicking the "See full info" button reveals previous addresses, additional telephone numbers, and aliases. Overall, this service is extremely useful for U.S. targets. The direct search URL is as follows.

https://www.advancedbackgroundchecks.com/names/michael-bazzell

Yasni (yasni.com)

On the surface, Yasni appears to be another standard people search engine. Much of the content received will be duplicate data, but there are a few areas where Yasni works differently. The home page will give three search options. For most OSINT purposes, the last option is the desired search. It will accept a real name or username and forward you to a results page. Real name search will present a large number of links associated with your target's name. As with other engines, many of these results will be about a person other than your target. The first box on the results page will include a "lives/works in" option that will display the cities of the users identified with the search. Clicking on a location that looks appropriate for your target will load a new results page that will provide all search results about your specific target. Yasni will identify news articles, websites, and social networks related to your target. By default, the search is conducted internationally. Yasni is a German site and searching outside of the United States is one of the strengths of the service. The search bar includes an option to filter the results by specific countries, but the United States is not listed as an option. If you have a target that lives in another country, Yasni is a great tool. The direct search URL is as follows.

http://www.yasni.com/michael+bazzell/check+people?sh

Zaba Search (zabasearch.com)

This site appears to have several search options at first glance. Unfortunately, all but one will forward to an Intelius site, which will require a fee. There is one very specific free option on this page. Providing any real name and state of residence will provide a results page with full name, date of birth, address, and phone number. In my experience, this often includes unlisted telephone numbers and addresses. Clicking on practically anything else on this results page will take you to a sponsored link. When I use this resource, I only rely on the information obtained on the first result page. The direct search URL is as follows.

https://www.zabasearch.com/people/michael+bazzell/

People Search Now (peoplesearchnow.com)

This database appears to have the same parent company as True People Search, and possesses the same data. However, this search should be included in the event that a target has removed details from other related websites. Due to increased awareness of exposed personal information, I am seeing many people request removal of personal online data. The direct URL is as follows.

https://www.peoplesearchnow.com/person/michael-bazzell

WebMii (webmii.com)

This service emphasizes information associated with social networks. I have never located any home addresses or telephone numbers, but I have found online images that were not available on any other search engine. This is not the most productive option, but one to consider when desperate for details. The direct search URL is as follows.

http://webmii.com/people?n=%22Michael%20Bazzell%22

Social Searcher (social-searcher.com)

I explained this service within the username chapter, but it also applies here. It focuses on social media presence, including mentions within posts. The direct search URL is as follows.

https://www.social-searcher.com/search-users/?q6=Michael+Bazzell

Truth Finder (truthfinder.com)

There is nothing very special about this service, as it will likely have data similar to the other sites already mentioned. However, there is one major annoyance. When you search this site, you are bombarded by fake progress meters that insinuate that a huge report is being prepared about your target. On average, a real name search takes over 14 minutes due to these constant "please be patient while we find more details" notifications. The solution to this is to submit a direct URL as follows.

https://www.truthfinder.com/results/?firstName=Michael&lastName=Bazzell&state=ALL

People By Name (peoplebyname.com)

The title of this site is misleading. It only allows the query of a telephone number, which is not beneficial when searching by name. However, the following URL structure displays all profiles associated with the provided name.

http://www.peoplebyname.com/people/Bazzell/Michael

White Pages (whitepages.com)

This is the official White Pages website that will conduct a reverse name or address search. The results often include known residents and neighbors. This data is pulled from public information and is rarely complete. The direct search URL is as follows.

https://www.whitepages.com/name/Michael-Bazzell

Clustr Maps (clustrmaps.com)

There is nothing too exciting here, as it is more of the same. Since this service offers a direct URL query, we should add it to our arsenal. The URL structure follows.

https://clustrmaps.com/persons/Michael-Bazzell

Find People Search (findpeoplesearch.com)

This service possesses a database identical to many of the better options previously presented. It also appears that records are not being updated and the details are becoming quite stale. This can be a good thing. If you target has moved and established a new address publicly, Find People Search might be the only option which displays a previous address. There is no direct URL submission option, so a manual search is your only option.

Public Records (publicrecords.directory)

This service displays a few unique details about an individual. It taps into various voter registration databases, campaign contribution disclosures, and vehicle registration data. None of these categories appear consistently within search results, but a search here is always justified. There is no direct URL submission option, so a manual search is your only option.

Rocket Reach (rocketreach.co)

This impressive site does not identify home addresses. Instead, it focuses on business details likely scraped from LinkedIn and online resumes. Conducting a search on the website only presents a login form and an account is mandatory. However, Google can help us. I conducted the following search.

site:rocketreach.co "michael bazzell"

This resulted in three Rocket Reach profiles, all of which could be viewed without an active account. The first was someone else with my name. It disclosed the following details.

Location: Baltimore, Maryland, United States
Work: Senior Artist @ Firaxis Games, Senior Artist @ MicroProse Software, Artist @ Innerprise Software
Education: Maryland Institute College of Art, Bachelor Of Fine Arts (Communication, Illustration), 1985 - 1989
Michael Bazzell's Email: @firaxis.com @kryptotoyz.com @erols.com
Michael Bazzell's Phone Numbers: 410667XXXX 410726XXXX

The site then teases the ability to uncover the redacted results for free. However, this requires a free account.

Official USA (officialusa.com)

While many targets will not be present within this data set, those who are will be quite exposed. A typical result includes full name, home address, email address, phone number, and home sale records. The URL structure follows.

https://www.officialusa.com/names/michael-bazzell/

Addresses (addresses.com)

This service appears to be using other public datasets, but the redactions of details seem to be less than the other premium sites. It is common to see full telephone numbers, addresses, and family connections. The URL structure follows.

https://www.addresses.com/people/michael-bazzell

ID Crawl (idcrawl.com)

This service attempts to build a full online profile from a name. Results are hit or miss, but the following direct URL access makes for an easy query.

https://www.idcrawl.com/michael-bazzell

Classmates (classmates.com)

Classmates is a very underrated resource for the internet searcher. Unfortunately, you must create a free account to take advantage of the worthy information inside the site. This free account can contain fictitious information and it is necessary to complete a profile on the site to access the premium features. After you are logged in, you can search by first and last name. If you know the school that was attended, the results will be much more accurate. This should provide the school attended as well as the years the target attended the school. My new interest in this site is due to the availability of scanned yearbooks. The collection is far from complete, but there are a surprising number of complete yearbooks available to browse. Analyzing this content can be very time consuming, as the yearbook must be manually browsed one page at a time. The information obtained should be unique from any internet search previously conducted.

Rebate Tracking

I find this technique more "interesting" than valuable to an investigation, but you might locate a nugget of information which assists you. As an example, let's look at the store Menards. Menards is an American home improvement company which pushes rebates in order to make products appear low-priced. When you buy a product, you can retrieve a rebate slip and mail it off. In a few weeks, numerous checks in small amounts begin arriving at your home. This game provides small discounts, but at what risk? This is where the online rebate center comes in. Let's start with a visit to https://rebateinternational.com/RebateInternational/tracking.do. This page allows you to enter a first initial, last name, numeric portion of an address, and postal code of your target. This result appears similar to Figure 17.01.

Redeemed Date	Rebate #	Description	Purchase Date	Rebate Amount	Rebate Status	Check #
	668	DEFENSE ZONE SANITIZER	03/17/2021	$3.55	Mailed on 04/21/2021	XXXXXX5347
	658	SAVE 11%	03/17/2021	$1.09	Mailed on 04/21/2021	XXXXXX5347
	657	SAVE 11%	03/10/2021	$4.31	Mailed on 04/21/2021	XXXXXX5347
03/10/2021	633	DEFENSE ZONE HAND SANITIZ	12/28/2020	$2.99	Mailed on 03/05/2021	XXXXXX0196
12/28/2020	554	SAVE 11%	05/26/2020	$3.93	Mailed	XXXXXX5704

Figure 17.01: A Menards rebate tracking screen.

We now see the purchased items, purchase dates, rebate amounts, and payment details. This gives us a glimpse into the shopping habits of the target and could lead to patterns of behavior which might expose someone's routines. It could also provide pretext value. I could initiate an email phishing attack focused on a specific purchased item announcing a problem with the rebate. That would be so targeted that the recipient would probably click whatever link I sent. This rebate service even stores all purchases within a static URL which is prone to abuse. If my name were Michael Smith and I lived at 1212 Main Street in Houston, Texas, my URL would be the following.

https://rebateinternational.com/RebateInternational/trackResults/M/Smith/1212/77089

Note that this URL does not require any credentialing and is open to brute-force scraping. Next, let's take a look at Lowe's home improvement. They offer a rebate confirmation website at https://lowes-rebates.com/en-us/RebateStatus which allows entry of a cellular number or home address, as seen in Figure 17.02.

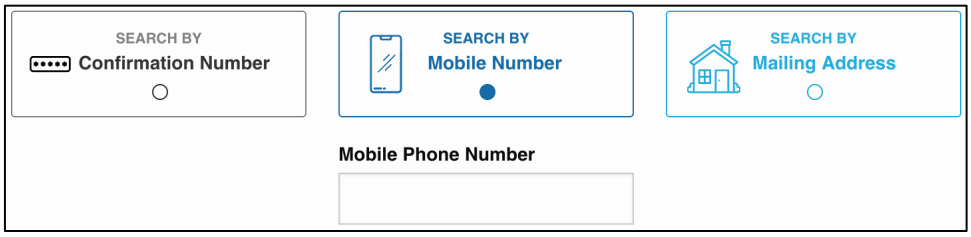


Figure 17.02: A Lowe's rebate tracking entry.

This allows us to potentially translate an unknown cell number into a name and address or vice-versa, as well as obtain shopping history of the target. These two examples are only a small selection of the possibilities. You will find online portals for Micro Center, Budweiser, Miller, Kohls, Auto Stores, and other retail establishments. I have found the following three Google searches to assist with finding companies which offer similar options.

https://www.google.com/search?q=intitle:"rebate center"
https://www.google.com/search?q="track+my+rebate"
https://www.google.com/search?q="track+your+rebate"

Utility Inquiries

If you know your target's telephone number and postal code, you may be able to uncover an entire home address. Consider the following example with the home telephone and internet service provider CenturyLink. The following address connects to their internet troubleshooting page.

https://www.centurylink.com/home/help/internet/test-your-internet-at-the-network-interface.html

Clicking "Open Troubleshooter" in the lower right area opens a query page asking for the telephone number and postal code. Submitting this data then presents a confirmation screen which includes the customer's name, full address, and telephone number. I have found many local utilities which offer similar search portals.

IntelTechniques Names Tool

The abundance of free person search tools can get overwhelming. They each have strengths and weaknesses, and none of them are consistent on accurate results. In years past, I would manually visit each site and enter my target information. This would usually result in small, yet valuable, pieces of information from each service. Today, I use a custom search tool that I created to search all possible sites simultaneously. This free tool is in your offline downloads and online at **https://inteltechniques.com/tools/Name.html**. You can either enter the first and last name of your target within the search fields of each service, or enter this information only once at the final set of search fields. This latter option will launch several new tabs and conduct your search across each service. During my live training courses, I am often questioned about the absence of the ability to enter a middle initial or name. While I could make this an option, I find that a lot of people search websites do not always possess a middle name. Entering this data could harm an investigation by omitting valuable results.

Resumes

Resume searching was mentioned earlier. Those methods will identify many documents, especially if the word "resume" is inside the file or file name. These results are only a portion of available content that could be extremely valuable to your investigation. I believe that resumes are an ideal target since they usually contain sensitive information that is not posted anywhere else. Many people will include a cellular number and personal email address on a resume, but would never consider placing these details on a social network. If the resume is publicly available, regardless of whether the target realizes this, we can gather good intelligence. The following techniques aim to assist in locating this valuable data. Detailed searches within Google or Bing will identify many resumes hosted publicly on websites and cloud-based document storage services. If my target's name is Michael Bazzell, I have found the following exact searches valuable on Google, Bing, and Yandex.

"Michael Bazzell" "Resume"
"Michael Bazzell" "Curriculum Vitae"
"Michael Bazzell" "CV"
"Michael Bazzell" "Resume" filetype:doc
"Michael Bazzell" "Curriculum Vitae" filetype:doc
"Michael Bazzell" "CV" filetype:doc
"Michael Bazzell" "Resume" filetype:pdf
"Michael Bazzell" "Curriculum Vitae" filetype:pdf
"Michael Bazzell" "CV" filetype:pdf
"Michael Bazzell" "Resume" site:docs.google.com
"Michael Bazzell" "Curriculum Vitae" site:docs.google.com
"Michael Bazzell" "CV" site:docs.google.com

While these queries will likely locate any resumes with text, they will fail on many resume images. Numerous resume hosting websites have realized that various data scraping engines scour their resume collection and "steal" their content. This has encouraged some services to store images of resumes that do not contain text that can be easily searched. While this is a decent layer of protection, it is not enough to keep out of Google results. Since Google scans images for Optical Character Recognition (OCR), it knows what words are within an image. After conducting the previous queries within traditional search engines, attempt them through Google Images (images.google.com). A search of "Mary Johnson" "Resume" on Google Images revealed hundreds of images of resumes. A manual inspection of each identified many pieces of sensitive information.

CV Maker (cvmkr.com)

This website allows users to create free professional resumes and CVs. Currently, over 5 million have been created and are stored within the service. The home page does not offer a search option, as this service is not intended to be used as a people finder. However, we can rely on a Google search to get us the content we want. The following identifies the resume of our target.

site:cvmkr.com "john pratt"

The search result opens a PDF. Within that file is the un-redacted content, which identifies his full email address, telephone number, and home address. On rare occasions, I have found this PDF option to be missing from my target profile. When this happens, we can create a direct link to the full details. In this example, our target's page is at cvmkr.com/7J0N. The following URL presents the entire PDF with the visible details. Basically, adding "?pdf=1" at the end of the URL should always present the full resume view. Since Google indexes all of the PDF files that are located, you can also perform searches for telephone numbers and email addresses using the site operator previously mentioned.

https://cvmkr.com/7J0N?pdf=1

Indeed (resumes.indeed.com)

Indeed has a powerful collection of resume data. Because the term "resume" is not present in any of the content pages, you will likely not obtain this data during your standard searches. Entering your target name on Indeed under the "Find Candidates" option may present new results. Contact information is usually redacted. However, detailed work experience, education, and location are commonly present.

Ripoff Report (ripoffreport.com)

If your target conducts any type of business with the public, he or she will likely upset someone at some point. If your target regularly provides bad service or intentionally commits fraud within the business, there are likely many upset victims. Ripoff Report is a user-submitted collection of complaints about businesses and individuals. I have had investigations into shady people and businesses where these reviews were beneficial.

Gift Registries

Decades ago, people were surprised at the gifts presented to them after a wedding or birth. Today, we create online registries identifying the exact products desired, and within moments someone can purchase and ship the "thoughtful" gift with very little effort. As an investigator, I have always enjoyed the plethora of personal details within these registries, which tend to stay online long after the related event. Before identifying the best resources, let's take a look at the types of details we can acquire from some random targets.

Partner Name: When I am investigating someone, that person usually knows they are under a microscope. He or she tends to stop posting to social media and starts scrubbing any online details. However, their partner tends to ignore the threat of investigation and continues to upload sensitive information applicable to the target.

Therefore, online wedding and baby registries help me identify the most lucrative target aside from the original suspect. In an example from the wedding registry website theknot.com, I received over 200 results for Michael Wilson, which also includes the name of the future spouse.

Maiden Name: In the example above, the results only identified future weddings. However, modifying the year in the search menu allows us to view past weddings. This will divulge a woman's maiden name. This can be beneficial in order to better locate a Facebook page or other family members that may be off my radar. I can also use this to then search old yearbooks, criminal details, and previous addresses.

Date / State: Many counties will only share marriage certificates if the requestor knows the exact names of each party and the exact date of the event. We have everything we need in order to file a request. Marriage certificates often include full details of all parents, witnesses, and the officiant. Furthermore, I now have their anniversary date which can be helpful during a phishing attack or social engineering attempt. You might be surprised at the number of people that use their anniversary as a security question to an online account.

Ceremony Details: The Knot and other wedding registry sites offer the couple a free website to announce details about the upcoming (or past) event. This usually includes an embellished story about how they met, fell in love, and he proposed. While this could be good knowledge for social engineering, I am usually more interested in the wedding party. This will usually include the closest friends of my target, which will be next on my investigation list.

Items: While it may be fun to look at the items desired by a couple, there is much we can learn about their lives based on these details. In an example from Figure 17.03, I now know that a random Michael Wilson, who is getting married in San Antonio, will be going to his honeymoon in Maui (#2), snorkeling (#3), at the airport carrying a Lowepro backpack (#4), checking red/black suitcases (#5), capturing everything on a Canon HD camcorder (#6), dining at the Lahaina Grill (#7), and staying at a fancy nearby hotel (#8).

Other recent examples associated with actual targets identify the types of phones used, vehicles driven, and subjects of interest. While The Knot requires both a first name and last name to conduct a search, providing two asterisks (**) as the first name will present every entry online including the provided last name.

Children: The items within a baby registry will usually provide little to no value. Knowing the brand of diapers preferred or favorite crib style has never helped me in the past. However, knowing a due date and location of the target can be beneficial for future searching. Unfortunately, The Bump only allows searching of upcoming births, and not any past profiles. Fortunately, Google has our backs. The following Google search revealed multiple baby registries from the past few years associated with Michael Wilson.

site:registry.thebump.com "Michael Wilson"

Gifts: The most fruitful registries in regard to identifying personal preferences of a target are the various gift registries. Of all these, Amazon is the most popular. The following are the most common wedding, baby, and gift registries, with direct links to the most appropriate search pages. I highly encourage you to conduct a detailed Google "Site" search after attempting the proper method.

The Knot: https://www.theknot.com/registry/couplesearch
The Bump: https://registry.thebump.com/babyregistrysearch
Amazon Baby: https://www.amazon.com/baby-reg/homepage/
Amazon Wedding: https://www.amazon.com/wedding/
Walmart Wedding: https://www.walmart.com/lists/find-wedding-registry
Walmart Baby: https://www.walmart.com/registry/baby/search
Target Wedding: https://www.target.com/gift-registry/
Target Baby: https://www.target.com/gift-registry/baby-registry
Kohl's Wedding: https://www.myregistry.com/kohls-wedding-registry.aspx

Ikea Wedding: https://info.ikea-usa.com/giftregistry
Bed Bath & Beyond Wedding: https://www.bedbathandbeyond.com/store/page/Registry
Macy's Wedding: https://www.macys.com/registry/wedding/registrysearch
Registry Finder: https://registryfinder.com
My Registry: https://www.myregistry.com

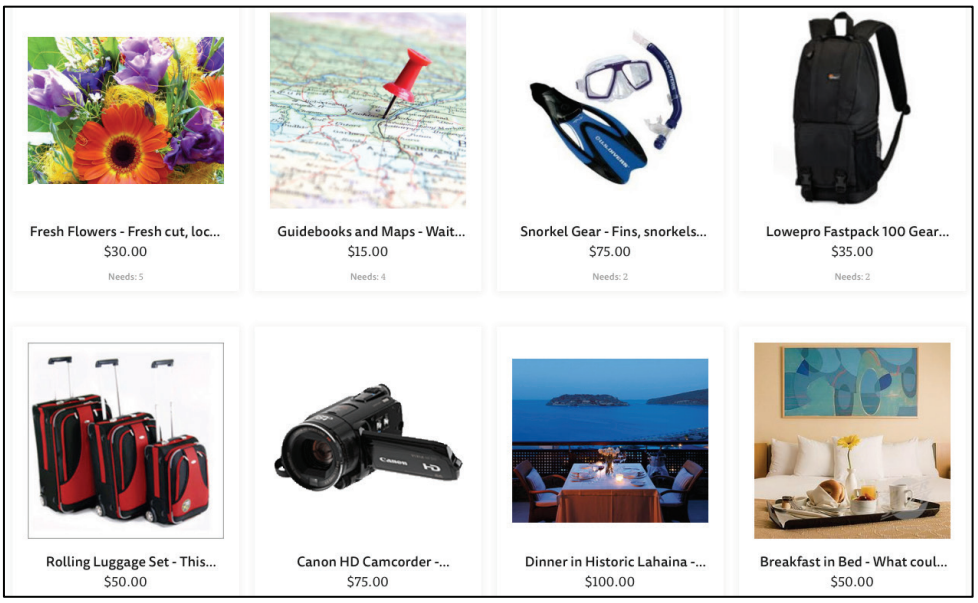


Figure 17.03: A search result from a gift registry website.

Find a Grave (findagrave.com)

While I assume that your goal is to find living targets, you should also have a resource for locating proof of deceased individuals. I have used this website numerous times to locate the graves of recently deceased people. While not necessarily "proof" of death, it provides a great lead toward locating a death certificate and living family members. In 2021, I began finding better results at **People Legacy** (peoplelegacy.com).

Addresses

The target of your investigation may be an address of your suspect. You may want to know who else lives at a residence. There are dozens of websites that possess databases of address information. I have outlined a few here that are unique from those already discussed. Additionally, the following websites which were previously discussed all allow reverse search of a residential address.

https://www.fastpeoplesearch.com/
https://www.whitepages.com/reverse-address
https://www.peoplefinder.com/
https://www.peoplesearchnow.com/
https://www.truepeoplesearch.com/
https://radaris.com
https://www.intelius.com/
https://www.advancedbackgroundchecks.com/address.aspx
https://www.spokeo.com/reverse-address-search
https://thatsthem.com/reverse-address-lookup
https://homemetry.com
https://cyberbackgroundchecks.com

Voter Registration (www.blackbookonline.info / USA-Voter-Records.aspx/ voterrecords.com)

Many people will have their address and telephone number unlisted in public telephone books. This prevents their information from appearing on some websites. If any of these people are registered voters, their address may still be public. In order to locate this data, you will need to connect to the county clerk of the county of residence. The link here will display a list of all fifty states. Clicking the state of the target will present all of the counties with known online databases of voter registration content. These will often display the full name of the voter and full address. This can be sorted by name or address depending on what information you have about the target. Later chapters present additional voter registration search options.

Unclaimed Property

Every state in the U.S. possesses an unclaimed property website. Common forms of unclaimed property include savings or checking accounts, stocks, uncashed dividends, online payments, refunds, and leftover utility balances. Most of these pages only require a last name to conduct a search. The results almost always include full name, home address, property type, and amount owed. This can provide a great search resource, and a possible pretext for a phone call. A complete collection of these sites with easy access to each state is at **Unclaimed** (unclaimed.org).

Google (google.com)

If all else fails, or you believe you are missing something, check Google. Searching the address should identify any leftover information about your target address. When searching, place the street address in quotes excluding the city. An example search may appear similar to "1234 Main" "Bethalto" IL. This will mandate that any results include the exact address and the exact city name, but they do not necessarily need to be right next to each other on the page. If you place the entire address including the city inside a single set of quotes, you would miss any hits that did not have this exact data. This search should reveal numerous home sale websites that may have unique interior photos.

Non-U.S. Targets

This chapter is heavily focused on U.S. targets. If you have a subject of interest outside of America, I have found the following to be beneficial.

Australia: www.peoplesearch.com.au
Canada: www.canadapages.com
Canada: www.infobel.com/en/Canada
France: www.infobel.com/en/France
Germany: www.infobel.com/en/Germany
Spain: www.infobel.com/en/Spain
UK: www.192.com
UK: www.peopletraceuk.com
UK: www.gov.uk/government

CHAPTER EIGHTEEN
TELEPHONE NUMBERS

There are hundreds of websites that claim the ability to search for information on telephone numbers and addresses. These vary from amazingly accurate results to sites that only include advertisements. If I have a target telephone number, there are three phases of my search. First, I want to identify the type of number and provider. The type could be landline, cellular or internet, and the provider could be the company supplying the service. Next, I want to identify any subscriber information such as the name and address associated with the account. Finally, I want to locate any online web content with a connection to the target number. This can all lead to more intelligence and additional searches. The majority of cellular numbers can now be identified if they are registered in someone's name. If you have an address, you will want to identify any associated people or telephone numbers. This chapter will highlight the sites which can assist with these tasks.

Carrier Identification

Ten years ago, I often queried telephone number porting websites to identify the provider of my target's telephone number. This would identify the cellular company that supplied service to my suspect. I would use that information within my court order to demand subscriber data about the target. Knowing the provider was essential as to not waste time requesting records from companies that had no data to provide. The websites used back then have either disappeared, or now charge a substantial fee for access. Five years ago, I noticed that an overwhelming amount of my target telephone numbers were connected to Voice Over Internet Protocol (VOIP) services such as Google Voice and Twilio. This would often be indicated by a result of "Broadband" or "Internet" instead of something obvious such as "Verizon". Until recently, the absence of a specific provider was a hurdle during investigations. Today, we have more sophisticated services that can identify exact provider details on practically any number.

Free Carrier Lookup (freecarrierlookup.com)

I begin with this option because it has been the most stable and does not limit searching as others do. This site requests any domestic or international telephone number and produces a report which includes the country, type of service, and provider associated with the number. While many online services can identify the provider of a cellular or landline number, this option provides the best details about VOIP numbers. I submitted several telephone numbers associated with various providers that I could personally confirm. The following identifies the results of these searches. The first column represents the provider as I knew it, the second column is the type, and the third is the provider displayed by Free Carrier Lookup. This is far from complete, but I wanted to demonstrate the ability to convert internet-based numbers into identifiable companies.

Verizon	Mobile	Verizon Wireless
AT&T	Mobile	AT&T Wireless
Google Voice	VOIP	Google/Level 3
Sudo	VOIP	Twilio/Level 3 (SMS-Sybase) (MMS-SVR)
Blur	Landline	Twilio/Level 3 Communications
TextNow	VOIP	Enflick/Bandwidth.com (SVR)
On/Off	VOIP	Peerless Network

If this service cannot provide the data you need, or if you want another source to provide more confidence in the result, you should also consider the sites **Carrier Lookup** (carrierlookup.com) and **Phone Carrier Check** (phonecarriercheck.com). If both of these services disappear by the time you need them, a Google search of "carrier lookup" should provide new alternatives.

Scout (scout.tel/phone-number-lookup)

This newer service provides a few pieces of information about a telephone number which I have not seen available anywhere else. The following is a partial result of a target telephone number query.

administrative_area_level_3	Alton Township
administrative_area_level_2	Madison County
administrative_area_level_1	Illinois
operating_company_name	AT&T
line_type	mobile
ported	false
risk_rating	unlikely
risk_level	28

We now know that this number was not ported from another carrier; it is a mobile number; and it possesses a low risk rating. These ratings increase for VOIP, forwarding, and masked numbers, and might indicate that the number you are searching is not assigned to any specific individual.

Caller ID Databases

Once you have identified the provider, you should focus on the subscriber data associated with the number. We will tackle this in several different ways, as many things changed in 2019. First, we should focus on the most reliable options, which often require registration for a free trial. This chapter becomes quite technical quickly, but the tools at the end will simplify everything.

In 2013, I began experimenting with reverse caller ID data. These are the same databases that identify a telephone number on your landline caller ID display. Often, this will include the name associated with the number. Until recently, this was something that only appeared on landline numbers, but that has changed. Now many cellular telephone numbers have name information associated with them. This name information is usually extracted from the cellular telephone service provider. I was immediately shocked at the accuracy of these results while searching cellular telephone numbers that were otherwise untraceable. On many of my investigations, this technique has eliminated the need to obtain court subpoenas to discover subscriber information.

The reason we can access this data is because it is necessary for telephone systems that do not already possess caller ID options. New systems that operate over the internet, referred to as VOIP systems (Voice Over Internet Protocol), do not receive caller ID data natively. This is something that we have taken for granted while it was provided by our telephone companies. Today, many businesses must purchase access to this data from resellers. This presents us with an opportunity.

I scoured the internet for every business that provides bulk caller ID data to private companies. Some offer a free website for testing; some require you to submit queries through their servers; and others make you register for a free trial. I have tested all of them and identified those that are easy to access and give the best results. First, I will focus only on easy and reliable ways to search an individual number through specific web addresses. **I explain more robust options via APIs, including an automated script, within Chapter Twenty-Eight.**

Twilio (twilio.com/lookup)

This company provides VOIP services to many apps, companies, and individuals. A feature of their internet-based phone service is the ability to identify incoming calls through caller ID. Fortunately for us, they provide a page on their site that allows queries against their database, but it requires you to register for a free trial account. Locate the "Sign up" button and create a free account. Click the confirmation link in the email and you should be all set. If you are required to verify a telephone number, Google Voice numbers are accepted. You should be presented with a $15 credit within Twilio. This should provide numerous searches.

Once you are logged in to an account, click "Explore Products" in the left menu. Scroll down to the "Lookup" feature on the right. This is the screen which allows querying of numbers, and you may want to bookmark this page. I usually only select the "Include caller name" unless I want confirmation of the carrier. These searches are $0.01 each, which means you should be able to conduct 1500 queries on this single trial. While a bit of an annoyance to set up the first time, the work is justified. Make sure you have a valid email and Google Voice account ready before registering. If Twilio demands to know why you created an account, I would not mention OSINT or this book. They often cater to people who own a "small businesses which wants to transition from landlines to VOIP telephone service". That was a hint. If pushed further, tell them you are a developer who has been asked to review their API, and explain you will only need internal testing without an outside number.

On the Lookup search page, insert the target number and allow the page to generate a result, which will appear directly below. The result will include the type of provider and any name associated with the billing. If I only had one website to search, this would be it. Below is an actual redacted result.

PHONE NUMBER	+16180000000
NATIONAL FORMAT	(618) 000-0000
COUNTRY CODE	US
CALLER NAME	M Bazel
CALLER TYPE	null
CARRIER NAME	Google (Grand Central) - Level3 - SVR
CARRIER TYPE	voip

There are many other caller ID services, but they all require usage of API keys via Terminal commands. I will explain these in Chapter Twenty-Eight.

Apeiron (https://apeiron.io/cnam)

This service possesses data similar to Twilio, but is completely free and available through a traditional website. The results are not as comprehensive as the previous option, but could be valuable if you have Twilio account.

Truecaller (truecaller.com)

This service stands alone as the most creative telephone number lookup service. True Caller is an app for smart devices that displays caller ID information of incoming calls. If you receive a call on your phone, and the number is not in your contacts, True Caller searches its database and provides any results on your screen. You can then choose to accept or deny the call. This is fairly standard and is not the interesting aspect of this service. The fascinating part to me is the source of their caller database. It is mostly crowd-sourced.

When you install the app, you give it permission to collect all of your contacts and upload them to the original database. Basically, millions of users have uploaded their contact lists for the world to see. The next amazing thing to me is the ability to search within this data on the True Caller website. You must connect to the service via a covert Microsoft or Google account, but that is not difficult. When I first found this service, I was skeptical. I entered the cellular number of my government issued cellular telephone expecting to see no results. The response was "Mike Bazell". My jaw dropped. My super-secret number was visible to the world. This means that someone in my circle, likely another government employee, installed True Caller on his or her phone and had my information in their contacts. Until someone else populates data for this number, it will always be present in the database.

Real World Application: During my final year of government investigations, I queried a target number associated with a homicide through this service. The result was "Drug Dealer Matt". One of the target's customers must have installed True Caller. One of three potential suspects was named Matt, who earned our spotlight, and later an arrest.

Spy Dialer (spydialer.com)

This service offers a typical telephone number search tool, which also appears to extract data from crowd-sourced databases. I suspect that much of this data was acquired many years ago, but this can still be beneficial. Sometimes, I want historical data about a number, even if it is no longer issued to the target.

Caller ID Test (calleridtest.com)

This site was designed to input a telephone number and test the Caller ID display feature. It is nothing more than a standard lookup service, but I have found the data to be unique from other sources on some occasions. Unfortunately, I have also found the availability of this service to be unreliable. While the site is usually present, the results don't always populate. However, this resource should be checked when others have failed.

Telephone Search Websites

In previous editions of this book, I summarized a handful of people search websites which allowed the query of a telephone number. These sites all possess unique data sets and each should be searched. Most of the results originate from sources such as property tax data, marketing leaks, phonebooks, and various data breaches. Instead of explaining each site, which becomes quite redundant, I will display the static URL of a search submission. Many of these links avoid unnecessary loading screens and advertisements. This will help us with the automated tool at the end. Overall, we cannot control the results, and telephone search is mostly "what you see is what you get". Replace the demo number (618-462-0000) with your target number.

411 https://www.411.com/phone/1-618-462-0000
800 Notes https://800notes.com/Phone.aspx/1-618-462-0000
Advanced Background Checks https://www.advancedbackgroundchecks.com/618-462-0000
America Phonebook http://www.americaphonebook.com/reverse.php?number=6184620000
Caller Smart https://www.callersmart.com/phone-number/618-462-0000
Cyber Background Checks: https://www.cyberbackgroundchecks.com/phone/618-462-0000
Dehashed https://dehashed.com/search?query=6184620000
Fast People Search https://www.fastpeoplesearch.com/618-462-0000
Info Tracer https://infotracer.com/phone-lookup/results/?phone=6184620000
John Doe https://johndoe.com/phones/6184620000
Numpi https://numpi.com/phone-info/6184620000
Nuwber https://nuwber.com/search/phone?phone=6184620000
OK Caller https://www.okcaller.com/6184620000
People Search Now https://www.peoplesearchnow.com/phone/618-462-0000
Phone Owner https://phoneowner.com/phone/6184620000
Reveal Name https://www.revealname.com/618-462-0000
Reverse Lookup https://www.reverse-lookup.co/618-462-0000
Search People Free https://www.searchpeoplefree.com/phone-lookup/618-462-0000
Spytox https://www.spytox.com/reverse-phone-lookup/618-462-0000
Sync.me https://sync.me/search/?number=16184620000
That's Them https://thatsthem.com/phone/618-462-0000
True People Search https://www.truepeoplesearch.com/results?phoneno=(618)462-0000
US Phonebook https://www.usphonebook.com/618-462-0000
White Pages https://www.whitepages.com/phone/1-618-462-0000
WhoseNo https://www.whoseno.com/US/6184620000
Yellow Pages https://people.yellowpages.com/whitepages/phone-lookup?phone=6184620000
Zabasearch https://www.zabasearch.com/phone/6184620000
Google https://www.google.com/search?q=618-462-0000
Bing https://www.bing.com/search?q=618-462-0000
Yandex https://yandex.com/search/?text=618-462-0000

Search Engines

Google and Bing were once a great place to find basic information about a target phone number. These sites can still provide valuable information, but the amount of spam that will display in the results is overwhelming. Many of the links presented will link to sites that will charge a fee for any information associated. This information is usually the same content that could have been located with an appropriate free search. I do not recommend giving in to these traps. While we can't ignore a traditional search of telephone numbers, we can customize the queries in order to achieve the best results. Before explaining advanced telephone owner identification, we should take a look at appropriate search engine structure.

Most people use traditional search engines as a first step toward identifying the owner of a telephone number. The number is usually provided in a standard format such as 202-555-1212. This can confuse some search engines because a hyphen (-) is often recognized as an operator to exclude data. Some engines might view that query as a search for 202 but not 555 or 1212. Additionally, this search might identify a website that possesses 202-555-1212 within the content but not one that contains (202) 555.1212. If this is your target number, all of the following should be searched in order to exhaust all possibilities. The quotation marks are important to prevent the hyphen from being seen as an operator.

"2025551212"	"202 555 1212"	"(202) 555.1212"	"(202)555.1212"
"202-555-1212"	"(202) 5551212"	"(202)5551212"	
"202.555.1212"	"(202) 555-1212"	"(202)555-1212"	

This may seem ridiculous, but I am not done. Many websites forbid users to post a telephone number, such as many auction sites, but people try to trick this restriction. They will type out a portion of their number to disclose contact information. While not a complete list of options, the following should also be searched.

"two zero two five five five one two one two"	"202 five five five one two one two"
"two zero two five five five 1212"	"202 555 one two one two"
"two zero two 555 one two one two"	"202 five five five 1212"
"two zero two 555 1212"	

This list would not capture a post that included (202) 555 twelve twelve, but you get the point. After submitting these through Google, you should attempt each through Bing. In my effort to always provide search tools which automate and simplify these techniques, I have added this feature to your telephone tools presented at the end of the chapter. The right-middle portion of this page, displayed later in Figure 18.03, allows you to enter a numerical and written target telephone number. Clicking the submit button launches a series of JavaScript commands that launch eight new tabs within your browser. The first four are custom Google searches with the target data and the last four repeat the process on Bing. The following four searches are conducted on both services, using the example data entered previously.

"2025551212"OR"202-555-1212"OR"202.555.1212"OR"202 555 1212"

"(202) 5551212"OR"(202) 555-1212"OR"(202) 555.1212"OR"(202)5551212"OR"(202)555-1212"OR"(202)555.1212"

"two zero two five five five one two one two"OR"two zero two five five five 1212"OR"two zero two 555 one two one two"OR"two zero two 555 1212"

"202 five five five one two one two"OR"202 555 one two one two"OR"202 five five five 1212"

Notice that these queries use quotation marks to obtain exact results and the OR operator to search multiple options independently from each other. You will likely receive many false positives with this method, but you are less likely to miss any relevant results.

Sly Dial (slydial.com)

This service contacts the cellular provider of the target telephone number and sends you straight to the outgoing voicemail message. This can allow you to hear their voice and name without ringing the phone and hoping they do not pick up. Sly Dial does not work through a website and you do not need to create an account to use the service. Instead, you must call a general Sly Dial telephone number and follow the automated prompts. You must listen to a brief advertisement before your call is placed. Finally, the service will play the target's outgoing voicemail message through this audible telephone call. Since a website is not involved, there is no option to download an audio file of the call.

Sly Dial does not usually ring the suspect's telephone. It will likely not show "missed call" or any other indicator that a call occurred. In my testing, less that 5 percent of the attempts actually cause the target telephone to ring only one time. Calling the missed call back reveals nothing about the identity of the number. Ultimately, there is a very small chance that the target will know that someone attempted a call. In the rare occurrence that the telephone rings, the target will never know the identity of the person making the calls. To use the Sly Dial service, call 267-759-3425 (267-SLYDIAL) from any telephone service including landlines, cellular lines, or VOIP. Follow the directions during the call. If this number does not work, visit slydial.com for updates. I want to stress the following one additional time. Use these services at your own risk. If accidentally notifying your target that you are conducting these types of activities could compromise your investigation, avoid this technique.

Old Phone Book (oldphonebook.com)

I first noticed this niche service in late 2018. It provides historical White Pages landline listings from 1994-2014. The sources are official databases collected from many years of telephone CD-ROMs. These were purchased by various companies throughout several decades as a more convenient option than traditional phone books. The data is quite impressive, and the following direct URLs allow us to add these to our tools. Results include historic addresses attached to each year. Figure 18.01 displays an actual redacted result from the official website. This provides an old address, and assumes that the target moved to a new address between 1998 and 2001. This search is vital for background checks.

http://www.oldphonebook.com/searchphone2.php?syear=1994&sphone=6184620000
http://www.oldphonebook.com/searchphone2.php?syear=1995&sphone=6184620000
http://www.oldphonebook.com/searchphone2.php?syear=1996&sphone=6184620000
http://www.oldphonebook.com/searchphone2.php?syear=1997&sphone=6184620000
http://www.oldphonebook.com/searchphone2.php?syear=1998&sphone=6184620000
http://www.oldphonebook.com/searchphone2.php?syear=2001&sphone=6184620000
http://www.oldphonebook.com/searchphone2.php?syear=2002&sphone=6184620000
http://www.oldphonebook.com/searchphone2.php?syear=2003&sphone=6184620000
http://www.oldphonebook.com/searchphone2.php?syear=2007&sphone=6184620000
http://www.oldphonebook.com/searchphone2.php?syear=2008&sphone=6184620000
http://www.oldphonebook.com/searchphone2.php?syear=2013&sphone=6184620000
http://www.oldphonebook.com/searchphone2.php?syear=2014&sphone=6184620000

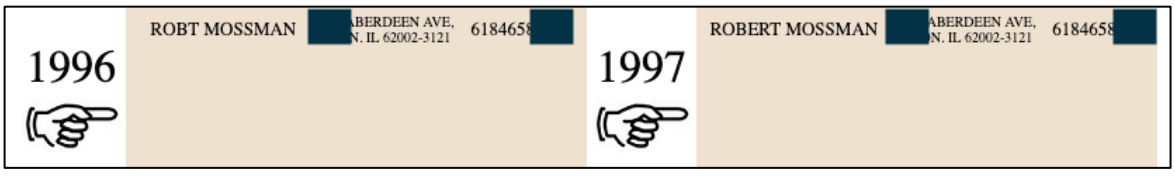


Figure 18.01: A partial redacted result from Old Phone Book.

Craigslist (craigslist.org)

Craigslist has already been discussed in earlier chapters, but the phone search options should be further detailed. Many people use Craigslist to sell items or services. The posts that announce the item or service available will often include a telephone number. These numbers will belong to a landline or cellular provider. This can be a great way to identify unknown telephone numbers. Some posts on Craigslist will not allow a telephone number to be displayed on a post. It is a violation of the rules on certain types of posts. Some people choose not to list a number because of automated "scrapers" that will grab the number and add it to databases to receive spam via text messages. Either way, the solution that most users apply to bypass this hindrance is to spell out the phone number. Instead of typing "314-555-1212", the user may enter "three one four five five five one two one two". Some will get creative and post "314 five five five 1212". This is enough to confuse both Craigslist's servers as well as the spammers. This can make searching difficult for an analyst. The hard way to do this is to conduct several searches similar to the following.

site:craigslist.org "314-555-1212"
site:craigslist.org "314" "555" "1212"
site:craigslist.org "three one four" "five five five" "one two one two"
site:craigslist.org "314" "five five five" "1212

This list can get quite long if you try to search every possible search format. One search that will cover most of these searches in a single search attempt would look like the following.

site:craigslist.org "314"|"three one four" "555"|"five five five" "1212"|"one two one two"

The "|" symbol in this search is the same as telling Google "OR". In essence, we are telling Google to search "314" or "three one four", then "555" or "five five five", and then "1212" or "one two one two". With this search, you would receive a result if any combination of the following was used.

314-555-1212
3145551212
314 555 one two one two
three one four 555-1212

This search will not catch every possible way to post a phone number. For example, if the user had typed "314 555 twelve twelve", the above technique would not work. The researcher must consider the alternative ways that a target will post a number on a website. It may help to imagine how you would post the target number creatively on a site, and then search for that method. Additionally, searching for only a portion of the number may provide results. You may want to try searching only the last four digits of the number. This may produce many unwanted results, but your target may be within the haystack. An automated option is included in the search tools.

Grocery Reward Cards / Loyalty Cards

Most grocery chains have adopted a reward/loyalty card system that mandates the participant to enroll in their program. The consumer completes an application and receives a plastic card to use during checkout for discounts. Many of these stores only offer a sale price if you are a member in the program. Most consumers provide a cellular telephone number to the program and use that number during checkout. This eliminates the need of possessing a physical card in order to receive the discount. Instead, they type their cell number into the card swiping machine to associate the purchase with their membership. These programs contain a huge database of telephone numbers and the registered users. There is no online database to access this data. However, you can obtain this data if you are creative.

Assume your target telephone number is 847-867-5309. If you have tried every technique mentioned at this point to identify the owner and failed, you may consider a query with a local grocery chain. The easiest method

is to enter the store, purchase a pack of gum, and enter the target telephone number as the reward/loyalty program number. You will likely receive a receipt with the target's name on the bottom. Figure 18.02 (left) displays a portion of the actual receipt that I received when using this number. If you prefer to avoid entering a store, drive to the company's gas station outside of the store. Figure 18.02 (right) displays the notification I received when entering this same number at the pump. Note that this number is fictional. However, it has been registered at practically every grocery store in the United States. Try to use it the next time you make a purchase.

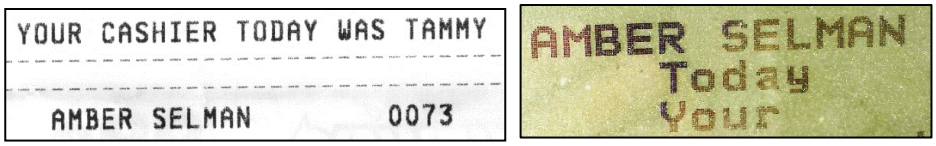


Figure 18.02: A receipt (left) and gas pump (right) identifying the owner of a cell number.

The Other Board (theotherboard.com)

NSFW! If you have any suspicion that the target of your investigation is involved in prostitution, drugs, or any related activity, The Other Board should be checked against the telephone number of your subject. This website aggregates all of the prostitution classifieds and review sites into one search. It extracts the telephone numbers from all online classified pages and allows you to search by the target telephone number. One of my training examples identified 37 online photos, 20 escort ads, several reviews by "Johns", ages used by the target, the last known location, and locations visited based on postings in online classifieds. Any time I have a target telephone number that is likely involved in criminal activity, I conduct a brief search on this site, via URL, as follows.

https://theotherboard.com/escort-reviews?phone=6185551212

Another option for this type of query is **The Erotic Review** (theeroticreview.com), with the following structure. Notice that the area code is separate from the number.

https://www.theeroticreview.com/reviews/newreviewsList.asp?Phone=5551212&AreaCode=618

Contact Exploitation

I mentioned this technique previously during the Android emulation chapter. If I receive no valuable information about a target telephone number with the previous options, I proceed to my Android emulator. I add the number as a friend named "John" in the contacts of the device. I then open any applicable application, such as Twitter, Instagram, and dating apps, and ask them to "find my friends". More often than not, an app links me to a social network profile which is associated with the only number in my contact list.

Real World Application: In 2019, I was asked to assist an attorney in a child custody and divorce case. He provided the cellular telephone numbers of the suspected cheating husband and an unknown number of a potential mistress identified from telephone records. I added both numbers to the contacts list in my Android emulator and checked every app daily. This method identified a dating profile which included the husband's photo, but with a different name. On the same day which he was served divorce papers, I observed both numbers show up in the encrypted communications app Signal. While it did not identify the person on the other end of the communication, the coincidence of the two numbers joining Signal on the same day is revealing. While writing this chapter, I successfully identified the owner of a "burner" number because he connected it to his WhatsApp account. I added the number to my Contacts within my Android emulator; Opened WhatsApp; and asked it to find any "friends". It immediately displayed an image of a face associated with the number. A reverse image search (explained later) identified a social network in my target's true name. I have also used this technique to identify Skype profiles associated with a cellular number.

Calls.ai (sync.ai/calls-ai)

This Android application is impressive. It queries a large database of crowd-sourced contact details from unknown sources. Most telephone number search results reveal only a first name, but I find them to be highly accurate. Conduct the following to use this app.

- Launch your Android virtual machine through VirtualBox or Genymotion.
- Launch the Google Play Store or Aurora Store and search "Calls.ai".
- Install Calls.ai from Sync.ai.
- Launch the Calls.ai app and allow default permissions.
- When prompted, associate with the Google account within the VM.
- Click the search option in the lower left and input any number.

As a test, I searched three cellular numbers which were assigned to me when I worked for the government. All three returned with positive results displaying "Mike", Michael" and "Bazzel". None of these phones were publicly attached to my name until someone shared their contact details with the world. While it is a nuisance to open an Android VM each time I need to conduct a search, the results from Calls.ai, Truecaller, and others justifies the time.

IntelTechniques Telephone Tool

Of all the automated search tools, this one may save the most time. You can submit an area code, prefix, and number and execute all queries. I also added a section of Caller ID services which allow query via direct URL. If you want to use those, you must add your API keys to the source code of the page. I prefer the script previously explained, but I want you to have options. The number and word area on the search tool replicates the techniques mentioned previously which attempt to search both the numbers and words of a target telephone number within Google, Bing, and Craigslist. Figure 18.03 displays the current state of the tool.

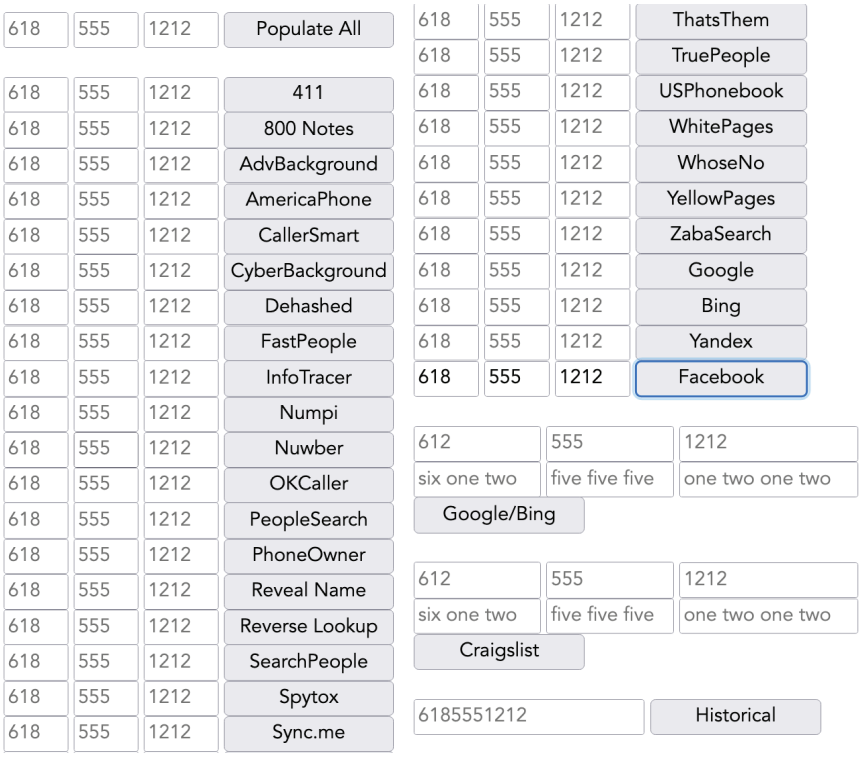


Figure 18.03: The IntelTechniques Telephone Tool.

CHAPTER NINETEEN
ONLINE MAPS

The presence of online satellite images is not news anymore. Most of you have already "Googled" your own address and viewed your home from the sky. This view can get surprisingly detailed when using the zoom feature. Alleys, sheds, and extended driveways that are hidden from the street are now visible thanks to this free service. Many tactical units will examine this data before executing a search warrant at a residence. Aerial maps are helpful for viewing the location of exiting doors, escape routes, stairs, and various obstructions. The rapidly growing availability of the Street View option now gives us more data. This chapter explains detailed use of Google, Bing, and other mapping services. At the end, I present my custom maps tool which provides an automated solution to collecting every possible view associated with your target of interest.

Google Maps (maps.google.com)

Google constantly makes changes to their online Maps service, attempting to streamline the entire Maps experience to make everything easier to use. Over the past two years, we have witnessed features come and go, only to randomly re-appear. The following basics of Google Maps are now default for all users.

Search Bar: The Google Maps search bar can now accept practically any type of input. A full address, partial address, or GPS coordinates will immediately present you with a mapped view. Company names and types of businesses, such as "café" will highlight locations that may be of interest. This search field is the first stop. Attempt any search relevant to your investigation and you may be surprised at how accurate Google is. You can collapse this entire menu by clicking the left arrow next to the search field. This will return you to a full screen view of the map.

Satellite View: The lower left area of any map will offer a satellite view. The satellite view is a direct view from the sky looking almost straight down. A satellite view of your target location is always vital to every investigation.

Globe View: The Globe view (globe icon) in the lower right is similar, but usually offers a unique view from a different date. It also presents a rotation option. While in the Globe view, click on the "3D" icon and then the small "rotation" icon to the right of the map. This will shift the view 45 degrees and a second click will shift an additional 45 degrees. This presents new angles of your target.

Street View: If the Street View option is available, Google has been in the area and captured a photo of the location from the street. Dragging and dropping the small orange person in the lower right menu will open a street view from ground level at the area specified. You can navigate through this view by clicking forward, clicking and dragging, or scrolling and zooming. This view can be zoomed in by double-clicking and panned left and right by dragging the mouse while holding a left click. Double-clicking an area of the street will refresh the window to the view from that location. Clicking the map in the lower left will return you to the standard map view.

Historic Street View: In late 2014, Google began offering the ability to view all stored street view images for any single location. This option is available within the standard street view layout within the search area of the upper left corner. Click on the small clock in order to launch a pop-up window. This new view will allow you to move a slider bar which will present different views. The month and year of image capture will also appear for documentation. Figure 19.01 displays this method which presents an additional view of a parking lot from a few years prior. Additional options include views from several years prior. This can often reveal additional vehicles or missing structures associated with an investigation.

Order 0138471223

Figure 19.01: Historic Street View options from Google Maps.

Distance Measurement: Google Maps reintroduced the distance measurement tool after completely disabling the classic maps interface in 2015. While in map or satellite view, right-click on your starting point and choose "Measure distance". Click anywhere on the map to create a path you want to measure. Further clicks add additional measuring points. You can also drag a point to move it, or click a point to remove it. The total distance in both miles or kilometers will appear under the search box. When finished, right-click on the map and select clear measurement.

GPS Coordinates: Clicking on any point will load a small window in the bottom center that identifies the exact GPS coordinates of the chosen location. If this is not visible, right-click any point and select "What's here". Searching via GPS will be more precise than other queries.

URL Queries: For the purposes of the search tool, which is explained at the end of the chapter, we always want to identify the GPS coordinates of our target. In my examples throughout the chapter, 41.947242 is the latitude coordinate and -87.65673 is the longitude. The static search URLs I explain throughout this chapter will assist us in quickly querying multiple services once we identify a target location. The static URL displaying various Google Maps views of our target location is as follows.

Google Standard Map View:
https://www.google.com/maps/@?api=1&map_action=map&center=41.947242,-87.65673&zoom=18&basemap=roadmap

Google Map View with Terrain:
https://www.google.com/maps/@?api=1&map_action=map&center=41.947242,-87.65673&zoom=18&basemap=terrain

Google Satellite View:
https://www.google.com/maps/@?api=1&map_action=map&center=41.947242,-87.65673&zoom=18&basemap=satellite

Google Street View

Google Street Views possess multiple options, including a 360-degree horizontal scope and vertical pitch. I have created the following URLs which we will use in our custom tools. These offer a view of north, east, south, and west with no pitch. After each URL loads, you can use the cursor to drag the view to fit your needs.

Google Street View (Down):
https://www.google.com/maps/@?api=1&map_action=pano&viewpoint=41.947242,-87.65673&heading=0&pitch=-90&fov=100

Google Street View (North):
https://www.google.com/maps/@?api=1&map_action=pano&viewpoint=41.947242,-87.65673&heading=0&pitch=0&fov=90

Google Street View (East):
https://www.google.com/maps/@?api=1&map_action=pano&viewpoint=41.947242,-87.65673&heading=90&pitch=0&fov=90

Google Street View (South):
https://www.google.com/maps/@?api=1&map_action=pano&viewpoint=41.947242,-87.65673&heading=180&pitch=0&fov=90

Google Street View (West):
https://www.google.com/maps/@?api=1&map_action=pano&viewpoint=41.947242,-87.65673&heading=270&pitch=0&fov=90

Bing Maps (bing.com/maps)

Similar to Google Maps, Bing offers a map view, satellite view, angled view, and street view. In my experience, the imagery provided by Bing will always be unique to the images available in Google Maps. A side-by-side comparison can be seen in a few pages with the custom maps tool. Bing does not always present the various 3D view options, which they call "Bird's Eye View". Fortunately, we can replicate all views with the following custom URLs.

Bing Standard Map View:
https://www.bing.com/maps?cp=41.947242~-87.65673&lvl=20

Bing Satellite View:
https://www.bing.com/maps?cp=41.947242~-87.65673&lvl=20&sty=a

Bing Satellite View (North):
https://www.bing.com/maps?cp=41.947242~-87.65673&lvl=20&sty=o&w=100%&dir=0

Bing Satellite View (East):
https://www.bing.com/maps?cp=41.947242~-87.65673&lvl=20&sty=o&w=100%&dir=90

Bing Satellite View (South):
https://www.bing.com/maps?cp=41.947242~-87.65673&lvl=20&sty=o&w=100%&dir=180

Bing Satellite View (West):
https://www.bing.com/maps?cp=41.947242~-87.65673&lvl=20&sty=o&w=100%&dir=270

Bing Street View (North):
https://www.bing.com/maps?cp=41.947242~-87.65673&lvl=20&dir=0&pi=0&style=x

Bing Street View (East):
https://www.bing.com/maps?cp=41.947242~-87.65673&lvl=20&dir=90&pi=0&style=x

Bing Street View (South):
https://www.bing.com/maps?cp=41.947242~-87.65673&lvl=20&dir=180&pi=0&style=x

Bing Street View (West):
https://www.bing.com/maps?cp=41.947242~-87.65673&lvl=20&dir=270&pi=0&style=x

Zoom Earth (zoom.earth)

This multiple satellite imagery website presents views from NASA, Bing, and ArcGIS. Occasionally, the ArcGIS data is more recent than Google or Bing. The smooth interface will easily provide a comparison of the available images for any location. One advantage of Zoom Earth is the ability to view satellite images in true full-screen mode. This allows creation of full-screen captures without branding, menus, or borders. This could be more appropriate for live demonstration instead of a standard Google or Bing window. The static URLs with satellite view are as follows. The "20z" is the maximum zoom level. The second URL is a historic view from the Economic and Social Research Institute (ESRI) database.

https://zoom.earth/#view=41.947242,-87.65673,20z
https://zoom.earth/#view=41.947242,-87.65673,20z/layers=esri

Here (wego.here.com)

This service has experienced a lot of transformation over the past ten years. Originally a Nokia company with a unique set of satellite imagery, it now mostly contains identical images to Bing Maps. However, I find unique images on occasion, and we should always consider every resource. The static URL with satellite view is as follows.

https://wego.here.com/?map=41.947242,-87.65673,20,satellite

Yandex (yandex.com/maps)

Satellite views from Yandex are often poor quality and low resolution when focused on the U.S. However, international locations tend to possess clearer results. The static URL with satellite view is as follows.

https://yandex.com/maps/?l=sat&ll=-87.65673%2C41.947242&z=20

Descartes Labs (maps.descarteslabs.com)

This service offers a very unique search option that I have not found present on any other site. After locating a target of interest, it displays a satellite view sourced from the National Agriculture Imagery Program (NAIP). The unique part is the ability to search based on image. In other words, you can select a monument, park, building, or any other view and request any images that match. As an example, I selected the baseball grounds at Wrigley Field and I was immediately presented hundreds of baseball fields all over the world. I have yet to determine how I would execute this strategy within an investigation, but this feature has potential.

https://search.descarteslabs.com/?layer=naip_v2_rgb_2014-2015#lat=41.94721&lng=-87.656502&skipTut=true&zoom=20

Snapchat (map.snapchat.com)

This was mentioned previously while discussing social networks, but should also be present here. This map allows you to zoom into a location and view any public stories posted through Snapchat. I find this does not work well when investigating a home address, but can be incredibly useful when researching a specific event, school, or public area. The static URL is as follows.

https://map.snapchat.com/@41.947210,-87.656502,20.00z

Wikimapia (wikimapia.org)

This is a great catch-all option. It combines the various satellite options for Google, Bing, and others. It also possesses crowd-sourced data and images. The menu in the upper right allows you to choose the satellite service while clicking on any location or building and presents a report of images, websites, and details about the target.

Dual Maps (data.mashedworld.com/dualmaps/map.htm)

This website provides a satellite view of a location on both Google Maps and Bing Maps simultaneously. The Google view on the left will also contain a search field in the lower left corner. Searching an address in this field will center both maps on the same address. This will provide a comparison of the satellite images stored on each service. This can quickly identify the service that has the better imagery. While this can provide a quick side-by-side comparison, upcoming automated solutions are preferred.

Real Estate (zillow.com, redfin.com, etc.)

These websites display information about homes which are currently available for sale or have recently sold. I usually search this option in hopes of locating interior images of a suspect home. This can often identify personal belongings, interests, and family information. It is a bit of a long-shot, but has been vital to a few of my investigations.

Vidi (vidi.world)

If you need live video from a specific location, you can hire someone through this site. If you wanted a live video stream in front of a specific business, you can offer a "bounty" for that task. If a member in that area accepts, the footage is sent electronically. We live in strange yet fascinating times.

Historic Imagery

Researching different satellite views of a single location can have many benefits. These views are all of the current content stored within each service. However, these mapping services continuously update their offerings and usually present the most recent option. You may want to view the previous content that was available before an image was updated.

Historic Aerials (historicaerials.com)

If you need satellite imagery from several years prior, you can visit Historic Aerials. The quality will often be poor, especially as you view imagery from previous decades. After you enter an address, you will be presented all available options on the left side of the page. Figure 19.02 displays several results of the same location over a twenty-year period. These views will be unique from all of the previously mentioned services.

World Imagery Wayback (livingatlas.arcgis.com/wayback)

I began using this site in 2018 as an alternative to Historic Aerials. The quality is superior, the number of satellite views is higher, and the functionality is smoother. Figure 19.03 displays four images acquired from the service of a specific location. Each provides a unique view from 2014, 2015, 2016, and 2017.

Old Maps Online (oldmapsonline.org)

If you simply need road maps from the past 100 years, this service has you covered. There are no images or satellite views, but street details from previous decades could be useful. You might learn that a target location possessed a different street name in the past or identify the approximate year when a specific road was built.

Real World Application: Combining several satellite views can provide much information about a target's residence. Before the execution of a search warrant, it is beneficial for police to collect as much map information as possible. This will give updated views of a drawn map, satellite imagery directly above the house, four angled views from the sky, and a complete view of the house and neighboring houses from the street, including vehicles. This can be used to identify potential threats such as physical barriers, escape routes, and video surveillance systems in place.

In 2020, I assisted a former colleague with a missing person investigation. Google and Bing maps were queried, but nothing helpful was found. However, a search on Mapillary (explained next) revealed a vehicle parked outside the victim's residence which was otherwise unknown to investigators. This clue opened a new path of investigation which ultimately led my colleague to the victim. We must always thoroughly check every resource.

Figure 19.02: Multiple views of a location through Historic Aerials.

Figure 19.03: Historic satellite images from World Imagery Wayback.

Crowd-Sourced Street Views

Street View maps from services such as Google and Bing are nothing new. Most of you can view the top and front of your home from multiple online websites. With street view options, these services are fairly responsible and block most faces and license plates. This makes it difficult for investigators trying to identify a suspect vehicle parked at a home or present at a crime scene prior to the incident. We have two services that offer unique street-level views that may remove these limitations.

Mapillary (mapillary.com)

This service appears similar to other mapping websites when first loaded. You see a typical map view identifying streets, landmarks, and buildings. Enabling the satellite view layer displays images from the Open Street Map project. The real power is within the crowd-sourced street view images. While in any map view, colored lines indicate that an individual has provided street-level images to Mapillary, usually from a GPS-enabled smart phone. This is actually quite common, as many people record video of their driving trips which could be used in case of an accident. These services make it easy and automated to upload these images. The sites then embed these images within their own mapping layers for the public to see. Clicking these colored lines reveals the street view images in the lower portion of the screen. Expanding these images allows you to navigate through that individual's images similar to the Google Street View experience. Figure 19.04 displays a street view layered over a satellite view. The username of the Mapillary member and date of image capture appears in the lower left.

In some of these images, the services appear to be redacting license plates with a typical "Blur Box" as seen in Figure 19.05 (left). A few feet later, the box disappears and a partially legible license plate is revealed, as seen in Figure 19.05 (right). It seems like these services are attempting to determine when a license plate is legible, and then blurring it. When the plate is farther away and more difficult to read, it is ignored. This can work in our favor. In Figure 19.06 we can use selective cropping and photo manipulation to obtain the registration. The left image appears unaltered as it is difficult to read. The right image was cropped; inverted with Photoshop; brightness turned down to 0%; and contrast heightened to 100%. The result is a legible license plate. I believe most registration plates can be made visible within these services. The following URL conducts a query for our target location.

https://www.mapillary.com/app/?lat=41.947242&lng=-87.65673&z=18

Karta (kartaview.org)

After exhausting your options on Mapillary, continue to Karta, previously called Open Street Cam. The site functions in the same way, but possesses unique images. These sites allow you to identify the uploader's username, mapping history, number of posted images, and profile image. You can also select to watch all of the captured images from a specific user as he or she travels daily. I can't begin to imagine the amount information available about a user's travel habits if he or she were to become a target of an investigation. While there is not coverage of every area like we see with Google Maps, the databases are growing rapidly, and should be included when using other mapping tools. We will use the following URL structure for our automated tools.

https://kartaview.org/map/@41.947242,-87.65673,18z

Land Viewer (eos.com/landviewer)

This resource will not present detailed views of your target's home. The images here are often generated from weather satellites, and restrict zoom levels to a city view. Most locations offer four active satellites that constantly retrieve images and five inoperative satellites that store historic imagery dating back to 1982. I have only used this resource to document potential weather at a crime scene (clouds, rain, or clear). The static URL follows.

https://eos.com/landviewer/?lat=41.947242&lng=-87.65673&z=11

Landsat Look (landsatlook.usgs.gov)

This is very similar to Land Viewer, but with additional historic images. There are three unique maps which can be viewed within a browser. The following URLs present each option.

https://landlook.usgs.gov/viewer.html
https://landsatlook.usgs.gov/viewer.html
https://landsatlook.usgs.gov/sentinel2/viewer.html

Each map possesses unique satellite views ranging from 1972 to present. The menu in the upper right of each map displays the image options. The instructions for accessing this data are not straightforward, so I will provide a typical scenario.

- Drag and zoom to your target location.
- Select your desired satellite in the upper-right menu.
- Choose desired date range.
- Click the "Show Images" button.

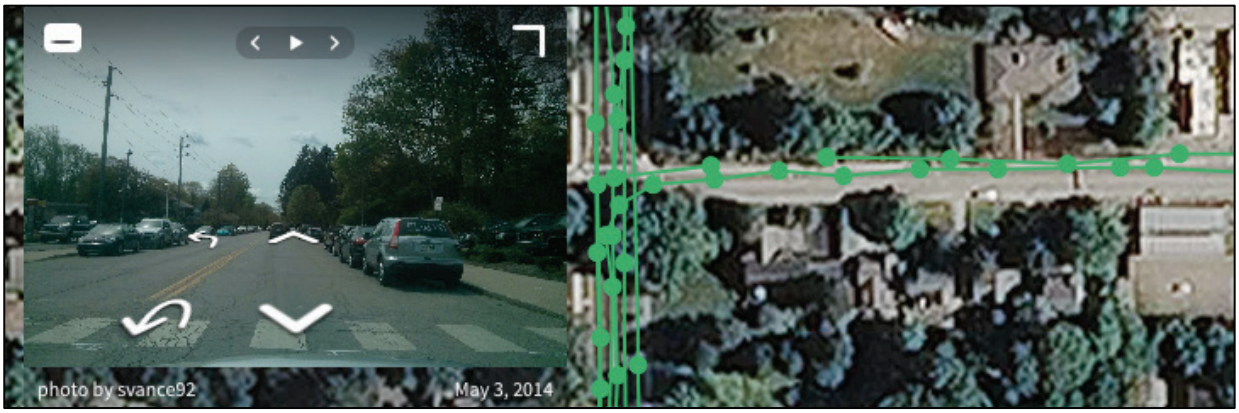


Figure 19.04: A crowd-sourced street view from Mapillary.

Figure 19.05: A vehicle with a blurred registration plate (left) and clear (right).

Figure 19.06: An illegible registration plate (left) and manipulated view (right).

Acre Value (acrevalue.com)

This site relies on Mapbox for aerial imagery, but offers a unique service unavailable in other mapping applications. It attempts to identify property boundaries for an address. Figure 19.07 displays an example. Acre Value requires a paid account in order to see property owner details. However, we can extract this information from the site with a few steps. First, right-click the site and launch the Inspector, as previously discussed, and click the "Network" tab. Next, click on the property of interest, as visible in Figure 19.07. You should see a "xhr" entry within the Inspector, which should appear similar to the following.

https://www.acrevalue.com/map/api/features/?box=-97.22935721804198,40.75791552615834,-97.18190577489744,40.766432787155054&feature=plots&subtype=sold-land&type=plots

After selecting this entry, click the "Response" tab on the right; expand "data"; expand "parcels"; and look for entries titled "0", "1", etc. Enter the "Parcel ID" for your target into the search field titled "filter properties". This should eliminate all options except your target and present something similar to Figure 19.08.

Figure 19.07: An Acre Value map listing.

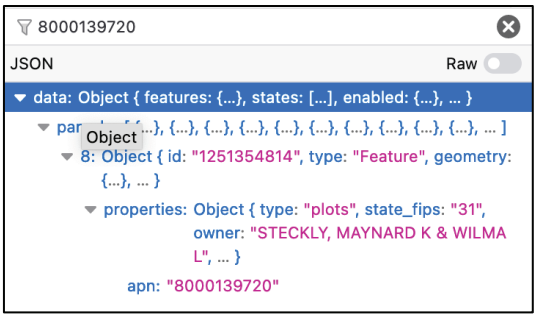


Figure 19.08: Inspector results identifying a land owner on Acre Value.

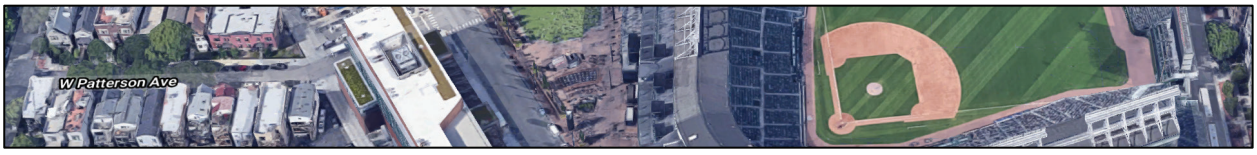
Apple Maps (satellites.pro)

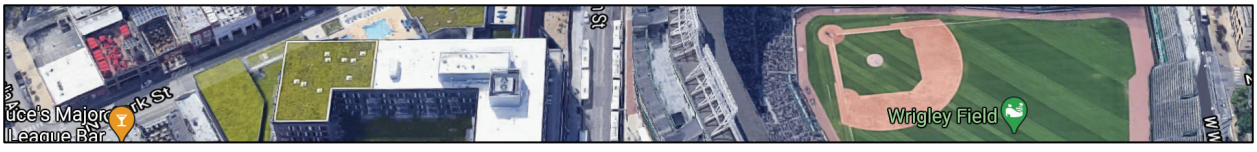
While not an official Apple product, Satellite Pro provides Apple Maps aerial views during search queries. A direct URL of our target location follows.

https://satellites.pro/USA_map#41.94721,-87.656502,18

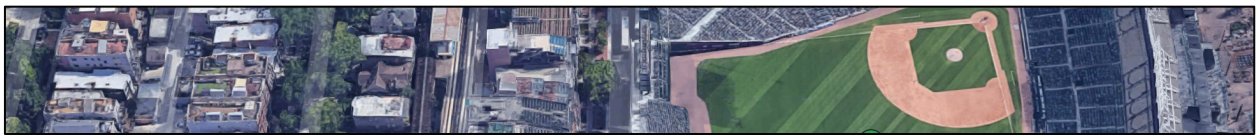
IntelTechniques Maps Tool (https://inteltechniques.com/tools/Location.html)

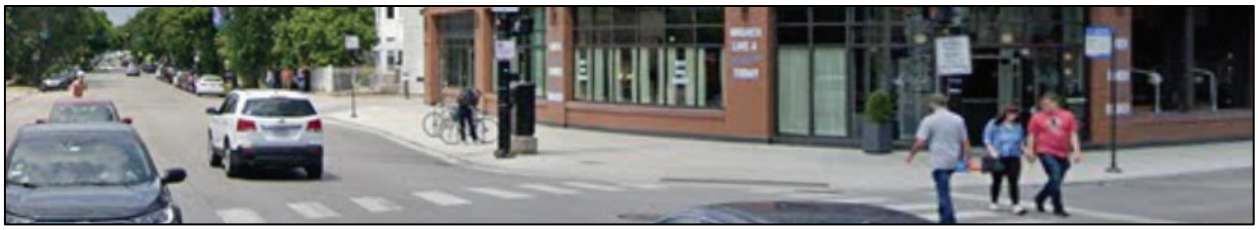
The first portion of this tool allows entry of a physical address in traditional format which can execute several searches previously mentioned. The first field presents a mapping API page which includes the GPS coordinates of your target. These will be used in the second portion of the page. This area allows entry of GPS coordinates for several mapping options. The "Populate All" simplifies entry into the query options. Each search will open results in a new tab, or the final "Submit All" will display all satellite imagery from multiple providers within new tabs. It currently fetches images from all options mentioned in this chapter which allow a static URL with GPS for submission. Figures 19.09 through 19.36 on the following pages display the results from these providers when searching Wrigley Field in Chicago, using the search tool to generate each result. Remember to allow pop-ups within your browser if you want all of the data to populate within new tabs automatically.



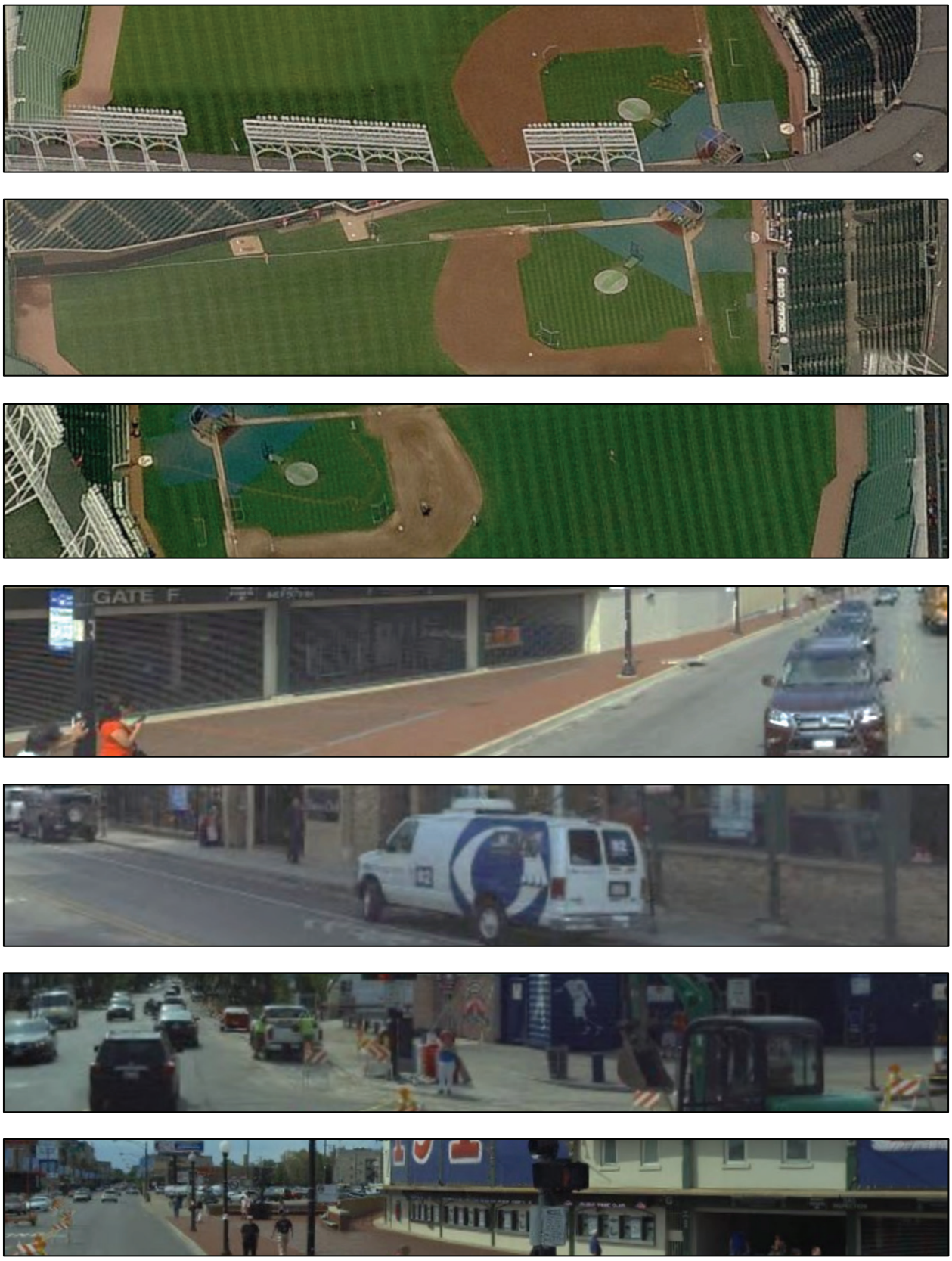


Figures 19.09 through 19.14: Google Satellite, Google Globe, Bing Satellite, Here Satellite, Google Globe North, and Google Globe East.



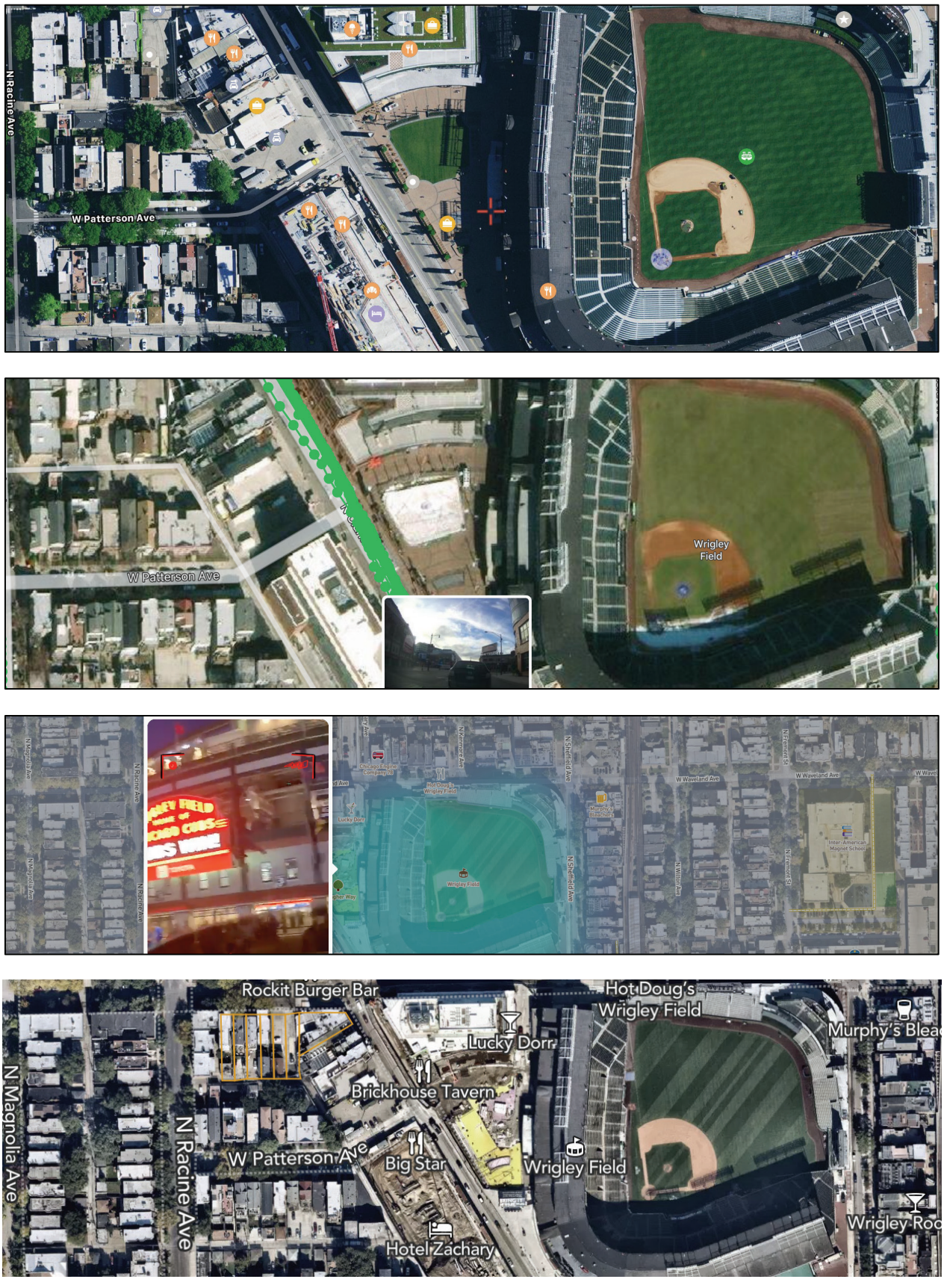


Figures 19.15 through 19.21: Google Globe South, Google Globe West, Google Street views of North, East, South, and West, and Bing Bird's Eye view of North.



Figures 19.22 through 19.28: Bing Bird's Eye views of East, South, and West, and Bing Street views of North, East, South, and West.

Figures 19.29 through 19.32: Zoom Earth, Zoom ESRI, Yandex Satellite, and Descartes.



Figures 19.33 through 19.36: Satellite Pro (Apple), Mapillary, Snapchat, and Acre Value.

Map Shadows

If you look closely at the previous images, you can see many shadows. We can use these to make a guess at the date and time of capture. In fact, analyzing shadows within any images, even those which do not originate with satellite imagery, can help identify the date and time of capture. There is no exact science to this (yet), but we have online tools available to assist our efforts. Consider the image in Figure 19.33. The shadow of Wrigley Field is due west of the building and does not appear to project north or south of the building. We can use **Shade Map** (shademap.app) to identify the most likely date and time of this satellite capture. After loading the site, search for Wrigley Field and allow the location to load. Next, we can begin manipulating the map.

Click the date in the lower-left and drag the month bar until the shadows seem to match the horizontal position within our target image in Figure 19.33. In my attempts, the month of March displayed shadows which did not extend north or south of the building at an angle. This month displayed shadows which extended directly east and west. Next, choose the time of day which matches the position of the shadows. My attempts identified 6:28 am Mountain (7:28 am Central) as the ideal match. Therefore, my best guess is that the satellite image captured in this example was approximately 7:30 am local time in late March. Figure 19.37 displays my result. I would never document an assumption of an exact date and time, but a range could be appropriate. Documenting that the shadows indicate capture between February and April from 7:00 to 8:00 am could suffice.

While this book is printed with black ink, hopefully you can match the overall shadows. Replicate this on your own computer to see the true value. This technique could also be used when you identify a photograph posted to social media. If you know the approximate location and time of year, it should be fairly easy to establish the approximate time of capture.

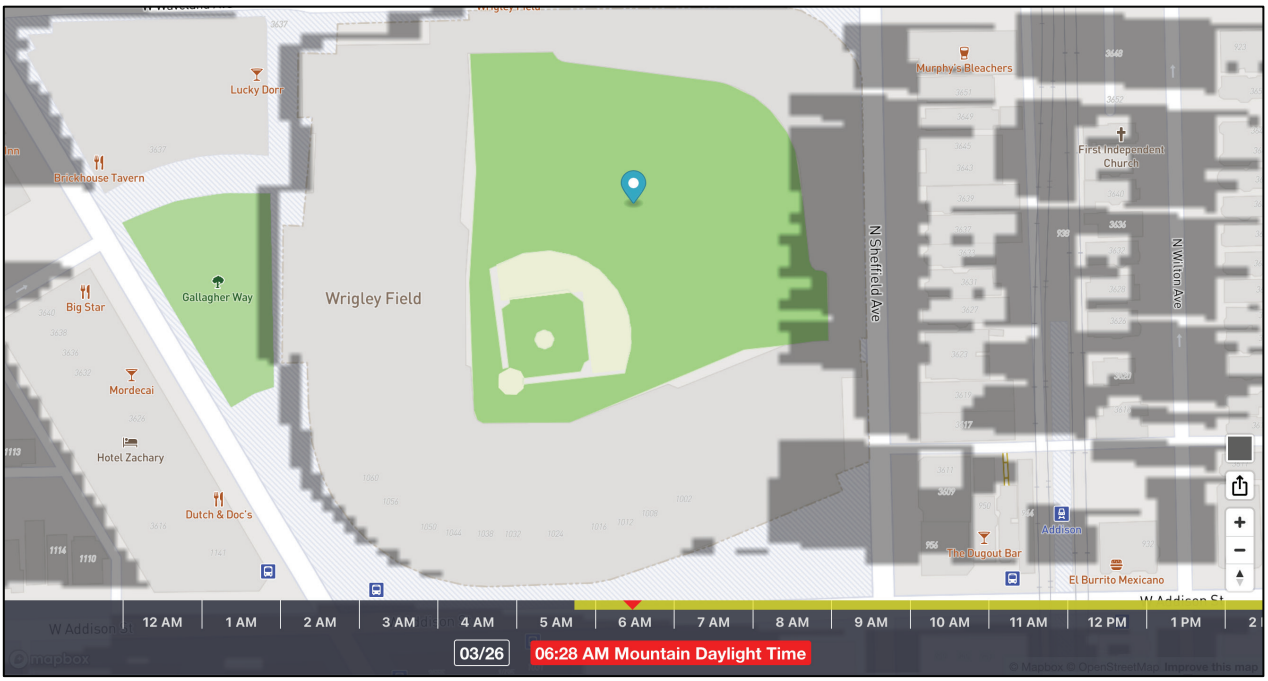


Figure 19.37: A Shade Map result.

Scribble Maps (scribblemaps.com)

The default view of mapping services such as Google and Bing may be enough for your situation. Occasionally, you may want to modify or customize a map for your needs. Law enforcement may want to create a map to be used in a court case; a private investigator may want to customize a map to present to a client; or a security director may want to use this service to document the inappropriate Tweets that were found during the previous

instructions. Scribble Maps offers one of the easiest ways to create your own map and add any type of visual aids to the final product.

The default view of your new map at Scribble Maps will display the entire world and a menu of basic options. I close this menu by clicking the small "x" in the upper right corner. You can then manually zoom into an area of interest or type in an address in the location bar at the top of the map. This will present you with a manageable area of the map. The lower right corner will allow you to switch from a traditional map view to a satellite or hybrid view.

The menu at the top of the map will allow you to add shapes, lines, text, and images to your map. Practicing on this map can never be replaced with any instruction printed here. Mastering the basics of this application will make occasional use of it easy. Figure 19.38 displays a quick sample map that shows a title, a line, a marker, and graphics. The menu can be seen in the upper left portion. When finished, the "Menu" button will present many options to print, save, or export your map. I also highly recommend **Free Map Tools** (freemaptools.com). This service provides multiple advanced options such as mapping a radius around a point of interest.

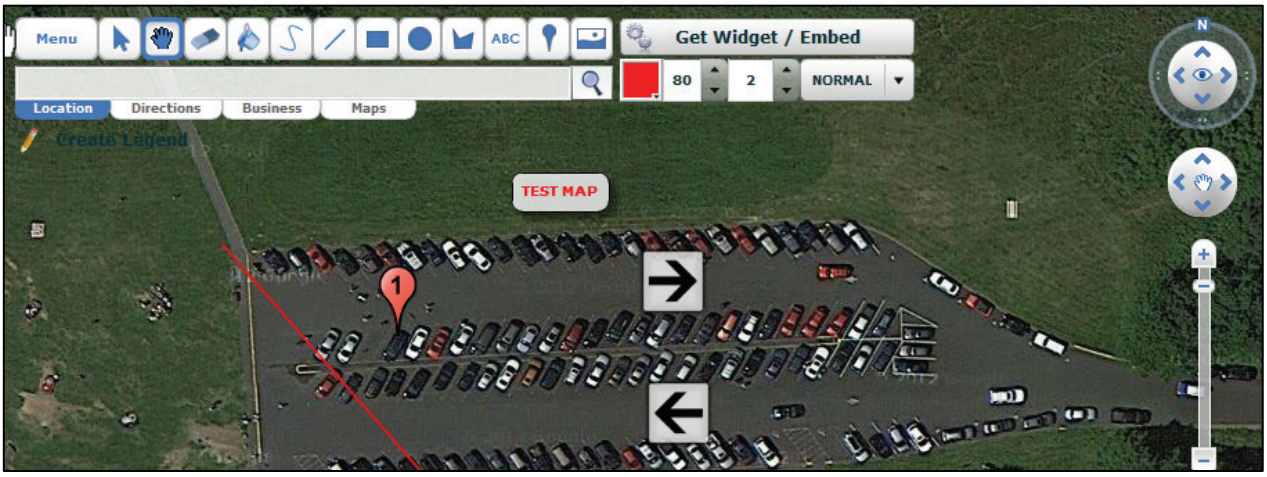


Figure 19.38: A basic custom map created with Scribble Maps.

CHAPTER TWENTY
DOCUMENTS

The open source intelligence discussed up to this point has focused on websites which include valuable information about a target. A category of intelligence that is often missed during OSINT research is documents. This type of data usually falls into one of three classes. The first is documents that include information about the target within the contents of the file. These can include online PDF files that the target may not know exist. The second class is documents that were actually created by the target. These files can make their way into public view unintentionally. Finally, the third class includes the metadata stored within a document that can include vital information about the true source of the document. The following techniques explain manual searching and retrieving of these documents and automated software solutions for analysis of found content.

Google Searching (google.com)

A very basic way of locating documents that are publicly available on a specific website, or related to a specific topic, is to use Google. The "filetype:" (or easier "ext:") search operators previously explained can be used for this task. An example of search queries for all Microsoft Word documents stored on the domain of inteltechniques.com would be the following.

site:inteltechniques.com ext:doc
site:inteltechniques.com ext:docx

If you wanted to locate all documents that reference a specific topic, you can use the filetype operator without a specific website listed. An example of a search query for all Excel spreadsheets that contain the acronym OSINT would be the following.

ext:xls "OSINT"
ext:xlsx "OSINT"

If you wanted to search for a specific person's name within any spreadsheets, such as John Doe, you would type the following single query.

ext:xls OR ext:xlsx "John Doe"

The following table includes the most common document file types and the associated file extensions. As previously explained, both Google and Bing are capable of searching any file type regardless of the file association. Remember that Bing demands "filetype" while Google seems to prefer "ext". Please note that this is a partial list, and I identify new possibilities constantly.

Microsoft Word	DOC, DOCX
Microsoft Excel	XLS, XLSX, CSV
Microsoft PowerPoint	PPT, PPTX
Adobe Acrobat	PDF
Text File	TXT, RTF
Open Office	ODT, ODS, ODG, ODP
Word Perfect	WPD

If you wanted to search all of these file types at once, the following string in Google or Bing would find most documents on the topic of OSINT. You could change that term to anything else of interest.

"OSINT" filetype:pdf OR filetype:doc OR filetype:xls OR filetype:xlsx OR filetype:docx OR filetype:ppt OR filetype:pptx OR filetype:wpd OR filetype:txt

This query basically tells the search engine to look for any reference to the term OSINT inside of a PDF, Microsoft Word, and other documents, and display all of the results. The Google Custom Search Engine (CSE) described in Chapter Nine is a great resource for this exact type of search. However, I highly recommend having an understanding of the manual search process. It will give you much more control than any automated solution. Additionally, Google CSEs limit the number of results. Therefore, I no longer recommend exclusively relying on it for a Document search. It simply cannot compete with a properly structured Google or Bing query. The custom search tools presented at the end of the chapter will further simplify all of this.

Google Docs (docs.google.com)

The idea of storing user created documents on the internet is gaining a lot of popularity. Keeping these files "in the cloud" eliminates the need for personal storage on a device such as a CD or flash drive. In addition, storing files on the internet allows the author to access and edit them from any computer with an internet connection. A common use of these document-hosting sites is to store them only during the editing phase. Once the document is finished and no longer needed, the user may forget to remove it from public view. Google is one of the most popular document storage websites. It allows users to embed the stored documents into their own websites if desired. Searching the site is relatively easy.

Many Google Mail (Gmail) users take advantage of Google's free service for document storage called Google Docs or Google Drive. When a document is created, it is private by default and not visible to the public. However, when people want to share documents with friends or coworkers, the sharing properties must be changed. While it is possible to privately share files with individual Google users, many people find it easier to make the documents public. Most of these users probably assume that the files will not be seen by anyone other than the intended recipients. After all, who would go out searching for other people's documents? We will.

The Google Docs and Google Drive websites do not offer the option to search these public files, but you can do this using Google search. Now that Google allows search engines to index most of the public files, you should be able to find them with some specific search methods. The following search examples will explain a few of the options that would be conducted on google.com. The exact search is listed with the expected result. These should be used as a starting point for the many possibilities of document searching.

site:docs.google.com "resume" - 29,800 online resumes
site:docs.google.com "resume" "Williams" - 985 resumes with the name Williams
site:docs.google.com "Corey Trager" - 1 document (resume) belonging to the target
site:docs.google.com 865-274-2074 - 1 document containing the target number

Google categorizes the documents that are created by the user. The examples below identify searches that would display documents by type.

site:docs.google.com/presentation/d - 325,000 PowerPoint presentations
site:docs.google.com/drawings/d - 18,600 Google flowchart drawings
site:docs.google.com/file/d - 945,000 images, videos, PDF files, and documents
site:docs.google.com/folder/d - 4,000 collections of files inside folders
site:docs.google.com/open - 1,000,000 external documents, folders, and files

In 2013, Google began placing some user generated documents on the "drive.google.com" domain. Therefore, any search that you conduct with the method described previously should be repeated with "drive" in place of "docs". The previous search for the telephone number would be the following.

site:drive.google.com 865-274-2074

Microsoft Docs (docs.microsoft.com)

Similar to Google Drive, Microsoft Docs offers that ability to store and share documents. The service is not as popular as Google Drive. However, there are thousands of publicly visible documents waiting to be found. The shared files are stored on the docs.microsoft.com domain. A query for resumes would be as follows. This search could be conducted on Google or Bing. The result on Google was 63,400 resume files with personal information.

site:docs.microsoft.com "resume"

Amazon Web Services (amazonaws.com)

Amazon Web Services (AWS) is a large collection of servers that supply storage and internet application hosting in "the cloud". Instead of purchasing expensive hardware, many companies and individuals rent these servers. There are numerous documents available for download from these servers when searched appropriately. I cannot overstate the value of searching Amazon's servers. This is where most of the voter data that was heavily discussed during the 2016 election originated. I have personally located extremely sensitive documents from this source on numerous occasions. The following structure will identify files indexed on google.com.

site:amazonaws.com

The following search examples will explain a couple of the options. The exact search is listed with the expected result. These should be used as a starting point for the many possibilities of document searching.

site:amazonaws.com ext:xls "password" - 114 Excel spreadsheets containing "password"
site:amazonaws.com ext:pdf "osint" - 260 PDF files containing "osint"

Another option is the Amazon CloudFront servers. CloudFront is a content delivery network (CDN) offered by Amazon Web Services. Content delivery networks provide a globally-distributed network of proxy servers which cache content, such as web videos or other bulky media. These are provided more locally to consumers, thus improving access speed for downloading the content. We can apply the same previous search techniques on this domain. The following search on Google yielded 129 results of pages on various CloudFront servers containing the acronym "OSINT".

site:cloudfront.net OSINT

Gray Hat Warfare (buckets.grayhatwarfare.com)

AWS hosts more than simple documents. Many companies host large databases and data sets within "buckets" on AWS servers. Custom Google searches may locate some of this content, but never a full index of all data. This is where Gray Hat Warfare excels. It has created a searchable database of over one billion files, all publicly stored with AWS servers. Free users are limited to the first 350 million of these files, which is substantial. A search of "OSINT" revealed only 11 results while "password" displayed over 41,000. The results are often large files which must be opened carefully. A search of "password xls" provided three documents with active credentials. A direct search URL is as follows.

https://buckets.grayhatwarfare.com/results/password

Google Cloud Storage (cloud.google.com)

This is Google's response to Amazon's AWS. It is a premium file storage web service for storing and accessing data on the Google Cloud Platform infrastructure. It is heavily used by all types of businesses and tech-savvy individuals. The number of publicly available sensitive documents is growing at an alarming rate. Below are a few examples.

site:storage.googleapis.com ext:xlsx OR ext:xls - 2,310 Spreadsheets
site:storage.googleapis.com "confidential" - 9,502 Documents
site:storage.googleapis.com "confidential" ext:pptx - 11 PowerPoint files marked as confidential

Presentation Repositories

With unprecedented online storage space at all of our fingertips, many people choose to store PowerPoint and other types of presentations in the cloud. Several free services have appeared to fill this demand. Of those, the following have the majority of publicly available documents.

Slide Share (slideshare.net)
ISSUU (issuu.com)
Slidebean (slidebean.com)
Prezi (prezi.com)
PPT Search (pptsearchengine.net)
Power Show (powershow.com)

Slide Share and ISSUU allow native searching within their websites. However, Prezi does not have this option. For all, I recommend a custom Google search with the site operator. If I want to locate presentations including the term OSINT from Prezi, I would use the following query.

site:prezi.com "osint"

Scribd (scribd.com)

Scribd was a leading cloud storage document service for several years. Since 2014, it has shifted its focus toward e-book sales. However, the plethora of stored documents is still accessible. This can be valuable for historical content posted, and likely forgotten, by the target. A search field is at the top of every page on the site within their collapsible menu. Searching for your target name should produce any public books stored through this service that includes the target name on any page of the publication. Clicking "Documents" in the menu will present more relevant information.

Most of these documents are intentionally stored on the site and any evidence of criminal activity will not be included. Instead, the primary use of the site for OSINT investigations is the large number of documents related to businesses. Entering any large corporation name should display several pages of viewable documents related to the company. Often, these include documents that the company's security personnel would not authorize to be online. Searching for "FOUO", an acronym for "for official use only", produced hundreds of results. While none of these appeared to be officially classified, they were not intended to be posted to a public website. If you are presented with an unmanageable number of results, the filter options appear directly above the first document result. These will allow you to search by language, size, file type, and date uploaded.

Identifying the user that uploaded a document is as easy as locating the document. In the upper-center of any page containing a document, there is an area that will identify the subject that uploaded the file. This also acts as a link to this user's profile on the website. The profile will display any information that the user supplied as well as a feed of recent activity of that user on the site. This can help identify other documents uploaded by a specific user.

PDF Drive (pdfdrive.com)

This service scans the internet for new PDF files and archives them. This can be helpful when the original source removes the content. The following static URL conducts a query across the entire domain.

https://www.pdfdrive.com/search?q=osint

WikiLeaks (search.wikileaks.org)

Some websites are created for the sole purpose of leaking sensitive and classified documents to the public. Wikileaks is such a site. When an Army soldier named Bradley Manning was arrested in 2010 for uploading classified government information to the site, Wikileaks became a household name. People then began to flock to the site to catch a glimpse of these controversial documents and videos. The official Wikileaks site finally provides a working search option. It will allow you to enter any search terms and will provide results of any leaked documents that contain these terms. Both the government and the private sector should be familiar with this site and the information that is identified with their respective agency.

Cryptome (cryptome.org)

Another site that strives to release sensitive and classified information to the public is Cryptome. Most of the information is related to freedom of speech, cryptography, spying, and surveillance. Much of the content could be considered conspiracy theories, but several official documents get released daily. Cryptome does not provide a search for their site and there are no third-party providers that cater to this service. Therefore, we must rely on Google or Bing to find the documents. A structured query of "osint" should function well. This technique using the search terms of "bradley manning" linked to 77 documents surrounding his investigation.

site:cryptome.org "osint"

Paste Sites

Paste Sites are not technically storage services for documents. They are websites that allow users to upload text for public viewing. These were originally designed for software programmers that needed a place to store large amounts of text. A link would be created to the text and the user could share the link with other programmers to review the code. This is still a common practice, but other users have found ways to abuse this technology. Many hacking groups will use this area of the internet to store compromised account information, user passwords, credit card numbers, and other sensitive content. There are dozens of sites that cater to this need, and very few of them have a search feature.

Pastebin (pastebin.com)

Pastebin is the most popular paste site in the United States. Criminal hacker groups often use this site to release illegally obtained data to the public. A previous release included the home addresses and personal information of many police officers near Ferguson, Missouri during protests and riots. More recently, stolen bank records and credentials from Venezuela were posted with encouragement to infiltrate the company. This is one of the sites that will allow for a search from within the site. This function performs a search through Google in the same way we could with the "site" operator. Typing in a target name, email address, or business name may reveal private information not intended for the public. For law enforcement, typing in the last four digits of a stolen credit card number may identify a link to the thief. If successful, the target is most likely outside of the country. Regardless, this is a valuable piece to the case and an impressive explanation to the victim. Unfortunately, most of the users leave a default username of "Guest". Pastebin no longer allows a direct URL search, but relies on Google for indexing, as follows.

site:pastebin.com "osint"

New paste sites come and go monthly. There is no way to present a current and complete list. However, I will focus on the most stable and prominent options which allow search. In a moment, I present a custom search tool which queries all of these at once. The following sites can be individually queried with the site operator, such as site:doxbin.org "osint".

0bin.net
cl1p.net
codepad.org
controlc.com
doxbin.org
dpaste.com
dpaste.de
dpaste.org
dumpz.org
friendpaste.com
gist.github.com
hastebin.com
heypasteit.com
ideone.com
ivpaste.com
jsbin.com
justpaste.it
justpaste.me
paste.debian.net
paste.ee
paste.centos.org
paste.frubar.net
paste.lisp.org
paste.opensuse.org
paste.org
paste.org.ru
paste.ubuntu.com
paste2.org
pastebin.ca
pastebin.com
pastebin.fr
pastebin.gr
pastefs.com
pastehtml.com
pastelink.net
pastie.org
p.ip.fi
privatebin.net
slexy.org
snipplr.com
sprunge.us
textsnip.com
tidypub.org
wordle.net
zerobin.net

Document Metadata

When an original document is found online, it is obviously important to analyze the visible content of the file. This includes the file name, written text, and an original location of the document. Digging deeper will expose more information. There is data embedded inside the document that cannot be seen by simply looking at the content of the file. This data is called metadata and can be very valuable to any type of investigation. It can often include the computer name on which the document was created, the username of the computer or the network, the software version used, and information about the network to which the computer is connected. The best way to view all of this information is to use software-based metadata viewers, but you can also view this "hidden" information online through a web browser.

Several online sites will allow you to upload documents for analysis. To do this, click the "browse" button on the pages detailed below. This will enable a file explorer that will allow you to select the document that you want analyzed. The result often identifies a created and modified date, the original title, three applications used to create the document, and a username. A further search of this username through the previously discussed techniques could produce a wealth of information about the author of the document. The following websites allow you to upload a locally stored document or submit a URL of a file for analysis. Please use caution with this technique. If the document is already posted online, there is very little risk of allowing a URL analysis. However, a locally stored file that has never been on the internet may require a second thought. If the content is sensitive, you may not want to upload to any service. If the file contains classified information, you could be jeopardizing your clearance. In these situations, use the methods discussed in a moment. If this is not a concern the following websites work well.

Extract Metadata (extractmetadata.com)
Jeffrey's Viewer (exif.regex.info/exif.cgi)
ExifInfo (exifinfo.org)
Get Metadata (get-metadata.com)

Document Metadata Applications

If I need to analyze the metadata stored within documents, I prefer to do so locally on my machine. I do not want to share any potential investigation details with an online resource. This is especially true if the documents are not already online. You may possess a folder of files which were retrieved from a suspect computer or emailed directly to you. In these scenarios, we should be cautious as to not distribute any evidence electronically to any websites. I present two solutions, with a third unstable option.

ExifTool (exiftool.org)

You already possess a document metadata viewer within your custom Linux virtual machine. It is called ExifTool and we installed it during the previous chapters. This is a terminal-based solution, but the function is quite

simple. Assume that you have used the previous techniques to download several Word documents in .docx file format onto your Desktop in a folder titled Evidence. The following steps within Terminal would navigate to the proper folder, generate a spreadsheet with the metadata included within the documents, and title it Report.csv on your Desktop in the Evidence folder.

```
cd ~/Desktop/Evidence
exiftool * -csv > ~/Desktop/Evidence/Report.csv
```

Let's conduct an example and take a look at the results. I performed the following Google search:

ext:docx "osint"

This provided 371 results, all of which were Microsoft Word documents. I downloaded the first four into the Evidence folder on my Desktop. After executing the commands above, I launched the spreadsheet. Figure 20.01 displays a small portion of the interesting results. This tells me the names of the individuals who created and last modified the documents; the companies involved; the software which was used; and even the amount of time they spent editing the content. This is extremely valuable information which should be collected during every investigation in which documents are obtained.

Application	AppVersion	Company	CreateDate	Creator	LastMo	By	TotalEditTime
Microsoft Office Word	12		2011:03:18 20:29:00Z	marko.prir	marko.(	ac	1.2 hours
Microsoft Macintosh Word	15		2016:04:25 14:56:00Z	Kirk Haye	Kirk Ha		26 minutes
Microsoft Office Word	12		2017:08:03 09:53:00Z	Hakon201	Hakon2		4 minutes
Microsoft Office Word	14	United States Army	2016:06:13 08:31:00Z	john.t.rich	IMO-P(	lin, Jonas A	3 hours

Figure 20.01: Document metadata results from ExifTool.

FOCA (github.com/ElevenPaths/FOCA)

You may desire a Windows-based solution which possesses a user-friendly interface. FOCA was once the premier document metadata collection and extraction tool. It was created to search, download, and analyze documents and their metadata in one execution. Unfortunately, Google and other search engines began blocking the search and download behavior of the software. Fortunately, the analysis portion still works perfectly. The following steps will download and install FOCA to your Windows VM or any other Windows device.

- Navigate to https://github.com/ElevenPaths/FOCA/releases.
- Click the hyperlink for the most recent "zip" file.
- Double-click the .zip file and extract the contents.
- Launch FOCA.exe from the "bin" folder within the "FOCAPro" folder.
- If prompted, select "Download and install this feature" to install the .net framework.

FOCA should launch and present a window with many options. Today, the vast majority of the features no longer work, but the document analysis is helpful. Assume you possess the same four documents mentioned previously on the Desktop of your Windows VM.

- Open FOCA and click the Metadata folder in the left menu.
- Drag and drop the documents into the FOCA window.
- Right-click any of the documents and choose "Extract all metadata".
- Right-click any of the documents and choose "Analyze metadata".

You can now click through the menus on the left to view any metadata details such as email addresses, names, and computers associated with these files. The benefit of this method is the user interface, but you sacrifice a reporting option. The previous ExifTool method is not as pretty, but the spreadsheet result is helpful. Ultimately, I believe you should be familiar with both options and rely on the method most appropriate for you.

Real World Application: Dennis Lynn Rader, also known as the BTK killer, sent a floppy disk to the Wichita Police Department containing a Microsoft Word document in reference to his killings. The police examined the metadata of this document and determined that it was made by a subject named "Dennis". Links to a Lutheran church were also located within this data. Conducting OSINT searches on these two pieces of information helped to identify the suspect and make an arrest.

Manual Metadata Extraction

There are some documents which store much more metadata within the content than what is presented within the official metadata associated with the file. I see this most commonly with PowerPoint files. Let's conduct a demonstration. I searched ext:pptx "osint" within Google. The first result was a PowerPoint presentation, which I downloaded. The previous methods announced all of the standard metadata we would expect to see. I then changed the name of the PowerPoint presentation file from "OpenSourceIntelligence-OSINT.pptx" to "OpenSourceIntelligence-OSINT.zip". This tells my computer that this file is now an archive.

I then decompressed the zip file which presented dozens of new files. These are the content behind the scenes of the PowerPoint itself. They include all of the images inside the presentation, which can then easily be analyzed for their own metadata, and text extraction of all words in the slides. The "app.xml" file confirms that the author was using PowerPoint from Microsoft Office 2016 (AppVersion>16.0000) and several files include unique identifiers for this user. Comparing these to other downloaded documents could prove that the authors of each were the same.

In a bit of irony, this PowerPoint file randomly selected from the Google results included the OSINT flowcharts which I present at the end of my books, but the owner claimed them as his own work with "Copyright of the Cybersecmentorship.org". Apparently, this technique can also help expose copyright infringement and blatant plagiarism.

Free OCR (free-ocr.com)

You may occasionally locate a PDF file that has not been indexed for the text content. These types of PDF files will not allow you to copy and paste any of the text. This could be due to poor scanning techniques or to purposely prohibit outside use of the content. You may desire to capture this text for a summary report. These files can be uploaded to Free OCR and converted to text documents. OCR is an acronym for optical character recognition. Basically, a computer "reads" the document and determines what the text is inside the content. The result is a new document with copy and paste capability.

Text Archives (archive.org)

The Internet Archive possesses massive collections of text files, books, and other documents. These files can be accessed with the "=TXT" parameter after a search. A direct URL for documents mentioning "inteltechniques" within the title or content would be the following.

https://archive.org/search.php?query=inteltechniques&sin=TXT

Google Books (books.google.com)

Google has scanned most books printed within the past decade and the index is searchable. Many of these scans also include a digital preview of the content. A direct URL query appears as follows.

https://www.google.com/search?tbm=bks&q=inteltechniques

Pirated Books (annas-archive.org)

I am hesitant to present this resource, but I have found it extremely valuable in my investigations. Many people use a service called Library Genesis to download illegal pirated e-books. Library Genesis does not offer any type of indexing of content, but Annas Archive acquires all of their content and provides a search interface. This allows us to search within the content of millions of pirated books without actually downloading any content and allows us to find references to specific targets within publications. While it may be tempting to download the PDF of the book, please don't. This likely violates your laws, policies, or ethics. The following URL would query "michael bazzell" within the entire collection.

https://annas-archive.org/search?q=michael%20bazzell

Book Sales (amazon-asin.com)

I have never had an investigation rely on book sale information, but I find it interesting. Amazon offers a lookup tool which displays details about the number of copies being sold of any book on their site. First, you must identify the ASIN assigned to the book. This is typically visible within the details section of the book's listing page. The previous edition of this book is B09PHL6Q4G. The following URL accesses all available details.

https://amazon-asin.com/asincheck/?product_id=B09PHL6Q4G

By visiting this page, you can access the fee I pay to Amazon for each sale ($7.90); the fee I pay to Amazon for fulfillment ($5.84); the fee I pay to Amazon for royalties (40%); the "Listing Quality" (Poor); Potential sales (Low); Estimated Daily Sales (11-13 books); Estimated Daily Income ($447.00 - $528.00); Main Search Keywords (open source intelligence techniques); and Amazon Buyer Keywords (100 techniques americas test kitchen). As you can see with that last entry, these results are not all accurate. Well, maybe the Potential Sales entry (Low). I never believe everything I see here, but it can be an indicator of sales versus a competing book.

Rental Vehicle Records

The details of rental vehicles are not technically documents, but the data seemed to fit this category the best. The following options have been controversially received during training and may not be appropriate for everyone. I present these methods to you as theories, and you should evaluate if the techniques are suitable for your research. Several vehicle rental companies offer an option to access your receipts online. This is probably designed for customers that leave a vehicle at the business after hours and later need a receipt. While the processes to retrieve these documents are designed to only obtain your own records, it is easy to view others.

Enterprise (enterprise.com)

At the bottom of every Enterprise web page is an option to "Get a receipt". Clicking this will present a form that must be completed before display of any details. Enterprise will need the driver's license number and last name. Providing this information will display the user's entire rental history for the past six months to three years. Testing with my own data provided two years' worth of results. Each document will link to the entire receipt from that rental. These receipts include the start date and time, end date and time, vehicle make and model, pick up location, total mileage, lease name, and form of payment. This information could be very beneficial to any drug case or private investigation.

Hertz (hertz.com/rentacar/receipts/request-receipts.do)

Similar to Enterprise, Hertz has a link at the bottom of this page titled "Get a receipt". You can search by driver's license number or credit card number with last name. The receipt will be similar to the Enterprise demonstration.

Alamo (alamo.com)

Alamo titles their receipt retrieval link "Past Trips/Receipts" and it is located in the bottom portion of every page. The process is identical to the previous two examples. The only difference is that you must choose a date range. I usually select a start date of one year prior to the current date and the end date of the current date.

Thrifty (thrifty.com/Reservations/OnlineReceipts.aspx)

This service requires a last name and either a driver's license number or credit card number.

Dollar (dollar.com/Reservations/Receipt.aspx)

Similar to Thrifty, this service requires a last name and either a driver's license number or credit card number.

IntelTechniques Documents and Pastes Tools

If these resources seem overwhelming, consider my custom document search tools. The first tool, available online at **https://inteltechniques.com/tools/Documents.html**, or the file titled "Documents.html" in your download from previous chapters, has two sections. The first section queries documents by file types. It allows entry of any terms or operators in order to locate PDF, DOC, DOCX, XLS, XLSX, CSV, PPT, PPTX, KEYNOTE, TXT, RTF, XML, ODT, ODS, ODP, ODG, ZIP, RAR, 7Z, JPG, JPEG, PNG, MPG, MP4, MP3, and WAV documents. The "Submit All" option will execute each search in its own tab. The second section allows entry of any terms or operators in order to locate files stored within Google Docs, Google Drive, Microsoft Docs, Amazon AWS, CloudFront, SlideShare, Prezi, ISSUU, Scribd, PDF Drive, and others. The "Submit All" option will execute each search in its own tab. The "Pastes" search tool presents a Google custom search engine (CSE) which queries all paste sites mentioned previously. In this example, I have searched the term "osint" and received over 2,000,000 results within the search tool.

CHAPTER TWENTY-ONE
IMAGES

Thanks to cameras on every data cellular phone, digital photograph uploads are extremely common among social network users. These images can create a whole new element to the art of open source intelligence analysis. This chapter will identify various photo sharing websites as well as specific search techniques. Later, photo metadata will be explained that can uncover a new level of information including the location where the picture was taken; the make, model, and serial number of the camera; original uncropped views of the photos; and even a collection of other photos online taken with the same camera. After reading this information, you should question if your online photos should stay online.

Google Images (images.google.com)

During my live training sessions, I always encourage attendees to avoid individual searches on various photo sharing websites such as Flickr or Tumblr. This is because most of these searchable sites have already been indexed by Google and other search engines. Conducting a search for "Oakland Protest" on Flickr will only identify images on that specific service that match. However, conducting the same search on Google Images will identify photos that match the terms on Flickr and hundreds of additional services. Similar to Google's standard search results, you can use the Search Tools to filter by date. Additionally, you can further isolate target images by size, color, and type, such as photographs versus line drawings. I no longer conduct manual searches across the numerous photo sharing sites. Instead, I start with Google Images.

Bing Images (bing.com/images)

Similar to Google, Bing also offers an image search. While it is not as beneficial as the Google option, it should never be overlooked. On several occasions, I have located valuable pictorial evidence on Bing that was missing from Google results. The function is identical, and you can filter search results by date, size, color, type, and license type. When searching for relevant data about a target, I try to avoid any filters unless absolutely necessary. In general, we always want more data, not less. The search techniques explained in Chapter Nine all apply to queries on Google Images and Bing Images.

Reverse Image Search

Advancements in computer processing power and image analysis software have made reverse image searching possible on several sites. While a standard search online involves entering text into a search engine for related results, a reverse image search provides an image to a search engine for analysis. The results will vary depending on the site used. Some will identify identical images that appear on other websites. This can be used to identify other websites on which the target used the same image. If you have a photo of a target on a social network, a reverse analysis of that photo may provide other websites on which the target used the same image. These may be results that were not identified through a standard search engine. Occasionally, a target may create a website as an alias, but use an actual photo of himself. Unless you knew the alias name, you would never find the site. Searching for the site by the image may be the only way to locate the profile of the alias. Some reverse image sites go further and try to identify other photos of the target that are similar enough to be matched. Some will even try to determine the sex and age of the subject in the photo based on the analysis of the image. This type of analysis was once limited to expensive private solutions. Now, these services are free to the public.

Google Reverse Image Search (images.google.com)

One of the more powerful reverse image search services is through Google. Rolled out in 2011, this service, now called Google Lens, is often overlooked. On any Google Images page, there is a search field. Inside of this field on the far right is a light grey camera icon that appears slightly transparent. Figure 21.03 (first) displays this

search field. Clicking on this icon will open a new search window that will allow for either a web address of an online image, or an upload of an image file on your computer. In order to take advantage of the online search, you must have the exact link to the actual photo online. Locating an image within a website is not enough. You will want to see the image in a web browser by itself, and then copy the address of the image. If I want to view the image from the actual location, I must right-click on the image and select "view image" with my Firefox browser. Chrome users will see "open image in new tab" and Internet Explorer users will see "properties" which will identify the URL of the image. This link is what you want in order to conduct a reverse image analysis.

In late 2022, I witnessed many changes to Google's reverse image search process. Today, if you paste a URL into Google's reverse image search, the default result will be "Visual matches" within "Google Lens", which can be seen in Figure 21.01. These are typically images similar to the target, but not a replica of the target. Clicking the "Find image source" button takes you to a new tab with results matching your target image. Adding text context to the search field in this new tab can improve accuracy. As an example, a reverse-search of a photo from LinkedIn might produce many inaccurate results, but including the name or employer of your target will often display only applicable evidence.

Another way to use this service is to search for target text details within the Google Images search page at images.google.com. The images in the results will present additional options when clicked. A larger version of the image will load inside a black box. The options surrounding the image will allow you to visit the page where the image is stored; view the image in full size; crop a portion of the image; or cycle through similar images.

I miss the way Google handled these queries before Google Lens, but we are stuck with whatever they provide us. Let's focus on each feature and identify exact URL submission options. I will start with the following image of a friend of mine (sorry Marcos), which can be seen in Figure 21.01 (left). The original URL follows.

https://s.hdnux.com/photos/01/30/23/25/23147846/3/1200x0.jpg

I can submit this URL directly to Google Lens with the following URL.

https://lens.google.com/uploadbyurl?url=https://s.hdnux.com/photos/01/30/23/25/23147846/3/1200x0.jpg

If I have a name or keyword, and only want to search images through Google, the following URL applies

https://www.google.com/search?tbm=isch&q=Marcos+Pulido

Once within the Google Lens page, you can navigate to the image source or manipulate the current target photo. In this example (Figure 21.01), I cropped the image using the embedded frame, and focused on the unknown person in the lower-left of the image. The result can be seen in Figure 21.02. On occasion, this has been helpful in identifying other people or objects within an image. It also provides an option to extract text from a photo, but I have not needed that yet.

Bing Reverse Image Match (bing.com/images)

In 2014, Bing launched its own reverse image search option titled "Visual Search". This feature can be launched from within any page on Bing Images by clicking the camera icon to the right of the search field. Figure 21.03 (second) displays this option. This service does not seem to be as robust as Google's. In my experience, I often receive either much fewer results, although they do match. On a few occasions, I have received matched images that Google did not locate.

TinEye (tineye.com)

TinEye is another site that will perform a reverse image analysis. These results tend to focus on exact duplicate images. The results here are usually fewer than those found with Google. Since each service often finds images

the others do not, all should be searched when using this technique. Figure 21.03 (third) displays the search menu. The icon on the left prompts the user to provide a location on the hard drive for image upload while the search field will accept a URL.

Yandex Images (yandex.ru/images)

Russian search site Yandex has an image search option that can conduct a reverse image search. Similar to the other methods, enter the full address of the online image of interest and search for duplicate images on additional websites. In 2015, Yandex began allowing users to upload an image from their computers. In 2020, I began noticing accurate results from Yandex which were not visible in results from Google, Bing, or TinEye. Today, Yandex may be your best reverse image search option. Figure 21.03 (fourth) displays the reverse image search icon in the far-right portion.

Baidu Images (image.baidu.com)

Similar to Yandex, the Chinese search engine Baidu offers a reverse image search. Baidu currently offers no English version of their website and only presents Chinese text. Navigating to the above website offers a search box that contains a small camera icon to the right. Clicking this presents options for uploading an image (button to left) or providing the URL of an online image within the search field itself. The results will identify similar images on websites indexed by Baidu. Figure 21.03 (fifth) displays the search page only available in Chinese. In my experience, this reverse search option fails more than it functions, and it has been removed from the custom search tools explained in a moment.

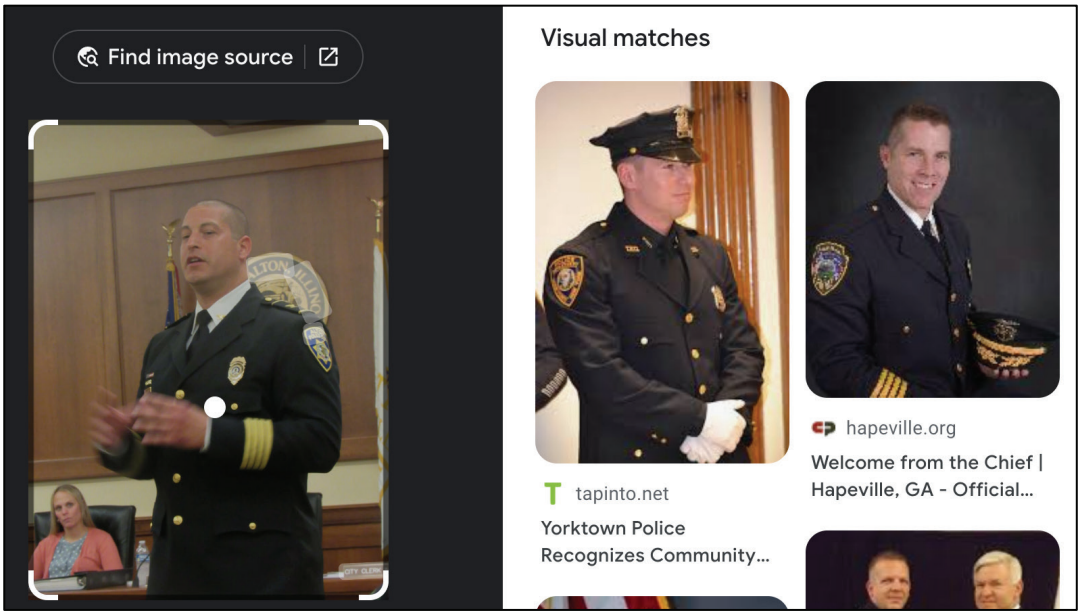


Figure 21.01: A Google Lens search via URL.

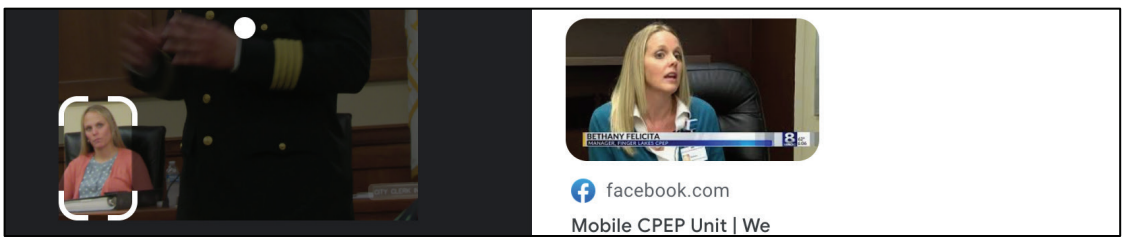


Figure 21.02: A Google Lens focused search result.

Regardless of the services that you are executing, I urge you to use caution with sensitive images. Similar to my view of analyzing online documents for metadata, I believe that submitting online photos within these engines is harmless. If the photo is already publicly online, there is very little risk exposing it a second time. My concern involves child pornography and classified photos. As a former child pornography investigator and forensic examiner, there were several times that I wanted to look for additional copies of evidence online. However, I could not. Even though no one would know, and the photos would never appear any place they should not, conducting reverse image searches of contraband is illegal. It is technically distributing child pornography (to Google). While working with a large FBI terrorism investigation, I had possession of ten photos on which I wanted to conduct a reverse image search. The photos were part of a classified case, so I could not. Overall, never submit these types of photos from your hard drive. It will always come back to haunt you.

Whenever I have any public images that call for reverse image searching, I always check all five of these services. While I rarely ever get a unique result on Baidu, it only takes a few seconds to check every time. This diligence has paid off in the past. These manual searches do not need to be as time consuming as one may think. We can automate much of this process to save time and encourage thorough investigations. First, we should take a look at direct URL submission. For the following examples, assume that your target image is the cover of my privacy book from the web page at inteltechniques.com/book7.html. The actual target image is stored online at the URL of https://inteltechniques.com/img/EP4-3D.png. The following direct addresses would conduct a reverse image search at each service listed.

Google: https://lens.google.com/uploadbyurl?url=https://inteltechniques.com/img/EP4-3D.png

Bing: https://www.bing.com/images/search?view=detailv2&iss=sbi&q=imgurl:https://inteltechniques.com/img/EP4-3D.png

TinEye: http://www.tineye.com/search/?url=https://inteltechniques.com/img/EP4-3D.png

Yandex: https://yandex.com/images/search?rpt=imageview&url=https://inteltechniques.com/img/EP4-3D.png

Baidu: https://graph.baidu.com/upload?image=https%3A%2F%2Finteltechniques.com%2Fimg%2FEP4-3D.png

Figure 21.03: Reverse image search options from Google, Bing, TinEye, Yandex, and Baidu.

Cropped Reverse Image Searching

Beginning in 2018, I noticed that both Google Images and Bing Images were returning fewer results than in previous years. It seemed as though each were trying to limit the number of matching photos, possibly with the intent to present only relevant images based on previous search history or whatever they "think" we want from them. In 2019, I had an investigation focused around an online image. When I conducted a reverse image search, I received one result, which was a copy I already knew existed. When I cropped the image to only display my target, I received more search results. I find this technique applicable to Google and Bing, but I believe it works best with Yandex. The following is a demonstration of this method, using a public image posted to Twitter.

Figure 21.04 (left) is the original image obtained from Twitter. A reverse image search through Google, Bing, and Yandex revealed numerous results, but none of them contained my target displayed on the far left. I cropped the image, as seen in Figure 21.04 (right), to display only the target and resubmitted to Yandex. This immediately identified numerous images of the target. Figure 21.05 displays one of these images. Both Google and Bing displayed no results from this cropped image. I cannot stress enough the importance of reverse-searching images through Yandex. I find their service superior to all others.

Figure: 21.04: An original image (left) and cropped version (right).

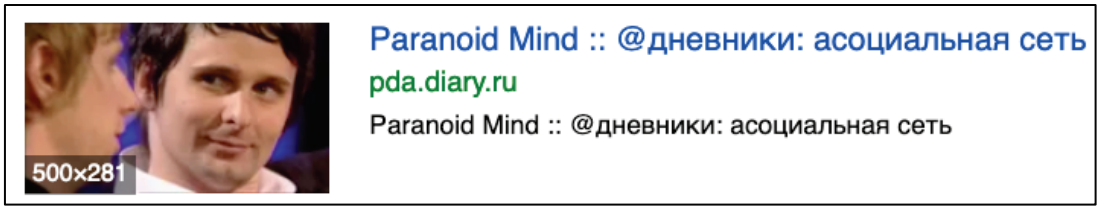


Figure 21.05: A reverse image search result from Yandex.

FaceCheck (facecheck.id)

While the major search engines will find most generic images, there are two services which stand out for faces and vehicles. I will explain each. FaceCheck surfaced in 2022, and is currently a free alternative to the paid service Pimeyes. It applies artificial intelligence to compare an online or uploaded image to a database of billions of photos which were previously shared online. This is not exact pixel comparison and the matching images do not need to be from the same place or time. This is a series of servers which analyze your target image and attempt to identify any stored images of the same person. The results are creepy to say the least. The online interface encourages you to upload an image, which I am cautious to do. This will likely add it to their collection. Instead, I only submit images which are already online and have likely already been indexed by this service. Let's revisit the following image from the Google Lens demonstration.

https://s.hdnux.com/photos/01/30/23/25/23147846/3/1200x0.jpg

The following URL submits this image, which is a feature otherwise hidden from their service.

https://facecheck.id/#url=https://s.hdnux.com/photos/01/30/23/25/23147846/3/1200x0.jpg

This populated the image and began the analysis, as seen in Figure 21.06. Notice that it automatically determined the face within the image and cropped out any unhelpful details. The final result was twelve images of my target, all of which were captured on various dates with different clothing and background, as partially seen in Figure 21.07. I can click each to access the original location and details. This has uncovered the true identities of countless people associated with my investigations. I suspect this will become a paid service someday, so take advantage of the free version while it lasts.

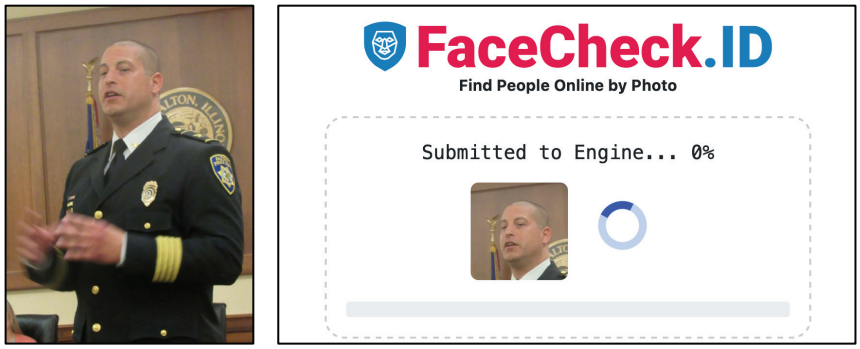


Figure 21.06: A target image and analysis by FaceCheck.

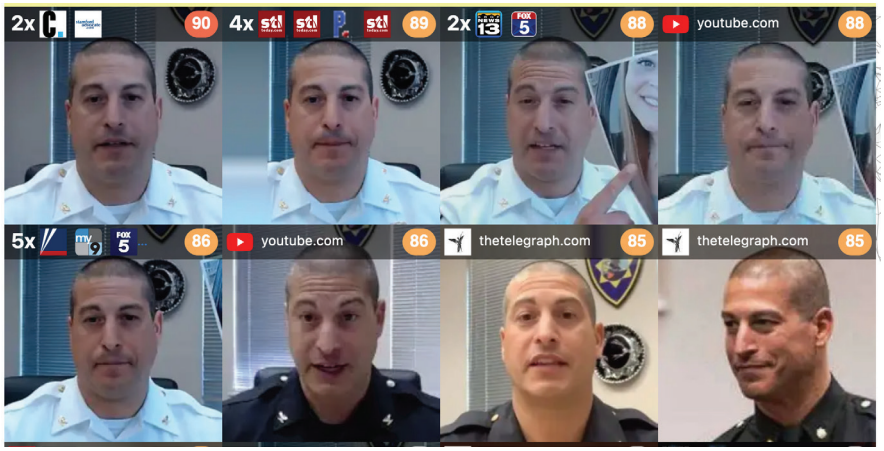


Figure 21.07: FaceCheck results identifying additional images of a target.

CarNet (carnet.ai)

This is another service which appeared in 2022, and has been extremely beneficial to my investigations. It allows you to upload an image, or provide a URL, of a suspect vehicle, and it will use machine learning to identify the year, make, and model when possible. Let's conduct a demonstration. As I wrote this, someone posted on Reddit asking for assistance identifying the vehicle within the following link, as seen in Figure 21.08 (left).

https://i.redd.it/xf4m1q7t6d0a1.jpg

I uploaded the blurry image into CarNet, which immediately identified it as a 2013-2017 4th generation Holden/Commodore, grey in color. That didn't sound familiar to me, so I assumed it was wrong. Someone much more knowledgeable about vehicles was credited on Reddit as posting the correct answer, a rare "Chevrolet SS built off the Australian Holden Commodore", as seen in Figure 21.08 (right). I have found this service extremely valuable when trying to identify vehicles captured on video doorbells and dash cameras.

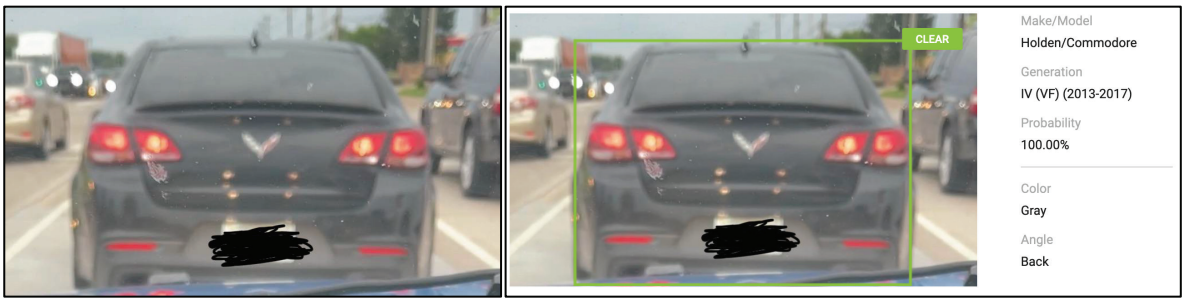


Figure 21.08: An image of an unknown vehicle.

Reddit Repost Sleuth (repostsleuth.com)

This service is a reverse image search engine that only provides positive results that appear on the website Reddit. It was originally launched as a way for users to identify when someone reposted a photo that had previously been posted on the website. The user could then "down-vote" the submission and have it removed from the front page. We can use this in investigations to locate every copy of an individual photo on Reddit. You can either provide a link to an image or upload an image from your computer. The following static URL submits our target image for reverse analysis on Reddit.

https://www.repostsleuth.com/search?url= https://inteltechniques.com/img/EP2.png

Root About (rootabout.com)

This is another specialized reverse image search utility which only queries against images stored on the Internet Archive and within Open Library (also an Internet Archive product). Any results will likely contain public images such as published photos and book covers. I have yet to receive any benefit to my investigations with this service, but it should still be available in your arsenal of tools. Root About does not support search via a direct URL.

Wolfram Image Identification Project (imageidentify.com)

While this is not a traditional reverse image search, it does provide value. The goal of this service is to identify the content of an image. If you upload a photo of a car, it will likely tell you the make, year, and model. An upload of an image containing an unknown Chinese word may display a translation and history details. The site prompts you to upload a digital file, but you can also drag and drop an image from a web page in another tab.

Real World Application: These reverse image search sites can have many uses to the investigator. In 2011, I searched a photo of damage to a popular historic cemetery that was vandalized. The results included a similar photo of the suspect showing off the damage on a blog. An arrest and community service soon followed. Later, while working with a private investigator, I was asked to locate any hotels that were using the client's hotel images on websites. A reverse image search identified dozens of companies using licensed photos without authorization. This likely led to civil litigation. More recently, a federal agent asked me to assist with a human trafficking case. He had a woman in custody who spoke little English. She was arrested during a prostitution sting and was suspected of being a victim of trafficking. A reverse image search from one online prostitution ad located all of her other ads which identified the regional areas that she had recently been working, a cellular telephone number connected to her pimp, and approximate dates of all activity.

Pictriev (pictriev.com)

Pictriev is a service that will analyze a photo including a human face and try to locate additional images of the person. The results are best when the image is of a public figure with a large internet presence, but it will work on lesser-known subjects as well. An additional feature is a prediction of the sex of the target as well as age.

Twitter Images

For the first several years of Twitter's existence, it did not host any photos on its servers. If a user wanted to attach a photo to his or her post, a third-party photo host was required. These have always been free and plentiful. Often, a shortened link was added to the message, which forwarded to the location of the photo. Twitter now hosts photos used in Twitter posts, but third-party hosts are still widely used. The majority of the images will be hosted on Instagram, which was previously explained. If you have already identified your target's Twitter page, you will probably have the links you need to see the photos uploaded with his or her posts. Many Twitter messages have embedded images directly within the post. Twitter now allows you to search keywords for photo results. After you conduct any search within the native Twitter search field, your results will include a filter menu on the top. The "Photos" results will only include images which have a reference to the searched keyword within the message or hashtag. You can also filter this search for people, videos, or news. The URL query for all Twitter images associated with "osint" would be https://twitter.com/search?q=osint&f=image.

Facebook Images

Facebook also allows a query for images. The traditional method is to search a keyword and navigate to the Photos filter. However, we can replicate this via URL athttps://www.facebook.com/search/photos/?q=osint.

Tumblr Images

Tumblr blogs have a heavy emphasis on images, and search engines do not always index associated keywords well. Therefore, we should consider a query specifically on the site. The URL query for images associated with "osint" would be https://www.tumblr.com/search/osint. All of these options are included within the custom search tools which are presented at the end of this chapter.

Photo-Sharing Sites

In order to find a photo related to a target, the image must be stored on a website. The most common type of storage for online digital photos is on a photo-sharing site. These sites allow a user to upload photographs to an account or profile. These images can then be searched by anyone with an internet connection. Almost all of these hosts are free for the user and the files will remain on the site until a user removes them. There are dozens of these services, many allowing several gigabytes worth of storage. While I mentioned earlier that a Google Images or Bing Images search was most appropriate for all photo sharing hosts, Flickr deserves a mention.

Flickr (flickr.com)

Flickr, purchased by Yahoo and now owned by SmugMug, was one of the most popular photo-sharing sites on the internet. Many have abandoned it for Twitter and Instagram, but the mass number of images cannot be ignored. The majority of these images are uploaded by amateur photographers and contain little intelligence to an investigator. Yet there are still many images in this "haystack" that will prove to be beneficial to the online researcher. The main website allows for a general search by topic, location, username, real name, or keyword. This search term should be as specific as possible to avoid numerous results. An online username will often take you to that user's Flickr photo album. After you have found either an individual photo, user's photo album, or group of photos by interest, you can begin to analyze the profile data of your target. This may include a username, camera information, and interests. Clicking through the various photos may produce user comments, responses by other users, and location data about the photo. Dissecting and documenting this data can assist with future searches. The actual image of these photos may give all of the intelligence desired, but the data does not stop there. A search for photographs related to the Occupy Wall Street protesters returned over 157,000 results.

Flickr Map (flickr.com/map)

Flickr attempts to geo locate all of the photos that it can. It attempts to identify the location where the photo

was taken. It will usually obtain this information from the Exif data, which will be discussed in a moment. It can also tag these photos based on user provided information. Flickr provides a mapping feature that will attempt to populate a map based on your search parameters. I believe this service is only helpful when you are investigating a past incident at a large event or researching the physical layout of a popular attraction.

Flickr API

There are three specific uses of the Flickr Application Programming Interface (API) that I have found helpful during many online investigations. The first queries an email address and identifies any Flickr accounts associated with it. The second queries a username, and identifies the Flickr user number of the connected account. The final option queries a Flickr user number and identifies the attached username. Unfortunately, all of these features require a Flickr API key. I have included a throwaway key within the search tools explained at the end of the chapter. However, it may not function for long after the book is published. If the key should be terminated by Flickr, simply request your own free key at https://www.flickr.com/services/api/. Once issued, replace my test key (27c196593dad58382fc4912b00cf1194) within the code of the tools to your own. A demonstration may help to explain the features. First, I submitted the following URL to Flickr in order to query my target email address of test@test.com.

https://api.flickr.com/services/rest/?method=flickr.people.findByEmail&api_key=27c196593dad58382fc4912b00cf1194&find_email=test@test.com

I immediately received the following result.

User id="8104823@N02" username>intellectarsenal

I now know that my target possesses a Flickr account associated with the email address, the username for the account, and the unique user number which will never change. Next, assume that we only knew the username. The following URL could be submitted.

https://api.flickr.com/services/rest/?method=flickr.people.findByUsername&api_key=27c196593dad58382fc4912b00cf1194&username=intellectarsenal

The response includes the following.

User id="8104823@N02"

Once you have identified the user number, we can submit the following URL.

https://api.flickr.com/services/rest/?method=flickr.people.getInfo&api_key=27c196593dad58382fc4912b00cf1194&user_id=8104823@N02

This returns the most details, including the following result from our target.

username>intellectarsenal
photosurl>https://www.flickr.com/photos/8104823@N02/
profileurl>https://www.flickr.com/people/8104823@N02/
mobileurl>https://m.flickr.com/photostream.gne?id=8084475

Navigating to the profile displays details such as the join date, followers, and photo albums. This may seem like a lot of work for a minimal number of details, but this is quite beneficial. There is no native email address search on Flickr, but we can replicate the function within the API. You may not find young targets sharing images here, but the massive collection of photos spanning the past decade may present new evidence which was long forgotten by the target.

Exif Data

Every digital photograph captured with a digital camera possesses metadata known as Exif data. I have already explained several applications which extract this data from documents and images, but we need to have a better understanding of the technology. This is a layer of code that provides information about the photo and camera. All digital cameras write this data to each image, but the amount and type of data can vary. This data, which is embedded into each photo "behind the scenes", is not visible by viewing the captured image. You need an Exif reader, which can be found on websites and within applications. Keep in mind that some websites remove or "scrub" this data before storing it on their servers. Facebook, for example, removes the data while Flickr does not. Locating a digital photo online will not always present this data. If you locate an image that appears full size and uncompressed, you will likely still have the data intact. If the image has been compressed to a smaller file size, this data is often lost. Any images removed directly from a digital camera card will always have the data. This is one of the reasons you will always want to identify the largest version of an image when searching online. The quickest way to see the information is through an online viewer.

Jeffrey's Exif Viewer (exif.regex.info/exif.cgi)

I consider Jeffrey's Exif Viewer the online standard for displaying Exif data. The site will allow analysis of any image found online or stored on a drive connected to your computer. The home page provides two search options. The first allows you to copy and paste an address of an image online for analysis. Clicking "browse" on the second option will open a file explorer window that will allow you to select a file on your computer for analysis. The file types supported are also identified on this page. The first section of the results will usually provide the make and model of the camera used to capture the image. Many cameras will also identify the lens used, exposure settings, flash usage, date and time of capture, and file size. In one example, I could see that the camera used was a Canon EOS Digital Rebel with an 18 - 55mm lens at full 55mm setting. Auto exposure was selected, the flash was turned off, and the photo was taken at 2:30 pm on May 7, 2011. Not all of this will be vital for the researcher, but every bit of intelligence counts.

Scrolling down the analysis page will then identify many camera settings that probably provide little information to the researcher. These include aperture information, exposure time, sharpness, saturation, and other image details. Mixed in with this data is the serial number field. This is most common in newer SLR cameras and will not be present in less expensive cameras. These cameras usually identify the make, model, and serial number of the camera inside every photo that they capture. A serial number of a camera associated with an image can be valuable data. This can help an analyst associate other photos found with a target's camera. If an "anonymous" image was found online that included a serial number in the Exif data, and another image was found of a target of the investigation, these two photos can be analyzed. If the serial number as well as make and model of cameras match, there is a good likelihood that the same camera took both images. However, it is important to know that this data can be manipulated. Using software such as ExifTool, a user can modify this data. While this is not a popular tactic to use, it is still possible.

Many new SLR cameras, and almost all cellular telephone cameras, now include GPS. If the GPS is on, and the user did not disable geo-tagging of the photos in the camera settings, you will get location data within the Exif data of the photo. Figure 21.09 (left) displays the analysis of an image taken with a camera with GPS. The data is similar to the previous analysis, but includes a new "Location" field. This field will translate the captured GPS coordinates from the photo and identify the location of the photo. Farther down this results page, the site will display an image from Google Maps identifying the exact point of the GPS associated with the photo. Figure 21.09 (right) displays this satellite view including a direction identifier. Since most cellular telephones possess an accelerometer, the device documents the direction the camera was facing. Most Android and iPhone devices have this capability. Your results will vary depending on the user's configuration of their GPS on the device.

Camera:	Apple iPhone 5
Lens:	4.1 mm
Exposure:	Auto exposure, Program AE, 1/452 sec, f/2.4, ISO 50
Flash:	Off, Did not fire
Date:	**March 24, 2013** 5:13:57PM (timezone not specified) (1 year, 10 months, 13 days, 18 hours, 20 minutes, 25 seconds ago, assuming image timezone of 6 hours behind GMT)
Location:	Latitude/longitude: **38°** 30' 43.8" North, **90°** 30' 0" West (38.512167, -90.500000) Location guessed from coordinates: *1526 Atlantic Crossing Drive, Fenton, MO 63026, USA*

Figure 21.09: A Jeffrey's Exif Viewer result identifying location with map view.

Cropped Images

Another piece of information that we can look for inside the Exif data is the presence of a thumbnail image within the photograph. Digital cameras generate a small version of the photo captured and store it within the Exif data. This icon size image adds very little size to the overall file. When a user crops the image, this original smaller version may or may not get overwritten. Programs such as Photoshop or Microsoft Photo Editor will overwrite the data and keep both images identical. Other programs, as well as some online cropping tools, do not overwrite this data. The result is the presence of the original and uncropped image within the Exif data of the cropped photo. An example of this is seen in Figure 21.10. A cropped photo found online is examined through Jeffrey's Exif viewer. The cropped full-size large photo is seen on the left. The embedded smaller original photo was not overwritten when cropped. We can now see what the image looked like before it was cropped. This technique has been used by police to identify child pornography manufacturers. These pedophiles will crop themselves out of illegal images to avoid identification. When photos of the children are found by police, an original uncropped image may be enough to identify and prosecute a molester. This is not limited to law enforcement. Some tech-savvy fans of television personality Catherine Schwartz examined a cropped photo on her blog in 2003. Inside the Exif data was the uncropped version which exposed her breasts and quickly made the rounds through the internet. We must remember this unfortunate lesson when we consider posting our own content to the internet.

Real World Application: In a civil litigation, a subject claimed an injury that prohibited him from work, walking, and a normal life. The suit claimed damages from pain and suffering and sought a monetary judgment for future lack of ability to work. A brief scan of the subject's online photo album revealed fishing trips, softball games, and family adventure vacations. With Exif information data intact, exact dates, times, locations, and cameras were identified and preserved. The subject withdrew his lawsuit.

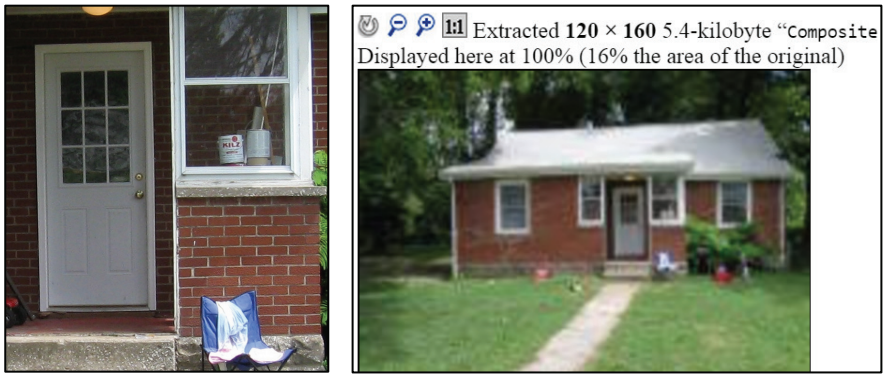


Figure 21.10: A Jeffrey's Exif Viewer summary result displaying an original uncropped photo.

Camera Trace (cameratrace.com/trace)

This site was designed to help camera theft victims with locating their camera if it is being used by the thief online. For that use, you would find a photo taken with the stolen camera, and drop it into the previous site for analysis. This analysis identifies a serial number if available. If one is located, type the serial number into Camera Trace. It will attempt to locate any online photographs taken with the camera. This service claims to have indexed all of Flickr and 500px with plans to add others. A sample search using a serial number of "123" revealed several results. The website urges users to sign up for a premium service that will make contact if any more images appear in the database, but I have never needed this.

Online Barcode Reader (online-barcode-reader.inliteresearch.com)

Barcodes have been around for decades. They are the vertical lined images printed on various products that allow registers to identify the product and price. Today's barcodes are much more advanced and can contain a large amount of text data within a small image. Some newer barcodes exist in order to allow individuals to scan them with a cell phone. The images can provide a link to a website, instructions for downloading a program, or a secret text message. I generally advise against scanning any unknown barcodes with a mobile device since malicious links could be opened unknowingly. However, an online barcode reader can be used to identify what information is hiding behind these interesting images.

Figure 21.11 displays the barcode search options from Online Barcode Reader. These include 1D, PDF417, Postal, DataMatrix, QR, and ID barcodes. After selecting the type of barcode image, you can select any PDF, TIFF, JPEG, BMP, GIF, or PNG file on your computer up to 4 MB in size. This could be a photo that possesses a barcode in the content or a digital code downloaded from a website. Screen captures of codes also work well. While sitting on a plane with Wi-Fi, I captured a photo of an abandoned boarding pass in the magazine holder in front of me. The barcode reader identified text information stored inside the code that was not present in text on the document.

Figure 21.11: Barcode input samples from Online Barcode Reader.

Additional barcode identification options are as follows.

Online Barcode (onlinebarcodereader.com)
Zxing (zxing.org)
Cognex (manateeworks.com/free-barcode-scanner)
Online Decoder (online-barcode-reader.com)

Image Manipulation

It is common to find images on the internet that have been manipulated using software such as Photoshop. Often it is difficult, if not impossible, to tell if these photos have been manipulated by visually analyzing them. A handful of websites use a technique to determine which portions of the photo have changed.

Foto Forensics (fotoforensics.com)

This site allows you to upload a digital image. After successful upload, it will display the image in normal view. Below this image will be a darkened duplicate image. Any highlighted areas of the image indicate a possible manipulation. While this site should never be used to definitively state that an image is untouched or manipulated, investigators may want to conduct an analysis for intelligence purposes only. Figure 21.12 displays original and manipulated images while Figure 21.13 displays the analysis of the images from Foto Forensics. This site will provide an analysis of an image from the internet or a file uploaded from a computer. It is important to note that any images uploaded become part of the website's collection and a direct URL is issued. While it would be difficult for someone to locate the URL of the images, it could pose a security risk for sensitive files.

Figure 21.12: An original photograph (left) compared to a manipulated photograph (right).

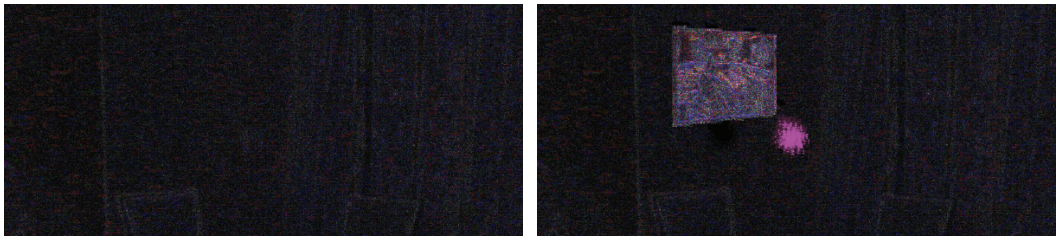


Figure 21.13: The original photograph (left) and manipulated image (right) on Foto Forensics.

Forensically (29a.ch/photo-forensics)

Forensically is a robust image analyzer that offers a huge collection of photo forensic tools that can be applied to any uploaded image. This type of analysis can be vital when image manipulation is suspected. Previous tools have offered one or two of the services that Forensically offers, but this new option is an all-in-one solution for image analysis. Loading the page will present a demo image, which is used for this explanation. Clicking the "Open File" link on the upper left will allow upload of an image into your browser for analysis. Images are NOT uploaded to the server of this tool; they are only brought into your browser locally. Figure 21.14 (left) is the standard view of a digital photo. The various options within Forensically are each explained and example images are included. Due to the black & white environment of this book, I have replicated all of this instruction in color on my blog at the following address.

https://inteltechniques.com/blog/2016/12/21/internet-search-resource-foresically/

The Magnifier allows you to see small hidden details in an image. It does this by magnifying the size of the pixels and the contrast within the window. There are three different enhancements available at the moment: Histogram Equalization, Auto Contrast, and Auto Contrast by Channel. Auto Contrast mostly keeps the colors intact; the others can cause color shifts. Histogram Equalization is the most robust option. You can also set this to none.

The Clone Detector highlights copied regions within an image. These can be a good indicator that a picture has been manipulated. Minimal Similarity determines how similar the cloned pixels need to be to the original. Minimal Detail controls how much detail an area needs; therefore, blocks with less detail than this are not considered when searching for clones. Minimal Cluster Size determines how many clones of a similar region need to be found in order for them to show up as results. Blocksize determines how big the blocks used for the clone detection are. You generally don't want to touch this. Maximal Image Size is the maximal width or height of the image used to perform the clone search. Bigger images take longer to analyze. Show Quantized Image shows the image after it has been compressed. This can be useful to tweak Minimal Similarity and Minimal Detail. Blocks that have been rejected because they do not have enough detail show up as black. Figure 21.14 (right) demonstrates this output.

Figure 21.14: A normal image view (left) and Clone Detector (right) in Forensically.

Error Level Analysis compares the original image to a recompressed version. This can make manipulated regions stand out in various ways. For example, they can be darker or brighter than similar regions which have not been manipulated. JPEG Quality should match the original quality of the image that has been photoshopped. Error Scale makes the differences between the original and the recompressed image bigger. Magnifier Enhancement offers different enhancements: Histogram Equalization, Auto Contrast, and Auto Contrast by Channel. Auto Contrast mostly keeps the colors intact; the others can cause color shifts. Histogram Equalization is the most robust option. You can also set this to none. Opacity displays the opacity of the Differences layer. If you lower it, you will see more of the original image. Figure 21.15 (left) displays manipulation.

Noise Analysis is basically a reverse de-noising algorithm. Rather than removing the noise it removes the rest of the image. It is using a super simple separable median filter to isolate the noise. It can be useful for identifying manipulations to the image like airbrushing, deformations, warping, and perspective corrected cloning. It works best on high quality images. Smaller images tend to contain too little information for this to work. Noise Amplitude makes the noise brighter. Equalize Histogram applies histogram equalization to the noise. This can reveal things but it can also hide them. You should try both histogram equalization and scale to analyze the noise. Magnifier Enhancement offers three different enhancements: Histogram Equalization, Auto Contrast, and Auto Contrast by Channel. Auto Contrast mostly keeps the colors intact; the others can cause color shifts. Histogram Equalization is the most robust option. You can also set this to none. Opacity is the opacity of the noise layer. If you lower it, you will see more of the original image. The result can be seen in Figure 21.15 (right).

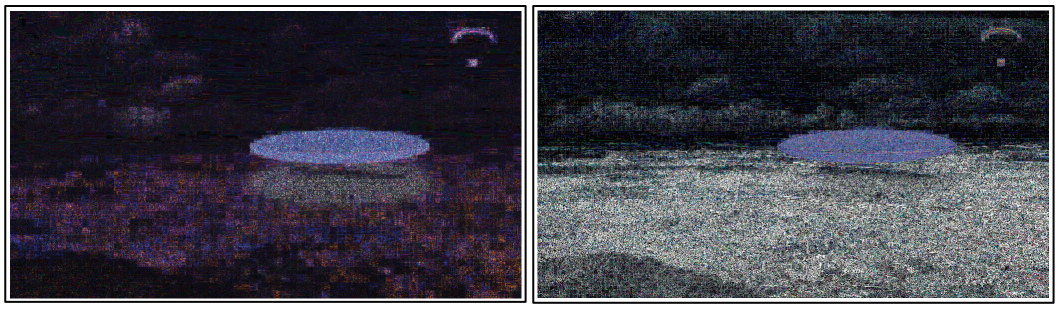


Figure 21.15: Error Level Analysis (left) and Noise Analysis (right) in Forensically.

Level Sweep allows you to quickly sweep through the histogram of an image. It magnifies the contrast of certain brightness levels. To use this tool simply move your mouse over the image and scroll with your mouse wheel. Look for interesting discontinuities in the image. Sweep is the position in the histogram to be inspected. You can quickly change this parameter by using the mouse wheel while hovering over the image, this allows you to sweep through the histogram. Width is the amount of values (or width of the slice of the histogram) to be inspected. The default should be fine. Opacity is the opacity of the sweep layer. If you lower it, you will see more of the original image.

Luminance Gradient analyzes the changes in brightness along the x and y axis of the image. It's obvious use is to look at how different parts of the image are illuminated in order to find anomalies. Parts of the image which are at a similar angle (to the light source) and under similar illumination should have a similar color. Another use is to check edges. Similar edges should have similar gradients. If the gradients at one edge are significantly sharper than the rest it's a sign that the image could have been copied and pasted. It does also reveal noise and compression artifacts quite well. Figure 21.16 (left) displays this view.

PCA performs principal component analysis on the image. This provides a different angle to view the image data which makes discovering certain manipulations and details easier. This tool is currently single threaded and quite slow when running on big images. Choose one of the following Modes: Projection of the value in the image onto the principal component; Difference between the input and the closest point on the selected principal component; Distance between the input and the closest point on the selected principal component; or the closest point on the selected principal Component. There are three different enhancements available: Histogram Equalization, Auto Contrast, and Auto Contrast by Channel. Auto Contrast mostly keeps the colors intact; the others can cause color shifts. Histogram Equalization is the most robust option. You can also set this to none. Opacity is the opacity of the sweep layer. If you lower it, you will see more of the original image. Figure 21.16 (right) displays this view.

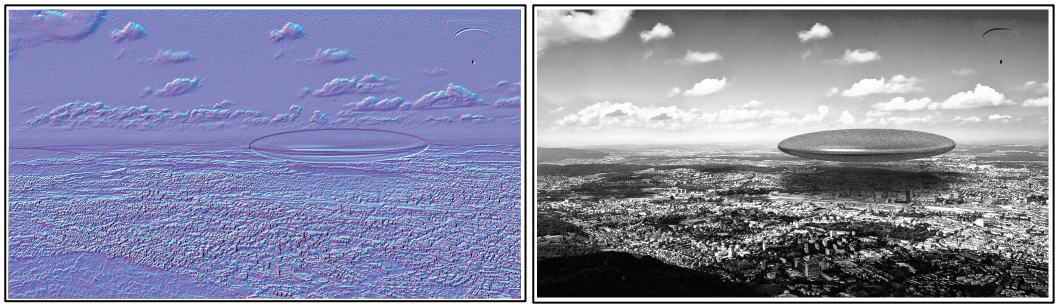


Figure 21.16: The Luminance analysis (left) and PCA analysis (right) within Forensically.

MetaData displays any Exif metadata in the image. Geo Tags shows the GPS location where the image was taken, if it is stored in the image. Figure 21.17 displays the result.

Thumbnail Analysis shows any hidden preview image inside the original image. The preview can reveal details of the original image or the camera used. Figure 21.18 displays the online image (left) while the original thumbnail displays a different view (right).

The next time you identify a digital image as part of your online investigation, these tools will peek behind the scenes and may display evidence of tampering.

Image Enlarging & Upscaling

There may be times when you have an image of poor quality which you may wish to enhance. Typically, this is not advised since you may be manipulating evidence, but there are scenarios where this may be justified. A blurry image of a license plate could warrant the manipulation of an image for clarity. Let's look at two options.

IMG Enlarger (imglarger.com)

This option requires a free account, and only magnifies the overall image. It simply doubles everything in size. I have not found this extremely valuable.

IMG Upscaler (imgupscaler.com)

This option does not require an account and uses various software programming to truly enhance an image. In 2021, I uploaded a blurry vehicle with a license plate which barely identified half of the digits. This tool clarified two additional digits which led to the discovery of the full registration.

Make	SONY	GPSVersionID	2,2,0,0
Model	ILCE-6000	GPSLatitudeRef	N
Orientation	1	GPSLatitude	47.35
XResolution	300	GPSLongitudeRef	E
YResolution	300	GPSLongitude	8.498
ResolutionUnit	2		
Software	darktable 1.6.6		
ModifyDate	2015:08:14 13:32:39		
YCbCrPositioning	2		
Rating	1		
RatingPercent	20		
DateTimeOriginal	Thu Jul 31 2014 09:05:43 GMT-0700 (PDT)		

- View on OpenStreepMap
- View on Google Maps
- Other Images around here on Flickr

Figure 21.17: Metadata from Forensically.

Figure 21.18: An online image (left) and original thumbnail image (right) on Forensically.

Aperisolve (aperisolve.com)

This new service replicates some of the methods previously explained, but adds a few unique options. I uploaded the image from the FaceCheck example and allowed the service to conduct a full analysis. It displayed the typical file attributes and metadata, but then displayed the image with various superimposed values and attempted to identify any steganography patterns. This allowed me to hunt for any hidden artifacts which may not have been present within the previous utilities. Overall, I reserve this tool for situations when I suspect modification to a file which may include hidden data.

IntelTechniques Images Tool

I have created an offline and online (**https://inteltechniques.com/tools/Images.html**) tool which automates most techniques presented within this chapter. The first search options replicate the reverse-search techniques explained for Google, Bing, and others. The "Submit All" option on this page executes keyword searches across all popular networks into separate tabs on your browser.

CHAPTER TWENTY-TWO
VIDEOS

Online videos may now be more common than online photographs. The cameras in smart data phones can act as video cameras. In some situations, uploading a video to the internet is easier than a photograph. Most social networks now act as independent video hosts for their platforms. Video sharing sites such as YouTube have made video publication effortless. For investigations, a video can contain a huge amount of intelligence. When any abnormal event happens, people flock to their phones and start recording. These videos may capture criminal acts, embarrassing behavior, or evidence to be used in a civil lawsuit. Obtaining these videos is even easier than creating them.

YouTube (youtube.com)

The most popular video-sharing site is YouTube. The official YouTube site declares that 500 hours of video are uploaded every minute, resulting in nearly 80 years of content uploaded every day. It further states that over a billion videos are viewed each day. These impressive statistics confirm the need to include videos as part of a complete OSINT analysis. YouTube is easy to search from the main search field on every page. This field can accept any search term and will identify video content or username. Users that upload videos to YouTube have their own "channel". Their videos are uploaded to this channel, and locating a user's channel will identify the videos uploaded by that user.

Many people use YouTube as a social network, leaving comments about videos and participating in discussions about various topics. If you locate a video of interest, it is important to also retrieve this text information. Each comment below a video will include the username that created the comment, which will link to that user's profile.

A search for "school bus fight" returned over 500,000 video links on YouTube. Adding a search term such as the city or school name may help, but it may also prohibit several wanted videos from appearing. The "Filter" option can be expanded to help limit the search scope. This button is above the first video result. This provides additional filter options including the ability to sort by the upload date (date range), type (video vs. channel), duration (short or long), and features (video quality). In the "school bus fight" example, the "uploaded this week" option was chosen. This resulted in only 700 videos which could easily be examined for any intelligence. The lower left portion of any video page includes a link to the profile of the user who submitted this video. This profile page includes all of the videos uploaded by that user and additional profile information. Several YouTube "hacks" have surfaced over the years. Many of these stopped working as YouTube made changes to the environment. Of those still functioning, I find the following techniques helpful to my investigations.

Bypass Age and Login Restriction

Several YouTube videos have been tagged as violent, sexual, or otherwise inappropriate for young viewers. Others demand that you log in to a Google account in order to view the content for unclear reasons. Either way, this is an unnecessary roadblock to your investigation. As an OSINT investigator, I prefer to not be logged in to any personal or covert account while I am researching. Any time you are searching through a Google product while logged in to an account, Google is documenting your every move. This can be unsettling. One easy technique should remove this restriction. Navigate to the following website and notice the inability to view the video. If you are not logged in to a Google account with a verified age, you should see a warning about mature content. This video cannot be played.

https://www.youtube.com/watch?v=SZqNKAd_gTw

We can now append the beginning of this URL with "nsfw", as follows.

Order 0138471223

https://www.nsfwyoutube.com/watch?v=SZqNKAd_gTw

You should be able to play the video through this page. Please be warned that the content in this example contains very disturbing video, hence the blockage by YouTube. The download techniques explained in Chapter Four should also bypass this restriction.

Bypass Commercials with Full Screen

It seems lately that every long YouTube video I play possesses a 30 second commercial at the beginning. This is very frustrating when analyzing a large number of videos. The same URL trick will bypass this annoyance. Navigate to the following address and notice the commercial at the beginning.

http://www.youtube.com/watch?v=IEIWdEDFlQY

Alter this address slightly in order to force the video to play in full screen in your browser. This will also bypass any commercials. The URL should appear like the following.

https://www.youtube.com/embed/IEIWdEDFlQY

Display Thumbnails of Videos

When a user uploads a video, YouTube captures and displays a frame for that media. This is the still frame you see when searching videos and before a video is played. These possess a static URL, which will be helpful when we discuss reverse video searching. As an example, navigate to the following address to load a demo video.

https://www.youtube.com/watch?v=1nm1jEJmOTQ

Using that same video ID, navigate to the following address to view the main still frame. This is the image visible when a video is loaded within YouTube before playing.

https://i.ytimg.com/vi/1nm1jEJmOTQ/hqdefault.jpg

The address that displayed the main image is not your only option. An additional high-resolution image can usually be extracted from this specific video with the following address.

https://i.ytimg.com/vi/1nm1jEJmOTQ/maxresdefault.jpg

Furthermore, we can extract four unique frames with the following URLs.

https://img.youtube.com/vi/1nm1jEJmOTQ/0.jpg
https://img.youtube.com/vi/1nm1jEJmOTQ/1.jpg
https://img.youtube.com/vi/1nm1jEJmOTQ/2.jpg
https://img.youtube.com/vi/1nm1jEJmOTQ/3.jpg

In a moment, our tools will query all of these images for download and reverse image searching.

Identify and Bypass Country Restriction

Many videos on YouTube are allowed to be viewed in some countries and blocked in others. If you encounter a video that will not play for you because of a country restriction, you have options. We will use the following video as a demonstration.

http://youtube.com/watch?v=cgEnBkmcpuQ

Order 0138471223

Visiting this URL from a U.S. IP address should present " Video unavailable - The uploader has not made this video available in your country". Before proceeding, consider identifying from which geographical areas a video is restricted. After you have identified a video with possible country restrictions, paste the video ID into the following URL. Our video ID is cgEnBkmcpuQ.

http://polsy.org.uk/stuff/ytrestrict.cgi?ytid=cgEnBkmcpuQ

The result is a page with a world map. Countries in grey are allowed to view the target video while countries in red are not. Another service which replicates this is WatanNetwork. The following URL displays their map.

https://watannetwork.com/tools/blocked/#url=cgEnBkmcpuQ

While I cannot natively play this video due to my location, I can easily view the default and high resolution still frames with the technique described in the previous section. The following exact URLs display content otherwise not viewable.

https://i.ytimg.com/vi/cgEnBkmcpuQ/hqdefault.jpg
https://i.ytimg.com/vi/cgEnBkmcpuQ/maxresdefault.jpg

If a video is blocked from playing in your location, you can usually use a VPN which should allow viewing. Identify which countries are not blocked using the previous methods and select a server in one of those areas. The internet is full of "YouTube Proxy" websites which promise to play any blocked video, but I have found them to be unreliable.

YouTube Metadata

Most of the details of a YouTube video can be seen on the native page where the video is stored. Occasionally, some of this data may not be visible due to privacy settings or profile personalization. In order to confirm that you are retrieving all possible information, you should research the data visible from YouTube's servers. The most comprehensive way to do this is through Google's YouTube API. Any Google account can request a free API key from Google at developers.google.com. You will need to create a new project and enable a YouTube API key. For your convenience and the search tools, I have already created the following key.

AIzaSyDNALbuV1FZSRy6JpafwUaV_taSVV12wZw

I suspect this key will be abused and disabled at some point, but I will keep the key used in the tools updated. Creating your own key prevents outages. We can now use this for the following query based on our target video.

https://www.googleapis.com/youtube/v3/videos?id=cgEnBkmcpuQ&part=snippet,statistics,recordingDetails&key=AIzaSyDNALbuV1FZSRy6JpafwUaV_taSVV12wZw

This presents a text-only view of all metadata associated with the target video (cgEnBkmcpuQ). While any YouTube video page displays estimated dates, like, dislikes, and comment counts, the metadata is more precise. The following is partial data extracted from our demonstration. I believe every YouTube investigation should document all metadata.

"publishedAt": "2012-07-24T18:33:57Z",
"channelId": "UCP6YCSvxq2HEX33Sd-iC4zw",
"viewCount": "656405279",
"likeCount": "1421566",
"dislikeCount": "717133",
"favoriteCount": "0", "commentCount": "1173"

Order 0138471223

YouTube Profiles

If you ever locate a video of interest, you should investigate the profile which hosts the content. As stated earlier, every YouTube video is associated with a profile. Clicking the uploaders name directly below a video should display this content. However, the profile page displays only a portion of available results. Let's conduct a demonstration. Assume you have identified a suspect video which is associated with the profile at https://www.youtube.com/user/SnubsieBoo. Viewing this page tells you that she has approximately 35,000 subscribers and videos. However, we can dig deeper into her account with the following URL.

https://www.youtube.com/feeds/videos.xml?user=SnubsieBoo

This text-only page presented a lot of data, but I am most interested in the following.

<name>Shannon Morse</name>
<yt:channelId>UCNofX8wmSJh7NTklvMqueOA</yt:channelId>
<link="https://www.youtube.com/channel/UCNofX8wmSJh7NTklvMqueOA"/>
<published>2006-08-16T23:23:03+00:00</published>

These details tell us her Channel ID assigned to her username and the exact date and time she created her YouTube account. All of this should be documented within our investigation. After this content, you can see the metadata of each video, which includes the following.

<published>2020-11-21T15:00:05+00:00</published>
<updated>2020-11-21T15:00:05+00:00</updated>
<media:starRating count="48" average="4.75" min="1" max="5"/>
<media:statistics views="374"/>

This tells us the exact creation and modification times of each video along with viewer details. Again, this text can be helpful in our report. If you created your own API key as explained in the previous page, you can query more details. The following uses my own key and her Channel ID.

https://youtube.googleapis.com/youtube/v3/channels?part=snippet&id=UCNofX8wmSJh7NTklvMqueOA&key=AIzaSyDNALbuV1FZSRy6JpafwUaV_taSVV12wZw

The results tell us she is in the United States ("country": "US") and has a custom YouTube URL at https://www.youtube.com/ShannonMorse ("customUrl": "shannonmorse"). Finally, we can retrieve a full-size image of her profile photo within this code. The following URL appears after "High". This links to a high-resolution (800x800) image of her profile picture, which is otherwise only available as a 160x160 icon. All of the search options on this page are available in the Videos Tool presented later.

yt3.ggpht.com/ytc/AAUvwnix3Pc9x9SX4z85pV6MtKGGTndGxIGqV8_dWJ9bsPw=s800-c

Reverse Image Search

You learned about reverse image searching in the previous chapter. Since YouTube allows us to extract frames from any video without playing the video itself, we can easily automate reverse image searching of those images. We already know that the maximum resolution image for our target video is available at https://i.ytimg.com/vi/cgEnBkmcpuQ/maxresdefault.jpg. Therefore, the following URL would conduct a reverse image search via Google, which should identify additional copies of this video. The Videos Tool will replicate this across Bing, Yandex, and TinEye using the methods explained in previous chapters.

https://www.google.com/searchbyimage?site=search&sa=X&image_url=https://i.ytimg.com/vi/cgEnBkmcpuQ/maxresdefault.jpg

Order 0138471223

This URL presents 305 results, including links to the target video available within dozens of additional video platforms. While this works well on YouTube videos, complete reverse video searching across multiple networks will be explained later in this chapter. The search tools presented at the end automates all of these techniques.

Immediate Download Options

My preferred method for extracting YouTube and other online videos was explained in previous chapters while discussing YouTube-DL and yt-dlp within a Linux, Mac, or Windows OSINT machine. **This will always be the most stable and reliable option, and you should be proficient with the video download strategies explained within Chapter Four.** However, if you have no software or browser plugins available to you, there is another easy option. While you are watching any YouTube video, you can add "deturl.com/" to the address in order to download the video to your computer. To test this, navigate to the following.

https://www.youtube.com/watch?v=t2mU6USTBRE

Now, add "deturl.com/" to the beginning, as indicated in the following address.

https://deturl.com/www.youtube.com/watch?v=t2mU6USTBRE

You will be presented a new page with many options including the ability to download the video; download only the audio; convert the video to a different format; and bypass the age restriction as discussed earlier. Additional options include yout.com, keepvid.com, and y2mate.com.

YouTube Comments

As a reminder, the Video Download Tool previously presented for Linux, Mac, and Windows possesses YouTube-Tool, which extracts comments from video pages. I believe this type of documentation should be a part of every investigation associated with a YouTube video. However, you can also search through all video comments with an online tool. **YouTube Comment Finder** (ytcomment.kmcat.uk) allows a query of any term against all comments posted to a video. This is beneficial when a target video possesses thousands of comments which cannot be loaded into a single page.

The previous video currently has over 42,000 comments. After populating the YouTube URL into YouTube Comment Finder, we can search within those comments. Assume I wanted to know if anyone mentioned MTV within these 42,000 comments. A query for that term immediately displayed the seven results which include MTV. Furthermore, all results display the name of the poster; the full comment; the number of likes and replies; a link to the native comment on the YouTube page; and the exact date and time of post, as seen below.

Brian Teel
This video could never be made today. MTV would be boycotted just for playing it.
Show Comment | Likes: 0 | Replies: 0 | 2/20/2022, 17:07:38

This is a great benefit to our investigations. I wanted to reverse-engineer the method for this query in case the YouTube Comment Finder site ever shut down. It seems they are querying the Google API for the data. The following is the structure, assuming you are searching through the comments of YouTube ID "t2mU6USTBRE" for the term "MTV".

https://www.googleapis.com/youtube/v3/commentThreads?part=id,snippet&videoId=**t2mU6USTBRE**&pageToken=&order=Relevance&maxResults=100&searchTerms=**MTV**&textFormat=plainText&key=AIzaSyDNALbuV1FZSRy6JpafwUaV_taSVV12wZw

The results are not as pretty as displayed through the previous website, but the text-only view from an official Google source could be desired. The following displays the API view of the same comment.

```
    "kind": "youtube#commentThread",
    "etag": "qdR5vhARYWedx117YcpTnx9vECY",
    "id": "Ugyk7ac2hEf7tOAAnA54AaABAg",
    "snippet": {
      "videoId": "t2mU6USTBRE",
      "topLevelComment": {
        "kind": "youtube#comment",
        "etag": "Tp1XaLoiR_MaAFs-XL_6I8Y5Ur4",
        "id": "Ugyk7ac2hEf7tOAAnA54AaABAg",
        "snippet": {
          "videoId": "t2mU6USTBRE",
          "textDisplay": "This video could never be made today. MTV would be boycotted just for playing it.",
          "textOriginal": "This video could never be made today. MTV would be boycotted just for playing it.",
          "authorDisplayName": "Brian Teel",
          "authorProfileImageUrl":
"https://yt3.ggpht.com/ytc/AMLnZu8KhBLuYE6dKaC2N43QqL2eeLiPb3a36xPkuyk=s48-c-k-c0x00ffffff-
no-rj",
          "authorChannelUrl": "http://www.youtube.com/channel/UCMF3qEhJlHmXYarRyGe5MTQ",
          "authorChannelId": {
            "value": "UCMF3qEhJlHmXYarRyGe5MTQ" },
          "canRate": true,
          "viewerRating": "none",
          "likeCount": 0,
          "publishedAt": "2022-02-21T00:07:38Z",
          "updatedAt": "2022-02-21T00:07:38Z"}},
      "canReply": true,
      "totalReplyCount": 0,
      "isPublic": true
```


YouTube Channel Crawler (channelcrawler.com)

As previously explained, anyone can search YouTube and filter by Channels. This allows you to only see results which possess one or more videos within a designated Channel. Unfortunately, the results place emphasis on the most popular channels. Within an investigation, it is much more likely that your target will not have thousands of views or followers. Instead, a channel with no subscribers is more common. Finding these poorly-visited channels is quite difficult with official search options.

Instead, we can find these lesser-known collections with YouTube Channel Crawler. Let's conduct an example demonstration. I queried the term "Guns" within YouTube, clicked the filters option, and chose to only display Channels. I received numerous results, and every Channel featured over 100,000 subscribers. I would never find my target there. Now, let's use our crawler. I chose the term of "Guns", no limit to the results, a maximum of 40 subscribers and 40 total views, and did not specify a date range. Figure 22.01 displays partial results. As you can see, these Channels receive very little attention, but were at the top of my results due to the filters I applied.

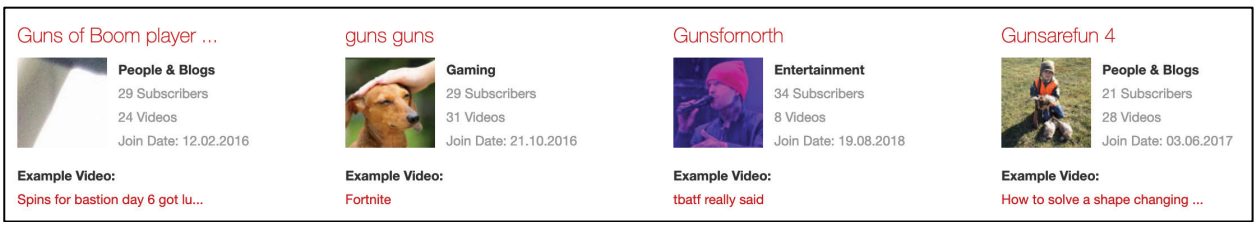


Figure 22.01: Channel results on YouTube Channel Crawler.

Order 0138471223

YouTube Unlisted Videos

YouTube videos can be classified as "Public", "Private", or "Unlisted". Public videos appear within YouTube search results; private videos require an invitation from the host; and unlisted videos sit in between. Unlisted videos will not appear within search results, but they can be seen by anyone possessing the direct URL. There are two methods to discover unlisted videos. First, we can conduct a search on Google such as the following.

site:youtube.com "This video is unlisted" intitle:osint

This can be unreliable, as it presents videos which contain "This video is unlisted" within the description provided by the uploader. I find **Unlisted Videos** (unlistedvideos.com) to be more reliable. Conduct a keyword search on this site to identify videos which are unlisted and not present within search results.

YouTube is not the only video sharing service on the internet. Wikipedia identifies dozens of these sites, but searching each of them can become tedious. These sites are no longer restricted to pages of video files with standard extensions such as mp4, mpg, and flv. Today, services such as Instagram allow embedded videos which do not conform to yesterday's video standards. Many new services present the viewer with animated gif files that only appear as true videos. Fortunately, search engines like Google, Bing, and Yandex offer a search across all of the types.

Google Videos (google.com/videohp)

A search on YouTube for "school bus fight" returned over 500,000 results. However, Google Videos returned 3 million results. These include the results identified in the previous YouTube search plus any videos from other sites that meet the search criteria. This will often lead to duplicate videos that have been posted by news websites and social networks. Google can filter these results by duration time, date and time captured, and video source. The top menu of any Google video results page will display these options. A Google search for the term "female street fight", including filters for videos with a short duration that were posted this week from any source, returned over 900 results. These results could either be further filtered with search terms or quickly viewed by still frame to determine relativity to the investigation.

Bing Videos (bing.com/videos)

One feature that makes Bing a favorite site for searching videos is the instant video playback option. When viewing a video search results page, simply hovering the cursor over the video still shot will start the video playback from the beginning of the video. This eliminates the need to navigate to each video page for playback to determine the value of the video. Bing also offers filtering by length and source. The "select view" toolbar at the top of each search result page will allow you to sort the results by either the best match or the most recent. Whether using Google or Bing to locate videos, I recommend turning off the safe search feature. This feature is designed to prohibit some videos with adult content from displaying. With investigations, it is often these types of videos that are wanted.

Yandex Videos (yandex.com/video)

In 2020, Yandex's video search option became a contender for OSINT usage. While Google and Bing are constantly removing videos, which violate their policies associated with violent and inappropriate content, Yandex seems to allow anything. Similar to my recommendations for general search and images, Yandex Videos should always be queried when researching any video file. Direct query URLs for all three services follows.

https://www.google.com/search?tbm=vid&q=osint
https://www.bing.com/videos/search?q=osint
https://yandex.ru/video/search?text=osint

Order 0138471223

Social Network Videos

Google, Bing, and Yandex index social networks for video pages, but these search engines can never replicate internal queries through popular networks such as Twitter, Facebook, Reddit, and others. We should always consider a keyword search directly within these services. The following assumes "osint" is your search term and provides a direct URL for each network query.

Twitter: https://twitter.com/search?q=osint&f=video
Facebook: https://www.facebook.com/search/videos/?q=osint
Reddit: https://www.reddit.com/search?q=site:v.redd.it%20AND%20osint
TikTok: https://www.tiktok.com/tag/osint

When you locate videos embedded within social networks, use the previous methods to download any evidence. I always start with the Video Download Tool. If it is a live stream, I use the Video Stream Tool. If neither are available or functioning, browser extensions may work. If you get desperate, try various free third-party tools such as **Twitter Video Downloader** (twittervideodownloader.com), **FDown** (fdown.net), and **Instagram Downloader** (igram.io).

Deleted Videos

It has become very common for people to remove their YouTube videos. This often happens when unwanted attention is generated and the user regrets the post. The following technique will not always work, but I have had many successes within my own investigations. Consider the following video which has been removed from YouTube.

https://www.youtube.com/watch?v=9ZmsnTDLykk

The Internet Archive has been mirroring YouTube videos for years and often possesses their own independent copies. We can look for this with the following URL.

https://web.archive.org/web/https://www.youtube.com/watch?v=9ZmsnTDLykk

This identifies 276 captures of this video page. However, these are HTML archives, and the video will not play within any of them. These pages are beneficial for locating comments associated with your target video, but not for the video content itself. We can use the following URL to play the full resolution version of the archived video.

https://web.archive.org/web/2oe_/http://wayback-fakeurl.archive.org/yt/9ZmsnTDLykk

We can now right-click on the video to save our own offline copy. You would only need to replace your target YouTube video ID with the one listed here (9ZmsnTDLykk). Our search tools will replicate this for you at the end of the chapter.

Reverse Video Searching

There was a brief mention earlier of conducting a reverse image search on the still captures of a YouTube video. This would use the same techniques as mentioned in Chapter Twenty-One for images. While there is no official reverse video search option, applying the techniques to still captures of videos can provide amazing results. This method is not limited to YouTube. We can conduct reverse image searches on videos from many sites. As the popularity of online videos is catching up to images, we must always consider reverse video searches. They will identify additional websites hosting the target videos of interest. Before explaining the techniques, consider the reasons that you may want to conduct this type of activity.

Order 0138471223

- School resource officers and personnel are constantly notified of inappropriate video material being posted online. Identifying these videos, consisting of fights, malicious plans, and bullying, may be enough to take care of the situation. However, identifying the numerous copies on other websites will help understand the magnitude of the situation.
- Global security divisions monitoring threats from protest groups will likely encounter videos related to these activities. However, identifying the numerous copies on various websites will often disclose commentary, online discussions, and additional threats not seen in the primary source video.
- Human trafficking investigators can now reverse search videos posted to escort providers similar to the now seized site Backpage. The presence of identical videos posted to numerous geographical areas will highlight the travel and intentions of the pimps that are broadcasting the details.

There are limitless reasons why reverse video searching should be a part of your everyday research. The following methods will get you started with the most popular video websites. These techniques can be replicated as you find new services of interest.

YouTube: As explained earlier, YouTube offers four still frames for every video uploaded plus a high-resolution image. Obtain the URLs outlined during that instruction, and provide each to Google, Bing, Yandex, TinEye, and Baidu as a reverse image search as previously explained. The Videos Tool presented later will automate this process.

Vimeo: Vimeo does not natively offer URLs with a video ID that display screen captures of multiple frames. However, they do provide a single high definition still capture for every video. This is stored in the Application Programming Interface (API) side of Vimeo, but it is easy to obtain. As an example, consider your target is at https://vimeo.com/99199734. The unique ID of 99199734 is assigned to that video. You can use that number to access the video's API view at https://vimeo.com/api/oembed.json?url=https://vimeo.com/99199734. This address will display a text only page with the following content.

Type: "video",
version: "1.0",
provider_name: "Vimeo",
provider_url: "https://vimeo.com/",
title: "Billy Talent 'Try Honesty'",
author_name: "S M T",
author_url: "https://vimeo.com/user10256640",
is_plus: "0",
html: "<iframe src="https://player.vimeo.com/video/99199734" width="480" height="272" frameborder="0" title="Billy Talent 'Try Honesty'" webkitallowfullscreen mozallowfullscreen allowfullscreen></iframe>",
width: 480,
height: 272,
duration: 247,
description: "Music Video Directed by Sean Michael Turrell.",
thumbnail_url: "https://i.vimeocdn.com/video/513053154_295x166.jpg",
thumbnail_width: 295,
thumbnail_height: 167,
upload_date: "2014-06-25 21:29:06",
video_id: 99199734,

The portion relevant to this topic is the thumbnail URL. Following that description is an exact address of the "medium" image used at the beginning of each Vimeo video. Removing the "_295x166.jpg" from the end of the URL presents a full size option, such as the following.

https://i.vimeocdn.com/video/513053154

We will use this static URL within our tools. Also note that this view identifies the exact date and time of upload. The video page view only identified the date as "1 Year Ago". A reverse image search of the thumbnail URL, using our video search tools in a moment, will produce additional websites which host the same (or similar) video.

Others: Repeating this process for every video sharing website can get redundant quickly. Instead, consider the following. Practically every online video possesses a still image that is displayed to represent the video before being played in search results. This image is likely a direct link that can be seen in the source code. Searching "jpg" and "png" may quickly identify the proper URL. Providing this image to various reverse image search websites will likely display additional copies of the target video within unknown websites.

Internet Archive (archive.org)

The premise of this site is to permanently store open source movies, which can include commercial and amateur releases. The search option at the beginning of every page allows for a specific section of the site to be searched. Selecting "community video" will provide the best results for amateur video. A large number of anti-government and anti-American videos are present and ready for immediate download. Unlike YouTube, this site does not make it easy to identify the user that uploaded the videos. Furthermore, it does not link to other videos uploaded by the same user. To do this, you will need to look for some very specific text data. As an example, consider that Internet Archive user Enver_Awlaki is your target. His video profile is located at http://www.archive.org/details/Enver_Awlaki. One of his video pages is stored at the address of https://archive.org/details/Awlaki_to_americans.

Below the video frame in the center of the page are several options on the lower right. These allow you to specify video files with different file types. Below these options is a link titled "Show All". Clicking the link provides a view of the files associated with the video as seen in Figure 22.02. The eighth link on this list forwards to the metadata associated with the video. This data includes the title, description, creator, email address used to upload, and the date of upload, as seen in the text below the example.

Enver_Awlaki.thumbs/	26-Feb-2011 09:08	-
Enver_Awlaki_archive.torrent	26-Jun-2016 22:58	33.4K
Enver_Awlaki_avi.avi	26-Feb-2011 00:05	417.1M
Enver_Awlaki_avi.gif	26-Feb-2011 09:13	308.9K
Enver_Awlaki_avi.ogv	26-Feb-2011 10:46	186.6M
Enver_Awlaki_avi_512kb.mp4	26-Feb-2011 09:52	202.4M
Enver_Awlaki_files.xml	26-Jun-2016 22:58	36.6K
Enver_Awlaki_meta.xml	26-Jun-2016 22:58	686.0B
Enver_Awlaki_wmv.gif	26-Feb-2011 09:04	312.1K

Figure 22.02: An Internet Archive video options page.

<mediatype>movies</mediatype><collection>opensource_movies</collection>
<title>Awlaki_to_americans</title><description>UmmaNews</description>
<uploader>ibnumar@islamumma.com</uploader>
<addeddate>2012-03-31 22:47:36</addeddate>
<publicdate>2012-04-01 00:09:10</publicdate>

This view quickly identifies the email address of ibnumar@islamumma.com as the verified uploader of the video content. It also displays the exact date and time of upload and publication. In this example, notice that the author waited over an hour to publish the content. Since 2016, I have seen the Internet Archive become a very popular place to store video, especially from international subjects that may be blocked from traditional American services such as YouTube.

Order 0138471223

TV News Archive (archive.org/details/tv)

At the time of this writing, the TV News Archive, another part of archive.org, had collected 2,094,000 television news broadcast videos from 2009-2020. Furthermore, it extracts the closed captioning text from each video and provides a search option for this data. This allows you to search for any words verbally stated during these broadcasts in order to quickly locate videos of interest. A search of the term "Bazzell" resulted in 35 videos that mentioned someone with my last name within the broadcast. Selecting any result will play the video and all text from the closed captioning. The menu on the left will allow filtering by show title, station, date, language, and topic. I have found this resource valuable when vetting a potential new hire for a company.

Video Closed Captions (downsub.com)

YouTube and other providers attempt to provide captioning subtitles for as many videos as possible. This automated process transcribes any spoken dialogue within the audio of the video file and documents the words to text. To see this text while watching a video, click on the closed captioning icon (cc) in the lower right area of the video box. When the icon changes to a red color, the subtitles will display. These subtitles are contained within a small text file associated with the video. It also includes timestamps that identify the frame in which each piece of text is spoken. YouTube does not provide a way to obtain this text file, but Downsub does. Copy an entire URL of any YouTube video with closed captioning. Paste this link into this website and execute the process. This will display download links for the captioning inside the video. Links for each language will download text files with an .srt file extension. These automated captions are not usually completely accurate. Slang and mumbled speech may not transcribe properly. Any time you collect and submit a YouTube video as part of a report, I recommend obtaining this caption file as well. Even though the actual text may not be accurate, it can help during official proceedings with identifying a specific portion of a video.

Live Video Streams

If you are investigating any live event that is currently occurring, live streaming video sites can be a treasure of useful intelligence. These services offer the ability for a person to turn a cell phone camera into an immediate video streaming device capable of broadcasting to millions. The common set-up is for a user to download a host service's application to a smartphone. Launching the application will turn on the video camera of the phone and the video stream is transmitted to the host via the cellular data connection or Wi-Fi. The host then immediately broadcasts this live stream on their website for many simultaneous viewers to see. An average delay time of five seconds is common.

There are now several companies that provide this free service. The following are listed in my order of preference for investigative needs. Each site has a search option to enter the keywords that describe the live event you want to watch. You may also see Twitter links to these services while monitoring targets.

YouTube (youtube.com/live)
LiveStream (livestream.com)
Twitch (twitch.com)
LiveU (liveu.tv)
YouNow (younow.com)
UScreen (uscreen.tv)

Consider the Video Stream Tool presented in Section One when you want to view and capture any live video streams. It provides the most robust and stable option, especially if you keep your tools updated with the automated scripts.

Order 0138471223

Real World Application: During several large events, I have used live streams to capture the majority of my intelligence. In one investigation, I was assigned the task of monitoring social networks during a large protest that had quickly become violent to both officers and civilians. While Twitter and Facebook occasionally offered interesting information, live streams provided immediate vital details that made a huge impact on the overall response of law enforcement, fire, and EMS. The live video streams helped me identify new trouble starting up, victims of violence that needed medical attention, and fires set by arsonists that required an immediate response.

Petey Vid (peteyvid.com)

This is nothing more than a simple video search engine, but I find it valuable. In a recent case, I searched a very unique keyword on Google and found no results. Petey Vid found a video from an otherwise unknown video host (bitchute.com). This is now in the tools and checked often.

Listen Notes (listennotes.com)

This is not a video utility, but I believe it fits best within this chapter as a media service. Listen Notes queries a keyword through millions of audio podcasts and presents any results. As an example, I conducted a search of the term "privacy" and received 10,000 matches of that word within the indexed podcasts. We can take this a step further with an email address search. I searched "privacypodcast@protonmail.com" and received one result based on show notes of an old episode. As the number of podcasts continue to grow, and audio to text technology improves, this technique will become more useful in our investigations.

Video Analysis

If you have discovered a target video on a YouTube or Vimeo page, you may want to analyze individual frames without using the utilities presented within Chapters Four and Six. You can use **Anilyzer** (anilyzer.com) for this task. Enter the video URL; select the appropriate provider; and click "Watch Video". you can now click through each still frame or modify the playback speed to fit your needs. I use this tool when I need a quick way to scrutinize a handful of frames, especially with surveillance videos.

YouTube Handles

In 2022, YouTube began rolling out user-defined handles as a way to connect to your favorite media. This was already somewhat available within the URL. As an example, the following two links connect to the same content. The first is a user's page and the second is their handle.

https://www.youtube.com/ambermac
https://www.youtube.com/@ambermac

I currently do not see much investigative value, aside from an additional URL to query when researching usernames. I believe most people who possess a custom YouTube handle already possess a profile with the same name.

IntelTechniques Videos Tool

If you feel overwhelmed at this point, I understand. Navigate to the "Videos" page in your Tools download, or online at **https://inteltechniques.com/tools/Videos.html**, to access an all-in-one option similar to the previous tools. This should replicate and simplify the processes which were explained throughout this chapter. I hope you find it helpful. Please note that some fields only require a YouTube video ID, while others require search terms or usernames.

Order 0138471223

CHAPTER TWENTY-THREE
BROADCAST STREAMS

Within the previous chapter, I explained the analysis and collection of online videos. While I did briefly mention live video streams on services such as YouTube, there is a whole other online world of live broadcast streams. These are typically not pre-recorded, and often are not archived after the broadcast. Before we dive into the techniques, let's discuss the reason these streams could be important to our investigations. I will break this chapter up into four topics, each of which are explained below.

Live Television News: Most international television news services broadcast their shows live on the internet. This can be very valuable when you are investigating a live incident far from your current location. If you are in California, it may be difficult to watch the local news in Chicago to monitor current events there. Through online video streams, we can watch live local, state, national, and international news broadcasts from anywhere in the world. We will also use our own custom tools to easily access the streams important to you.

Live Commercial Radio: Similar to TV broadcasts, most commercial radio stations broadcast over the internet. This access can be equally as valuable when trying to listen to a live news broadcast from another state or country.

Live Web-Controlled Radios: Online radio broadcast streams are great, but they never replicate the true listening experience one would receive if listening locally through a receiver. There is often a delay; targeted commercials might play during the middle of a broadcast; and you might receive a different stream of audio based on your location (for marketing purposes). Sometimes, you might want to listen to a pure audio broadcast exactly as it would be heard in another state or country through a radio receiver. We will access physical radios, which are connected to external antennas, from all over the world via the internet. We will even control them, change the stations, and hear the same audio that we would receive if we were present at that location. This also applies to shortwave frequencies and pirate radio stations. As one of many examples, we can tune into a radio present in Ireland, which has an antenna targeting AM reception, and listen to stations live in that region, including all BBC content, and shortwave pirates broadcasting from the ocean. I believe this is one of the most fascinating pieces of technology we have today.

Live Web-Controlled Scanners: Similar to the previous topic, we will eavesdrop on local scanner radios which will allow us to hear emergency radio traffic in real time all over the world. If you are investigating a live event, there is no better source of news than the police, fire, and EMS working at the scene. If we miss anything, we can tap into online archives of the transmissions.

Nerd alert! I confess that I tend to geek-out on audio and video broadcasts, especially those in other countries. This might stem from my early days of local, shortwave, and scanner radio monitoring. 13-year-old me could have never understood how I could connect to practically any live video or audio stream anywhere in the world in real time. While you may not have the same fascination as I do for various broadcasts, please approach with an open mind. These broadcasts can be extremely valuable to your own investigations.

Order 0138471223

Live Television News Streams

Only a few years ago, the local news which was served to your television was determined by your location. An antenna might receive the analog or digital feed from a local news station's building, or cable television might bring in that same video stream. If you lived in Chicago, you received Chicago news. More recently, satellite television began offering local news stations, often as an affiliate for ABC, CBS, or NBC, along with national programming. Some satellite services even offered the ability to purchase channels from other major metropolitan areas. Today, most local, national, and international news services offer their content via the internet through their websites. This is nothing new, and I assume most readers already know they can retrieve this content. However, I believe there is a better way.

When you visit a website to stream local news, you also receive all associated web trackers, advertising scripts, and forced commercials before and during the broadcast. Some stations will not let you maximize the video window because they don't want your eyes interrupted from their flashy ads and popup annoyances. My preference is to identify the video stream and view it through my own web-based tools which bypass most annoyances. The following tutorial will walk you through a few common scenarios which will help identify the streams for your own needs. At the end, I present my custom online and offline tools which already have most of this content embedded into them. You can decide if my tools suffice for your work or if you should create your own. A lot of the content within this section may seem technically complicated. Please know that all of the coding is optional, and you can rely on my tools if desired. However, an understanding of the technology and creation of your own tool now can lead to fewer headaches later. Let's begin.

CBS offers a choice of 24-hour national and local news coverage. A national feed can be found at cbsnews.com. Clicking the play icon streams their live national feed in your web browser on top of the other elements within their web page. However, I prefer to access the video feed itself. In Chapter Three, I explained the use of the Firefox add-on called Stream Detector. We can also use this to identify these live news streams. Before clicking the play option, make sure that Stream Detector is installed and active within Firefox. After beginning the live video stream, you should notice new results within the Stream Detector tool. Figure 23.01 displays my results.

Type	Filename	Size	Source	Timestamp	✖
HLS	1753312.m3u8	-	IntelTechniques Video Feed Tool	10/12/2022 1:49:24 PM	✖
HLS	master.m3u8	-	IntelTechniques Video Feed Tool	10/12/2022 1:49:24 PM	✖
HLS	master.m3u8	-	IntelTechniques Video Feed Tool	10/12/2022 1:49:23 PM	✖

Figure 23.01: A Stream Detector result.

I prefer to read these results from the bottom to the top, and I typically look for the first entry of any "m3u8" file in the list. In this scenario, the first entry loaded within the page was "master.m3u8". Left-clicking this file copies it into your computer's clipboard. At the time of this writing, the first "master.m3u8" file possessed the following complete URL.

https://dai.google.com/linear/hls/event/Sid4xiTQTkCT1SLu6rjUSQ/master.m3u8

This file serves as a "playlist" or set of instructions to play a live video stream. Placing this URL directly in your browser will likely download the file, but not play the video stream. Instead, we should create our own HTML file like we did with the rest of our tools within Chapter Eight. I titled my file cbs.html and placed it within the "video" folder in my online and offline tools package. The entire content appears in the following section.

```
<head><link href="https://vjs.zencdn.net/7.20.3/video-js.css" rel="stylesheet"
/></head>
<body>
<video
id="CBS"
autoplay="true"
class="video-js"
controls
preload="auto"
width="640"
height="480"
data-setup="{}">
<source
src="https://dai.google.com/linear/hls/event/Sid4xiTQTkCT1SLu6rjUSQ/master.m3u
8" type="application/x-mpegURL" /></video>
<script src="https://vjs.zencdn.net/7.20.3/video.min.js"></script></body>
```

Now, let's pick apart the important pieces. The line starting with `<head>` begins our file and incudes a link to an online repository of our video player (Video JS). This is the code which will play the video stream in a nice format. Within the `<body>` section, we can set our desired settings for the video output. This includes a unique identifier (CBS); the setting to automatically begin the stream (autoplay); and the desired width and height (640x480). You could modify any of these for your own needs. The `<source` section includes our URL which we retrieved from Stream Detector, and the final script source is required by our video player. Figure 23.02 displays the result when I open this file within a web browser. You can see the live file (or view the source code) at **https://inteltechniques.com/tools/video/CBS.html**. If you ever have issues, or simply want to update your version of Video JS, visit **https://videojs.com/getting-started/**. Notice the bottom controls of the player. This is Video JS in action. We can pause the stream; change the volume; load closed captioning; or display the stream full-screen with these controls.

Figure 23.02: A live video stream.

I suspect you are wondering why we took all of that effort to display a live video stream when we could have just visited the source website and play it from there. Aside from the tracking and website annoyances previously mentioned, this method allows us to build our own tool which can display any customized list of video streams. I currently maintain an online page at **https://inteltechniques.com/tools/Video.html** which possesses live feeds from most major U.S. cities as well as numerous international feeds. If you downloaded the custom offline tools as explained in Chapter Eight, you also possess an offline version of this page which you could easily modify to suit your own investigative needs. The following source code is an abbreviated version within both the online and offline tools.

Order 0138471223

```
<script type="text/javascript">
function dosearch(menu)
{window.open(menu, 'inner');}
</script>
<form onsubmit="dosearch(this.menu.value); return false;">
<select name="menu" id="menu">
<option value="" selected>Select</option>
<option disabled="">US:</option>
<option value="video/ABC.html">ABC National</option>
<option value="video/CBS.html">CBS National</option>
<option value="video/FOX.html">FOX National</option>
<option value="video/NBC.html">NBC National</option>
</select>
<input type="submit" style="width:80px" value="Launch" /></form>
<iframe name="inner" class="inner" width="640px" height="480px"
overflow="hidden" frameborder=0 scrolling="no"></iframe>
```

In this example, you can see a reference to the CBS.html file which we previously created. It is one option within the drop-down menu. You could add any files you created which contain a live stream. The "iframe" line tells the browser to load the chosen file within a frame on the page to keep the menu present. Figure 23.03 displays the online tool with the CBS option selected. I purposely left the menu visible to see a portion of the options included within my demo tools.

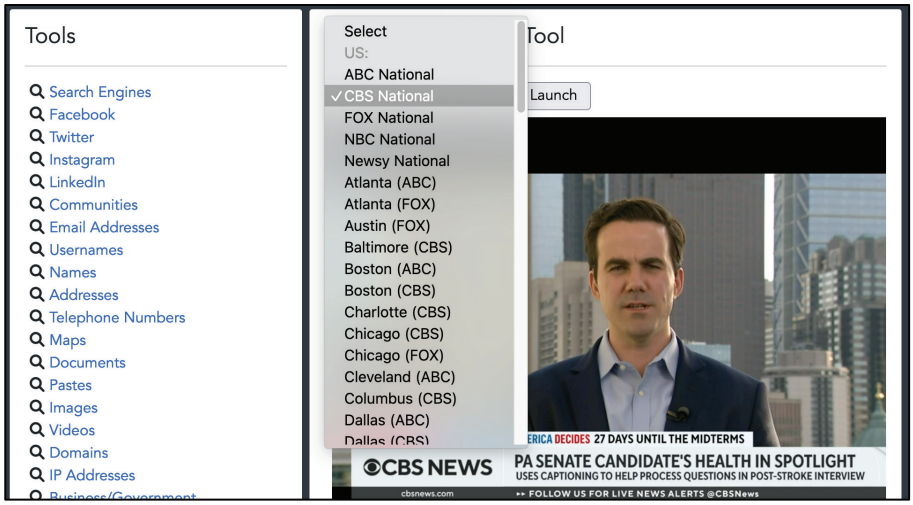


Figure 23.03: A custom streaming tool page.

Now that you have one CBS station completed, you could replicate the process for the CBS local coverage streams for Atlanta, Baltimore, San Francisco, Boston, Chicago, Dallas, Denver, Detroit, and many others. You can find each page under the "Local" section at cbsnews.com, or simply copy them from the pages I already created within the online and offline search tools. This allows you to instantly connect to a live news stream in every major U.S. city at any time. I have this running daily. As I write this in Los Angeles, I received word of a series of shootings in Chicago which I wanted to hear more about. I could wait for minimal coverage on CNN, or I could watch the live Chicago feeds from Fox and CBS to receive better details. There are countless ways in which this tool could be valuable to investigators.

Many live news streams broadcast on YouTube. Consider Al Jazeera as an example. If you navigate to **https://www.youtube.com/watch?v=F-POY4Q0QSI**, you will be presented their current live stream page. I can do without all of the Google tracking, user comments, and constant chat flowing down the screen. Instead, I want to embed this stream into my tools. I created a file called AlJazeera.html which contained the following entire content.

```
<body>
<iframe width="640" height="480" src="https://www.youtube-
nocookie.com/embed/F-POY4Q0QSI?controls=0&autoplay=1" frameborder="0"
allow="accelerometer; autoplay; clipboard-write; encrypted-media"
allowfullscreen>
</iframe>
</body>
```

This code sets the size of the video; identifies the stream source; informs the browser to automatically play the content; and allows the video to be maximized to full screen playback. Notice the portion I bolded (F-POY4Q0QSI). This matches the YouTube live stream ID from the previous YouTube URL. You can now replace this single identifier within this code to make a file for any other live YouTube stream. Let's do one more example. The European live streaming news source from Germany called DW News possesses a YouTube live stream at https://www.youtube.com/watch?v=GE_SfNVNyqk. The following code would embed this stream into a new file.

```
<body>
<iframe width="640" height="480" src="https://www.youtube-
nocookie.com/embed/GE_SfNVNyqk?controls=0&autoplay=1" frameborder="0"
allow="accelerometer; autoplay; clipboard-write; encrypted-media"
allowfullscreen>
</iframe>
</body>
```

Live examples of both of these pages can be analyzed at the links below. They are also both included in the Live Video Streams Tool at **https://inteltechniques.com/tools/Video.html**.

https://inteltechniques.com/tools/video/AlJazeera.html
https://inteltechniques.com/tools/video/GermanyDW.html

CBS and YouTube might provide all the coverage you need, but those are just the tip of the iceberg. You can locate practically any online live news stream; identify the stream file with Stream Detector; and create a new page for your own set of tools. From there, you can rely on the following channel aggregators to find new live streams of interest.

TubiTV (**tubitv.com/live**): This service streams hundreds of channels to your browser for free. You can also use their stream URLs to populate your own tool. As one example, I use their live ABC national stream located at https://live-news-manifest.tubi.video/live-news-manifest/csm/extlive/tubiprd01,ABC-News.m3u8 within my tool set. I also collected numerous local news channel URLs from their "Local" page.

Pluto (**pluto.tv/en/live-tv**): This service also presents hundreds of free live streaming channels including several national and international news feeds. They tend to block their links when posted within another website. However, they should work fine within your own offline tools ran from a folder on your computer. Figure 23.04 (left) displays the result when I tried to play a Pluto stream from my website's online tools. Their security settings prevent this. However, Figure 23.04 (right) shows the result when I launch the same file from my offline tools on my Desktop via my browser. Always try all options when you encounter issues.

USTV247 (**ustv247.tv**): This site might be illegally streaming premium services such as HBO, but they also offer streams which can be found for free. For example, their PBS stream is currently being broadcasted from the URL of https://h3.ustvgo.la/PBS/myStream/l_529711_10497200_1974.ts. Place this into a new html file relying on the Video JS code, and you have PBS in your tool set. My example is located within my online search tools at https://inteltechniques.com/tools/video/PBS.html, however, this page goes down often as USTV247 blocks my copying of their stream (the irony). If they block cross-site scripting for a channel you would like, placing it within your offline tools should bypass the restriction.

Order 0138471223

LocalNow (**localnow.com/channels**): If you are still craving local news options, LocalNow should have you covered. Use Stream Detector to identify feeds of interest and place them within an HTML file using Video JS as previously explained.

Roku (**therokuchannel.roku.com**): This option has limited live content, but it may be useful to you.

The rest is up to you. The tutorials here, along with the files I have already created for you in the offline search tools, should be all you need to create your own customized tool.

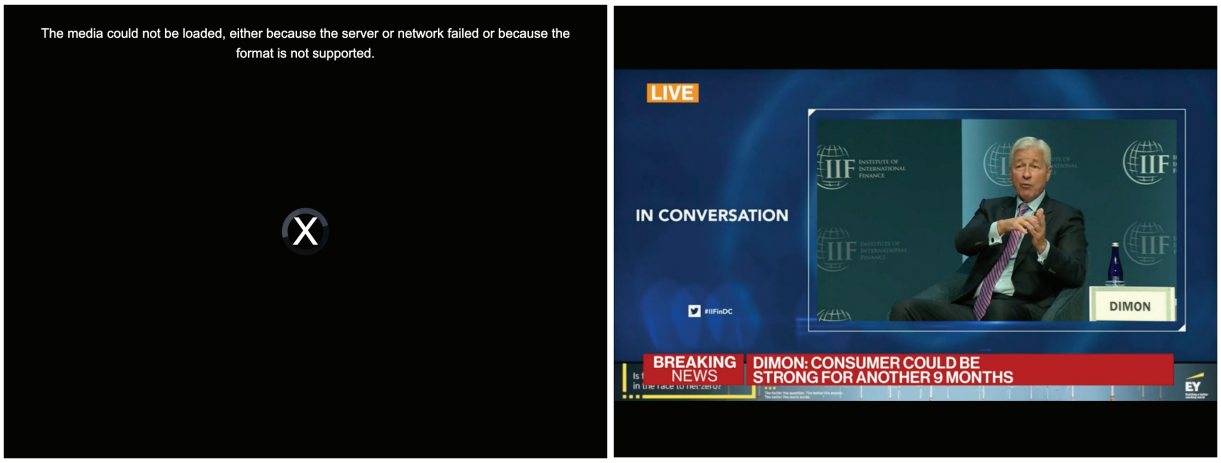


Figure 23.04: A blocked stream (left) which was bypassed in offline tools (right).

Live Commercial Radio

Live radio streams are old news. Most commercial AM and FM stations have a website which will stream their live broadcasts online. Some even offer second and third alternate streams with different content. If you have a favorite station, you may already stream a high-quality version of their content while you work. However, there is much more to discuss.

Similar to my reasons to stream news video within my own tool, I prefer to stream audio feeds outside of the provider's website. Some radio streaming websites insert additional advertisements before the radio stream, which are delivered through the website. Some make you sit through a video ad before the stream begins. If we can identify the desired streams, we can add them to our own tool which can bypass several annoyances. It also offers us a clean way to select the audio feed without interruptions.

Let's begin with something basic and easy. If you navigate to the BBC World News site at bbc.com/news/world, you will not find a link to their 24/7 audio stream. However, if you go directly to their "Sounds" site located at https://www.bbc.co.uk/sounds/play/live:bbc_world_service, their embedded web player will begin playing their World Service radio stream. Unfortunately, this is a MPD file which will not play by itself within our tools. A search for "BBC radio streaming URLS" presented me with numerous results which displayed pure streaming URLs for our needs. I now know that http://stream.live.vc.bbcmedia.co.uk/bbc_world_service is what I want. I can place this directly into my browser and stream live audio from the BBC. I could bookmark this URL for future use. If only every service was this easy.

Many live radio streams will be in "m3u8" format. Much like the video streams which use this format, we will need to play these streams through a third-party player. I currently rely on HLS Player (hlsplayer.org). Let's conduct another example. Al Jazeera Radio has a direct stream located at the following URL.

https://live-hls-audio-web-aje.getaj.net/VOICE-AJE/index.m3u8

However, inserting this into your browser simply downloads the m3u8 file. If we add our HLS Player link before the stream, it will play fine through our browser. The following link loads the Al Jazeera Radio URL into an embedded player for direct access to the audio stream.

https://www.hlsplayer.org/play?url=https://live-hls-audio-web-aje.getaj.net/VOICE-AJE/index.m3u8

Before we create our custom audio streaming tool to handle all of this, we should first consider sources for the streaming URLs we will need. Searching for your favorite stations might yield all of the results you need. I prefer to begin my search at **Radio Browser** (www.radio-browser.info). This crowd-sourced service attempts to possess direct audio streams for every radio station in the world. Searching your target streams' call letters, name, station, number, or location should get you results. My search of BBC immediately identified the URL we used in the previous example without much effort. This database powers many online streaming applications, including **Radio Garden** (radio.garden) and Radio Browser's own map of streams (radio-browser.info/map). We will use their API in our search tools next.

Hopefully, you now have an idea of the types of radio streams which you might like available to you when needed. Let's work through the first portion of the Live Audio Streams tool included with your offline search tools and online at **https://inteltechniques.com/tools/Radio.html**. The first four forms allow you to query a commercial radio station by search term, call letters, broadcast state, or broadcast country. These all query the Radio Browser database via their API and provide results as JSON text in a new tab. Search for the "url" of each result. Many results can be played directly within the browser. Those with "m3u8" at the end will need the HLS Player added to the URL as previously explained.

The next section includes a pre-configured "Radio News Stream" option. Selecting one of the streams and clicking "Play" will load the audio controls within the page and allow immediate streaming of the chosen content. Let's take a look at the source code for this section below.

```
<script type="text/javascript">
function dosearch2(menu2)
{window.open(menu2, 'inner');}
</script>
<form onsubmit="dosearch2(this.menu2.value); return false;">
<select name="menu2" style="width:200px">
<option>Radio News Stream</option>
<option value="https://www.hlsplayer.org/play?url=https://live-hls-audio-web-aje.getaj.net/VOICE-AJE/index.m3u8">Al Jazeera News</option>
<option value="http://stream.live.vc.bbcmedia.co.uk/bbc_world_service">BBC News</option>
<option value="https://www.hlsplayer.org/play?url=https://cbcradiolive.akamaized.net/hls/live/2041053/ES_R1SRE/master.m3u8">CBC News</option>
<option value="https://www.hlsplayer.org/play?url=https://live.amperwave.net/manifest/audacy-cbsnewsaac-hlsc.m3u8">CBS News</option>
<option value="http://radio.ukr.radio:8000/ur1-lv-mp3">Ukraine News</option>
</select>
<input type="submit" style="width:50px" value="Play" /></form>
```

The selection in bold is the area which determines which audio feeds will play when selected. Notice that selections such as the BBC do not need the HLS Player while Al Jazeera does. I included only a few entries here as a demonstration. You should consider modifying your own offline search tools to include the streams most valuable to you. I keep my local news stations ready to go. The quality is superior to the AM or FM broadcasts and I can easily stream from my mobile device if desired. The sky is the limit for your own tools.

Order 0138471223

Live Web-Controlled Radios

This may be my favorite portion of this chapter. If you have listened to my podcast or read my magazine, you know that I am fascinated by analog radio reception. Whether that means pulling in distant AM stations with my handheld radio or tuning into pirate shortwave broadcasts using an external outdoor antenna, there has always been something magical about eavesdropping on audio which was not meant to be heard in my location. While I still break out the portable radio on occasion, I rely most on internet-connected, remote-controlled radios located all over the world. Let's start with some manual access and control options before we automate our tasks within our custom tools.

I have mentioned **WebSDR** (websdr.org) within previous editions of this book. It allows you to connect to select analog radios via the internet and control the frequencies to which they are tuned. One can then listen to a radio broadcast exactly as it is heard from a specific location. I still visit WebSDR occasionally, but I now much prefer **KiwiSDR** (kiwisdr.com) for this purpose. We will start at their listing of publicly-accessible radios on their site at **kiwisdr.com/public**. At the time of this writing, there were 651 radio receivers online from all over the world.

The KiwiSDR is a software-defined radio (SDR) board that attaches to a BeagleBone miniature computer. It then connects to an internet connection for broadcast to the world. Most owners attach a high-quality indoor or outdoor antenna appropriate for their location. Most devices allow up to four simultaneous connections from the internet, and each visitor can control the tuning of the radio without impact to other listeners. These kind anonymous internet hosts provide a great service to us listeners.

While I am writing this, I am in Los Angeles. I can listen to practically any southern California radio station through any cheap receiver. However, assume I am interested in a broadcast from China or London. While commercial radio stations in those countries likely have an official audio stream, anything outside of the common AM and FM bands is not otherwise online. I can connect to these international radios; tune to any frequency between 1 KHz and 30,000 KHz; and hear exactly what I would receive if I were physically present at the radio's location. I find this astonishing. OK, let's get to the demonstration.

I randomly selected a radio in the UK The following URL connected me to receiver located in West Wales which possessed a Wellbrook ALA1530LN active loop antenna. You do not need to connect to this specific receiver, any other option available should appear exactly the same as my example.

http://sdrwales.hopto.org:8073/

Figure 23.05 displays the default view of most KiwiSDR interfaces. The first thing I like to do is to eliminate the purple popup "Welcome" menu, as it blocks my view and provides no value to me. Clicking the arrow in the upper right corner of that menu collapses it to the left side. This shows more of the waterfall display, which identifies frequencies which have active audio broadcasts. Seeing this overall 1 KHz to 30,000 KHz coverage immediately tells me how well this radio receives broadcasts within various bands.

Next, I tune to the desired frequency. Since the BBC broadcasts their live news service on the longwave spectrum at 198 KHz, I entered that within the frequency field (visible in the figure) and struck enter. This presented a crystal-clear live broadcast of the BBC longwave broadcast. However, many of these receivers default to a "LSB" or "Lower Side Band" mode. I prefer traditional "AM" mode, so I clicked the "AM" button for better sound. Next, I do not like the large banner at the top of the page, which is not visible within the figure. Clicking the grey "up" arrow should dismiss this. If the volume is low, sliding the volume bar up should correct the issue.

None of these actions are difficult, but they can become tedious when tuning to multiple frequencies during a session. Instead, I rely on keyboard shortcut keys for this. I typically type "xayyyVVVVVzzzzz" any time I connect to a stream. The following explains these keys.

x:	Closes all popup windows
a:	Selects the "AM" mode
yyy:	Removes all three banners from the top
VVVVV:	Amplifies the volume five times
zzzzz:	Zooms into the chosen frequency

The result of these actions can be seen in Figure 23.06. This presents a much cleaner view. You can see that I am tuned to a strong stream, but holding down alt and clicking the right arrow six times would take me to another longwave UK broadcast at 252 KHz. Clicking around would allow me to listen to longwave, AM, and shortwave broadcasts. The visible waterfall helps me locate strong streams without manually connecting to each frequency. Shift-z (uppercase "Z") zooms back out at any time.

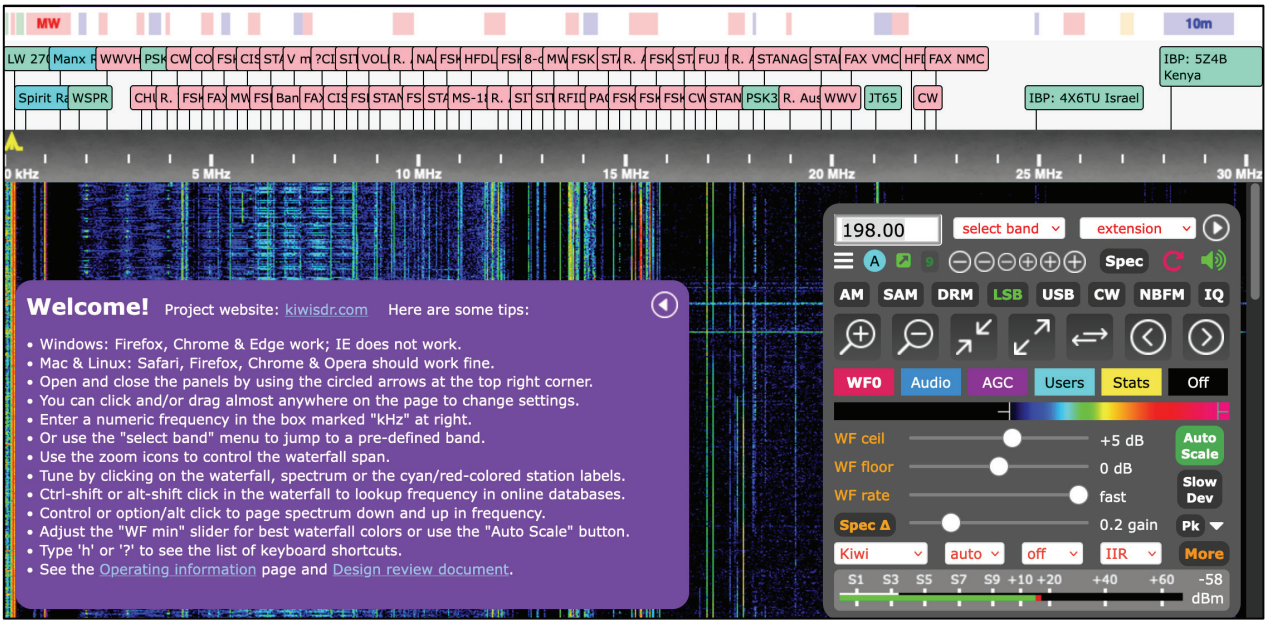


Figure 23.05: A default interface to KiwiSDR.

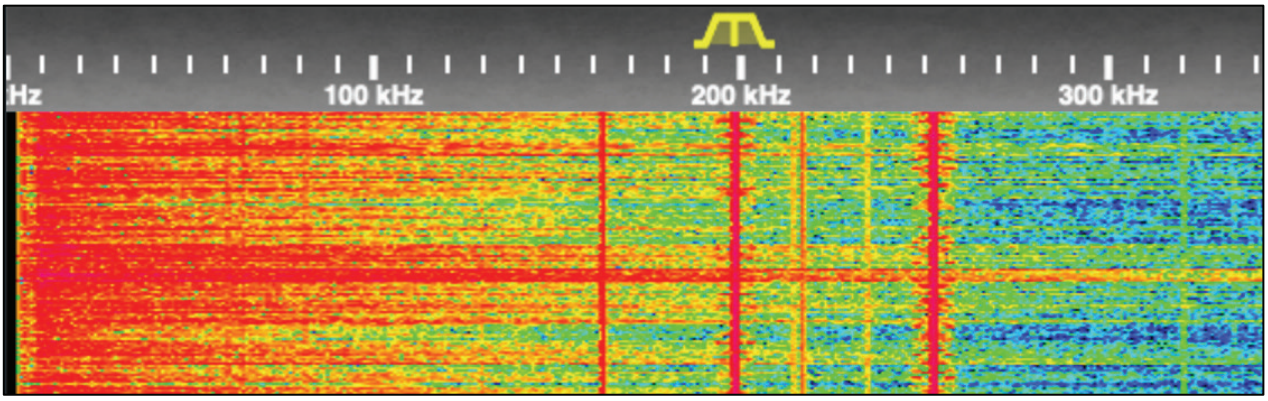


Figure 23.06: A cleaner interface to KiwiSDR.

To make things even easier, we can modify our URL to make all of these actions on our behalf. Using our original West Wales URL, the following URL connects to our desired radio; tunes to the BBC longwave frequency; closes all menus; eliminates all three banners; enhances the volume; and zooms into the band.

http://sdrwales.hopto.org:8073/?f=198am&keys=xyyyVVVVVzzzzz

Order 0138471223

You could bookmark this URL and immediately access this station with a minimal view at any time. You could replicate this and create URLs for numerous stations from unique locations. There are many options here. Before you commit, consider my online and offline search tools which you previously downloaded. I will use my Live Audio Streams tool at **https://inteltechniques.com/tools/Radio.html** as an example, but you could modify your own offline version if desired, as explained within Chapter Eight. While writing this chapter, I expanded the Remote Radio Access menu and selected a UK radio in Amersham. I then entered my desired frequency (198 KHz), and clicked "Play". The tool automatically conducted all of my desired tasks, and displayed the waterfall. You no longer need to remember the keyboard shortcuts or URL modifications. The tool should take care of everything. You could enter a different frequency or use your mouse to select strong signals. Of course, you can always replicate this from the main KiwiSDR site. I have simply chosen many international radios which work well for my needs.

I also find the site at **http://rx.linkfanel.net/** to be extremely helpful. It is linked within my tools and titled "Launch Live Shortwave Receiver Map". Figure 23.07 displays a partial view. This is a map version of all online KiwiSDR receivers. You can zoom into any location and select the receiver desired. This map also identifies areas which are currently in darkness, which typically aids in shortwave radio reception. As I write this, it is noon in California. That is a very bad time to listen to shortwave broadcasts due to sunlight propagation. However, it is nighttime in Australia, so I might want to connect to radios there if I want to find clear shortwave stations broadcasting to the entire world.

Figure 23.07: An international map of online KiwiSDR receivers.

I realize this can seem overwhelming at first. There are unlimited listening options which quickly change. Please allow me to explain how I take advantage of these receivers for fun and as part of my investigations. The fun part is obvious. I enjoy tracking down international shortwave broadcasts. I have heard many pirate stations on shortwave being broadcast from ships within international waters. These are often full of music, but I have also intercepted shows about hacking and disappearing from society. They also allow me to tune into a live KFI AM 640 stream when I am traveling internationally and need some familiar voices. There is something soothing about a real feed and not a digitally-compressed stream with commercials catered to my location. I want the real thing.

However, this is a book about online investigations, so I should stick to that. I find these radios to be vital to that need. I can tune into local radio stations which are broadcasting updates about current events. If I want to get a local take on the current situation in Ukraine, local stations in that area might give me more insight. A surprising number of these are in English. If I want to listen to a shortwave broadcast from China about

international affairs, it is a click away. Some of these receivers are also capable of receiving Citizens' Band (CB) frequencies. I can listen to individuals using their CB radios in practically any U.S. city from my home.

My main investigative use is HAM radio. These receivers allow access to all low and high amateur radio frequency bands. This allows me to listen to HAM operators from practically anywhere in the world. While a lot of this chatter is from hobbyists testing their equipment, I can also receive real-time updates about major events. If there is a wildfire or missing person rescue in an area, be assured that HAM operators are discussing updates unavailable to any news outlet. Is there police activity and road closures due to a barricaded subject? There is likely an amateur radio operator discussing the events in real-time. There is no limit to the power of this technique. I encourage you to familiarize yourself now, and everything will be easy when needed.

Live Web-Controlled Scanners

The receivers in the previous section cover everything from the bottom of the radio spectrum until 30 MHz (30,000 KHz). This will include most traditional radio broadcasts within longwave, AM, and shortwave. They will not receive anything within upper frequencies commonly used for emergency services, aviation, marine, or VHF amateur radio. For those, you would need a "scanner" type of radio, which typically begins at 25 MHz and can reach over 1 GHz. While we do not have scanner receivers which allow us to control the tuning, we do have numerous scanners which are pre-tuned and connected to the internet for our listening.

Broadcastify (broadcastify.com) is our best resource. This site has been a steady part of my arsenal for over a decade. It allows you to listen to live and archived streams of radio traffic from emergency transmissions. Broadcastify is the largest platform of live streaming audio from public safety, aircraft, rail, and marine related communications in the United States. From the "Listen" > "Browse Feeds" menu, you can drill down through states and counties to select your area of interest. From there, you rely on the willingness of internet strangers to share their audio feeds to the world. I can be in Los Angeles and listen to a stranger's police scanner monitoring frequencies in Chicago. I confess that on occasion, I listen to the police department which hired me in the 90's to hear what my former colleagues are doing. If you are investigating any event in real-time, this can be a wonderful free service.

I recently heard about an active shooter within a neighborhood where a few friends live. Instead of waiting for tomorrow's copy of the local newspaper to know what happened, I tuned into the live police frequencies for that district. I heard all updates in real-time, and could let my friends know. I was recently stuck in traffic due to a road closure and a lot of police activity. Connecting through my mobile device let me know that a bad accident had shut down the highway and that a specific alternate route was being planned. This allowed me to jump ahead of the crowd and be on my way. However, the real power is in the archives.

Every scanner broadcast is recorded at all times. A paid membership provides access to these archives. This has been a huge benefit to my investigations. I can go back in time to hear the original dispatch and all follow-up radio transmissions. Even better, I can do this without a subpoena or FOIA request. I have been a scanner enthusiast for much of my life, but this takes everything to a new level. In the early 90's, I would have never suspected that I could hear all frequencies at any time and "rewind" communications from anywhere.

A premium membership is $30 annually. This provides commercial-free listening; non-stop streaming; access to archives; custom listening templates; and programming functions through scanner software. The free tier allows limited streaming with commercials and manual query of frequencies. I know many people who get by with the free version. The first time you need access to the archives, the cost of membership will likely be justified. This service allows journalists to play dispatch recordings without waiting on a public release from a government agency. The scanner section of my Live Audio Streams tool allows you to select a state or country and immediately be taken to the Broadcastify page, which allows you to listen to any online streams or archives.

Order 0138471223

CHAPTER TWENTY-FOUR
DOMAIN NAMES

A specific web page may quickly become the focus of your investigation. Websites, also known as domains, are the main sites at specific addresses. For example, the website that hosts my blog, www.inteltechniques.com/blog, is on the domain of inteltechniques.com. The "www" or anything after the ".com" is not part of the domain. These addresses should always be searched for additional information. If your target has a blog at a custom domain, such as privacy-training.com, the content of the site should obviously be examined. However, digging deeper into the domain registration and associated connections can reveal even more information. Every time I encounter a domain involved in my investigation, I conduct full research on every aspect of the domain, including historical registration, visual depiction, and hidden content. I rely heavily on the automated tools presented at the end of the chapter to make the work easy.

Current Domain Registration & Hosting

Every website requires information about the registrant, administrative contact, and technical contact associated with the domain. These can be three unique individuals or the same person for all. The contact information includes a full name, business name, physical address, telephone number, and email address. These details are provided by the registrar of the domain name to the service where the name was purchased. This service then provides these details to Internet Corporation for Assigned Names and Numbers (ICANN). From there, the information is publicly available and obtained by hundreds of online resources. While ICANN declares that the provided information is accurate, this is rarely enforced. While most businesses supply appropriate contacts, many criminals do not. While we must consider searching this publicly available information, often referred to as Whois details, we will also need to dig deeper into domain analysis in order to obtain relevant results. First, we will focus on the easy queries.

Whois queries (pronounced "who is") are very simple searches, but are not all equal. While this data is public, it could change often. Some Whois search sites display the bare bones details while others provide enhanced information. There are dozens of options from which to choose, and I will explain those that I have found useful. After the demonstrations, I present my own custom online tool that automates many processes.

The biggest hurdle with Whois data is privacy controls. Many domain owners have started using private registration services in order to protect their privacy. These services provide their own data within the Whois search results, and only these companies know the true registrant. While a court order can usually penetrate this anonymity, I will discuss public resources to help in these situations later. Some web hosts now provide free masking services for domain owners, which is making this problem worse. Historical record queries will be vital if you see this happening. Until then, let's focus on current registration data. For the first example, I will use a target domain of cnn.com. Assume that this website is the focus of your investigation and you want to retrieve as much information as possible about the site, the owner, and the provider of the content. For the standard Whois search, as well as many other options, I prefer ViewDNS.info.

ViewDNS Whois (viewdns.info/whois)

This service provides numerous online searches related to domain and IP address lookups. Their main page (viewdns.info) provides an all-in-one toolbox, but the above website connects you directly to their Whois search. Entering cnn.com here presents the following information.

Domain Name: cnn.com
Updated Date: 2020-10-20T13:09:44Z
Creation Date: 1993-09-22T00:00:00.000-04:00
Admin Organization: Turner Broadcasting System, Inc.

Admin Street: One CNN Center
Admin City: Atlanta
Admin State/Province: GA
Admin Postal Code: 30303
Admin Country: US
Admin Phone: +1.4048275000
Admin Fax: +1.4048271995

The administrative and technical contacts were identical to the registrant shown above. This data identifies the company which owns site and associated contact details. We also know it was created in 1993 and the record was updated in 2020. This is a great start, if the provided details are accurate. I have found that ViewDNS will occasionally block my connection if I am connected to a VPN. An alternative Whois research tool is who.is.

ViewDNS Reverse IP (viewdns.info/reverseip)

Next, you should translate the domain name into the IP address of the website. ViewDNS will do this, and display additional domains hosted on the same server. This service identified the IP address of cnn.com to be 151.101.1.67 and stated the web server hosted dozens of additional domains related to this company. These included weather.cnn.com and cnnbusiness.com. If the results had included domains from websites all over the world without a common theme, it would have indicated that this was a shared server, which is very common.

ViewDNS Reverse Whois (viewdns.info/reversewhois)

This utility attempts to search the domain in order to locate other domains owned by the same registrant. In our example, it located 28 additional domains associated with our target. If the domain possessed private registration, this technique would fail.

ViewDNS Port Scanner (viewdns.info/portscan)

This online port scanner looks for common ports that may be open. An open port indicates that a service is running on the web server that may allow public connection. A search of cnn.com revealed that ports 21, 80, and 443 are open to outside connections. Port 80 is for web pages and port 443 is for secure web pages. These are open on practically every website. However, port 21 is interesting. ViewDNS identifies this as a port used for FTP servers, as was discussed previously. This indicates that the website hosts an FTP server and connecting to ftp.cnn.com could reveal interesting information.

ViewDNS IP History (viewdns.info/iphistory)

This tool translates a domain name to IP address and identifies previous IP addresses used by that domain. A search of cnn.com reveals the following details. The first column is the IP address previously associated with the domain, the second column identifies the current user and company associated with that IP address, and the last column displays the date these details were collected by ViewDNS.

IP Address	Location	IP Address Owner	Last seen
151.101.65.67	United States	RIPE NCC	2020-11-23
151.101.193.67	United States	RIPE NCC	2020-11-23
151.101.1.67	United States	RIPE NCC	2020-11-23
157.166.255.19	United States	Turner Broadcasting System, Inc.	2011-04-04
157.166.255.18	United States	Turner Broadcasting System, Inc.	2011-04-04
157.166.226.25	Atlanta	Turner Broadcasting System, Inc.	2011-04-04

Order 0138471223

ViewDNS DNS Report (viewdns.info/dnsreport)

This option presents a complete report on the DNS settings for the target domain. This tool is designed to assist webmasters and system administrators diagnose DNS related issues, but we can use it to peek into their settings, including DNS and mail server details.

The utilities hosted at ViewDNS are always my first stop for a couple of reasons. First, the site has been very reliable over the past ten years. More vital, it allows query via static URL. This is beneficial for submission directly from our tools. The following displays the URL structure for the previous techniques, with cnn.com as the target.

https://viewdns.info/whois/?domain=cnn.com
https://viewdns.info/reverseip/?host=cnn.com&t=1
https://viewdns.info/reversewhois/?q=cnn.com
https://viewdns.info/portscan/?host=cnn.com
https://viewdns.info/iphistory/?domain=cnn.com
https://viewdns.info/dnsreport/?domain=cnn.com

Historical Domain Registration

As stated previously, many domains now possess private registration. This means that you cannot see the owner of a domain publicly. Many hosts are now offering private registration as a free service, further complicating things for investigators. If you query a domain and see a name entry such as "WhoisGuard Protected", you know that the domain is protected. There are two ways to defeat this. The first requires a court order, which is outside the scope of this book. The second way to reveal the owner is through historical domain records. If the domain has been around a while, there is a very good chance that the domain was not always private. There are several free and paid domain history services, and I present my favorites in order of usefulness.

Whoxy (whoxy.com)

This is one of the very few premium services which offer a decent free tier. Searching my own domain, which currently possesses private registration, reveals valuable results. The general registration data was "protected" and confirms what I found on ViewDNS. Scrolling down the page reveals powerful historical records. These identify my real name, multiple email addresses used during various registrations, and a date next to each entry to tie it all together. Figure 24.01 displays one of the results. This confirms that during July of 2015, my site was briefly registered without privacy protection. The "9 Domains" link reveals even more information. Figure 24.02 displays this result which identifies numerous domains which I had previously created from 2007 through 2018. This resource has single-handedly exposed more private domain registrations than any other free service during my investigations throughout 2022. Furthermore, it allows submission via URL which will benefit our search tools, as follows.

https://www.whoxy.com/inteltechniques.com

The search option in the upper right allows us to query email addresses, names, and keywords. This can be extremely valuable when you do not know which domain names your target has owned. Searching "OSINT" reveals a surprising amount of people and companies purchasing domains associated with this topic. Whoxy offers many paid services if the free tier is not sufficient. Of the paid services, this one is the most affordable, allowing you to purchase small amounts of information without committing to any specific level of subscription.

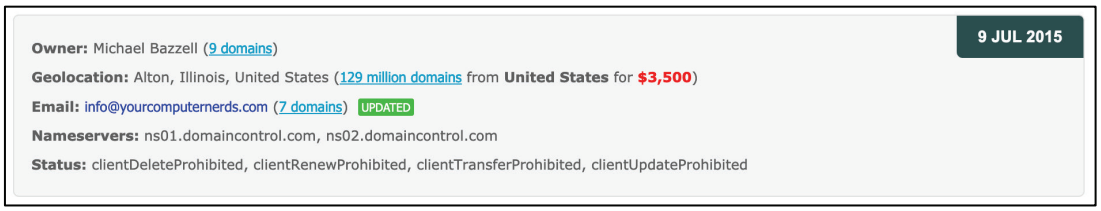


Figure 24.01: A Whoxy historical domain registration result.

DOMAIN NAME	REGISTRAR	CREATED	UPDATED	EXPIRY
inteltechniques.com	NameCheap, Inc.	21 Jul 2013	14 Apr 2018	21 Jul 2019
inteltechniques.online	GoDaddy.com, LLC	31 Mar 2018	31 Mar 2018	31 Mar 2019
privacy-training.com	GoDaddy.com, LLC	11 Jun 2016	31 May 2017	11 Jun 2018
yourcomputernerds.com	GoDaddy.com, LLC	18 Dec 2007	20 Oct 2015	18 Dec 2016
missouriinternationalexports.com	GoDaddy.com, LLC	1 Apr 2008	25 Mar 2016	1 Apr 2017
illinoisinternationalexports.com	GoDaddy.com, LLC	1 Apr 2008	25 Mar 2016	1 Apr 2017
humphreyinternationalexports.com	GoDaddy.com, LLC	1 Apr 2008	25 Mar 2016	1 Apr 2017

Figure 24.02: Additional domains owned by a target from Whoxy.

Whoisology (whoisology.com)

This service appeared in 2014 and becomes more powerful every month. Like Whoxy, it provides historical domain records as a reverse-domain search utility. The home page of Whoisology presents a single search field requesting a domain or email address. Entering either of these will display associated websites and the publicly available Whois data. This is where the publicly available free features end. Any further details require registration. I encourage all readers to create a free account. Once logged in as a free user, you receive much more detail within your searches. The first basic feature that you see is the display of standard Whois data which will identify the registered administrative contact, registrant contact, technical contact, and billing contact. These will often be the same individual for most personal websites. The advanced feature within this content is the ability to immediately search for additional domains associated within any field of this data. As an example, a search for the domain of phonelosers.org reveals the following data.

Name	Brad Carter (88)
Email	brad@notla.com (7)
Street	PO Box 465 (1,091)
City	Albany (42,428)
Region	Oregon (492,506)
Zip / Post	97321 (3,080)
Phone	8144225309 (4)

The name, address, and other data can be found on any Whois search website. However, the numbers in parentheses identify the number of additional domains that match those criteria. In this example, there are a total of 88 domains registered to Brad Carter, and seven domains registered to the email address of brad@notla.com. Clicking on any of these pieces of data will launch a new page with all of the matching domain information. As an example, clicking on brad@notla.com will display the 7 domain names associated with his email address. Clicking 8144225309 will display the 4 domain names associated with his telephone number. One of these is a new domain that is not directly associated with him. However, since it was registered with the same phone number, there is now a connection.

Order 0138471223

This type of cross-reference search has not been found through many other services. Another powerful feature of Whoisology is the historical archives. This service constantly scans for updates to domain registrations. When new content is located, it documents the change and allows you to search the previous data. As an example, a search of computercrimeinfo.com reveals the current administrative contact telephone number to be 6184628253. However, a look at the historical records reveals that on October 16, 2012, the domain contact number was 6184633505. This can be a great way to identify associated telephone numbers that have since been removed from the records. Whoisology will also provide details from the search of an email address. In my experience, Whoisology will provide a more detailed and accurate response than most other resources. However, this comes at a cost. Whoisology continues to minimize the free options in order to maximize sales. Your usage may vary by the time you try the free services. A search of the email address brad@notla.com revealed the following domains associated with that account.

notla.com
phonelosers.com
bigbeefbueno.com
callsofmassconfusion.com
snowplowshow.com
phonelosers.org

If you ever encounter an investigation surrounding a domain or any business that possesses a website, I highly encourage you to conduct research through Whoxy and Whoisology. They also offer access through their API at a cost. The individual queries through their website are free. Whoisology restricts free accounts to only one historical record and three searches every 24 hours. Because of this, Whoxy receives my overall recommendation. I believe Whoisology offers the most data for those looking to purchase a subscription in order to query numerous domains, but it can be quite expensive. **WhoisXMLAPI** (whoisxmlapi.com) offers limited free trial access to their historical domain data, but it requires access through their API, as explained later within Chapter Twenty-Eight.

Archive.org Domain Registration Data (web.archive.org)

You now know there are many ways to identify the owner of a domain through historical Whois captures. You might get lucky through a free preview of previous registration data or be required to purchase access in order to see all records dating back decades. We have one last free option which has been successful for me throughout several investigations. We can query the Wayback Machine for the exact historical URL of a domain registration. If you navigate directly to https://who.is/whois/phonelosers.org, you can see that the domain possesses WhoisGuard protection and the owner's identity is masked. This is displayed in Figure 24.03 (left). However, it is possible that the Wayback Machine captured this exact page. The following URL displays any results.

https://web.archive.org/web/http://who.is/whois/phonelosers.org

This URL defaults to a capture from 2017 which displays the owner's name, address, telephone number, and email. This can be seen in Figure 24.03 (middle). Clicking the earliest archive presents a capture of this data from 2010, as seen in Figure 24.03 (right). We now have accurate historical domain registration data without a premium membership, and two additional telephone numbers to investigate. This is not the only domain registration service indexed by archive.org. The following direct links query domain registration history from Who.is, Domain Tools, Whoxy, Domain Big Data, and Whoisology. Replace cnn.com with your target domain. The domain tools presented at the end of this chapter replicate each of these options.

https://web.archive.org/web/http://www.who.is/whois/cnn.com/
https://web.archive.org/web/https://whois.domaintools.com/cnn.com
https://web.archive.org/web/https://www.whoxy.com/cnn.com
https://web.archive.org/web/https://domainbigdata.com/cnn.com
https://web.archive.org/web/https://whoisology.com/cnn.com

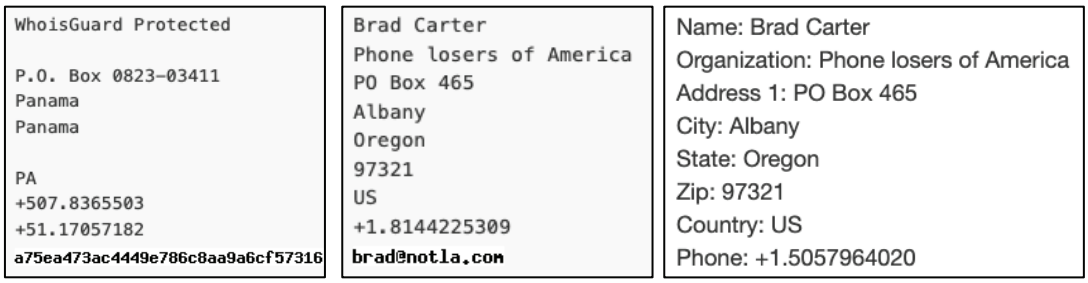


Figure 24.03: Results from who.is (left) and the Wayback Machine (middle and right).

Historical Content Archives

I previously mentioned The Internet Archive and explained how it collects copies of websites over time. This is very useful when we need to see the ways in which a website has changed. This is especially true if our target domain is no longer online. The direct URL for a domain query at their service is as follows.

http://web.archive.org/web/*/cnn.com

This view presents a graphical way to browse through various archives at a specified domain. However, we can also conduct a URL-based query which presents all full URLs which have been archived by the service. This text-only view can be accessed with the following URL structure.

https://web.archive.org/cdx/search/cdx?url=cnn.com/*&output=text&fl=original&collapse=urlkey

The result from this URL is tens of thousands of URLs, many of which no longer exist on the domain, but can be accessed via The Internet Archive. This text view is easy to copy and paste into a report.

If you queried my website, IntelTechniques.com, you should be notified that my domain is not indexed by their service. This is because I placed code within my site requesting the pages to be ignored by The Internet Archive's indexing engine. However, this request is ignored by other providers. Archive.org is not the only service which provides archived domains.

Archive Today (archive.is / archive.fo / archive.md)

This service also collects copies of websites and ignores all requests for deletion. My own site can be found at the following URL, which displays 183 captures dating back to 2013.

https://archive.is/*.inteltechniques.com

Mementoweb (mementoweb.org)

This service offers a "Time Travel" option which presents archives of a domain from several third-party providers. The following URL queries my own domain, which found 5 new archives.

http://timetravel.mementoweb.org/list/19991212110000/http://inteltechniques.com

Library of Congress (webarchive.loc.gov)

This option allows you to search by domain to discover all publicly available content in the Library of Congress Web Archives. A direct URL is as follows.

https://webarchive.loc.gov/all/*/http://inteltechniques.com

Order 0138471223

Portuguese Web Archive (arquivo.pt)

I was surprised to see my site indexed here with 12 historic archives dating back to 2017. The query URL structure is as follows.

https://arquivo.pt/page/search?hitsPerPage=100&query=site%3Ainteltechniques.com

Historical Screen Captures

While it may seem obvious, you should document the current visual representation of your target website. This may be a simple screen capture, but you may have advanced needs. When I am investigating a domain, I have three goals for visual documentation. I want documentation of the current site, any historical versions archived online, and then future monitoring for any changes. Previous chapters presented numerous methods for capturing live web pages and techniques for locating online archives. Let's briefly revisit these methods and then apply new resources.

Search Engine Cache

Searching your target domain within Google, Bing, and Yandex should present numerous results. The first is almost always the home page, and clicking the green down arrow usually presents a cached version of the page. This should be conducted first in order to identify any recent cached copies, as previously explained. In my experience, Google is going to possess the most recent cache, which can always be accessed via direct URL with the following structure.

https://webcache.googleusercontent.com/search?q=cache:inteltechniques.com

Website Informer (website.informer.com)

I am mostly interested in the screen capture available to the right of a search result. Figure 24.04 (upper left) displays their capture of my site in May of 2019.

URLScan (urlscan.io)

Similar to the previous option, this service provides very little valuable details. However, the screen captures are often unique. Figure 24.04 (upper middle) displays a result from August of 2018.

Easy Counter (easycounter.com)

The screen capture presented here was very similar to Website Informer, but it was cropped slightly different. This appears to be from May of 2019 and is seen in Figure 24.04 (upper right).

Domain Tools (whois.domaintools.com)

These screen captures are in high resolution and current. Figure 24.04 (lower left) displays the result from my domain captured in August of 2022.

Domains App (dmns.app)

This service offers the highest resolution image, which is very high quality. However, you must know the exact URL to receive a result. The following URL structure reveals a full-page screen capture of my site from April 2022. Figure 24.04 (lower middle) displays the result.

https://files.dmns.app/screenshots/inteltechniques.com.jpg

Hype Stat (hypestat.com)

Finally, the lowest-quality option we will receive is from Hype Stat. This is typically an older image. Figure 24.04 (lower right) displays their selection for my site from 2020.

The following direct URLs will be used for our search tool, displaying my own site as an example.

https://website.informer.com/inteltechniques.com#tab_stats
https://urlscan.io/domain/inteltechniques.com
https://www.easycounter.com/report/inteltechniques.com
https://whois.domaintools.com/inteltechniques.com
https://files.dmns.app/screenshots/inteltechniques.com.jpg
https://hypestat.com/info/inteltechniques.com

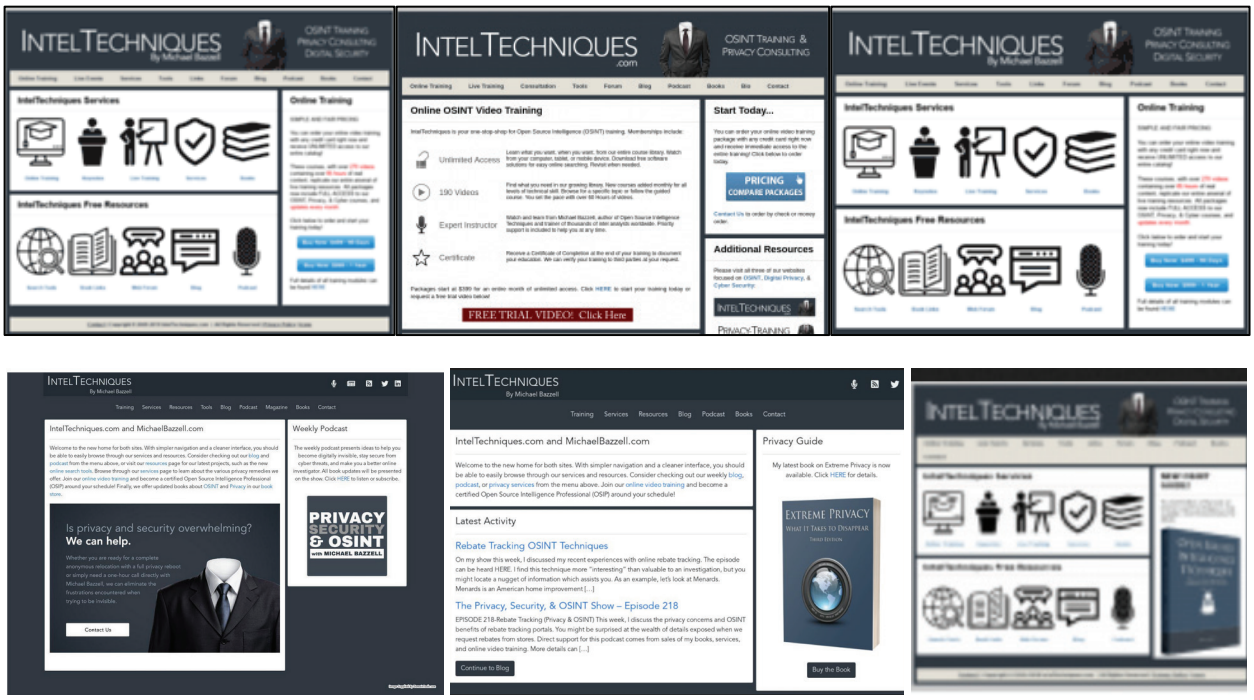


Figure 24.04: Historical screenshot captures from six resources.

Carbon Dating (carbondate.cs.odu.edu)

This free service provides a summary of available online caches of a website, and displays an estimated site creation date based on the first available capture. The following static URL would be used to query my own domain.

http://carbondate.cs.odu.edu/#inteltechniques.com

This provided an estimated creation date of "2013-11-16T08:47:11". Furthermore, it displayed multiple direct URLs to historical archives of my website within the Wayback Machine and other services. After checking these links and visiting any other archives at the Internet Archive, the following services fill in the gaps with screen captures from various dates.

Follow That Page (followthatpage.com)

Once you locate a website of interest, it can be time consuming to continually visit the site looking for any changes. With large sites, it is easy to miss the changes due to an enormous amount of content to analyze. This is when sites like Follow That Page come in handy. Enter the address of the target page of interest, as well as an email address where you can be reached. This service will monitor the page and send you an email if anything changes. Anything highlighted is either new or modified content. Anything that has been stricken through indicates deleted text. Parents are encouraged to set-up and use these services to monitor their child's websites. It does not work well on some social networks such as Facebook, but can handle a public Twitter page fine.

Visual Ping (visualping.io)

If you found the service provided by Follow That Page helpful, but you are seeking more robust options, you should consider Visual Ping. This modern Swiss website allows you to select a target domain for monitoring. Visual Ping will generate a current snapshot of the site and you can choose the level of monitoring. I recommend hourly monitoring and notification of any "tiny change". It will now check the domain hourly and email you if anything changes. If you are watching a website that contains advertisements or any dynamic data that changes often, you can select to avoid that portion of the page. Figure 24.05 displays the monitoring option for phonelosers.org. In this example, I positioned the selection box around the blog content of the main page. I also chose the hourly inspection and the "Tiny Change" option. If anything changes within this selected area, I will receive an email announcing the difference.

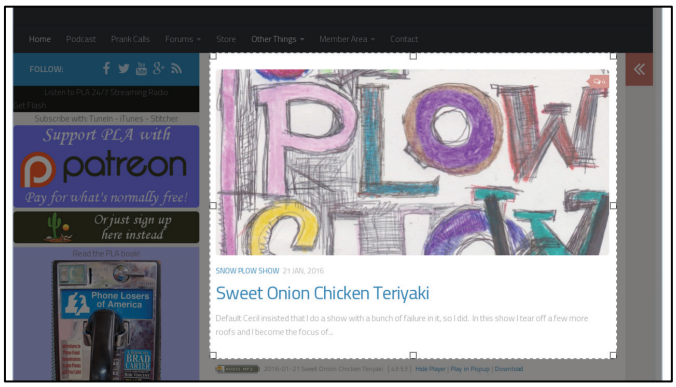


Figure 24.05: A portion of a web page monitored by Visual Ping for changes.

Email Address Identification

Lately, I am eager to identify any email addresses associated with a specific domain. This can lead to the discovery of employees and can be used for further breach data queries. We now have three reliable resources which should be queried any time we are investigating a domain name.

Hunter (hunter.io)

Previously, I explained how Hunter could be used to verify email addresses. This tool can also accept a domain name as a search term, and provides any email addresses that have been scraped from public web pages. The free version of this tool will redact a few letters from each address, but the structure should be identifiable. The Hunter website should take you where you need to be, but I find myself getting caught in Captcha loops and trial restriction notices. I now submit queries from the following URL structure.

https://hunter.io/try/search/cnn.com?locale=en

The result is a list of email addresses with minor redactions. It also includes the online source of the information which was scraped by Hunter. The following is an example.

http://edition.cnn.com/specials/cnn-investigates-team Jun 15, 2022
j.martin@cnn.com**
http://cnnpressroom.blogs.cnn.com/2011/05/17/cnn-launches-international-news-app May 1, 2022
cre.duffy@cnn.com**

Website Informer (website.informer.com)

Once I have exhausted Hunter, I switch to Website Informer. The results are typically fewer, but they are not redacted. The URL submission I use follows.

https://website.informer.com/cnn.com/emails

SkyMem (skymem.info)

Finally, this service also displays full email addresses which are not redacted. The following URL allows our search tool to automate the process.

https://www.skymem.info/srch?q=cnn.com

As with most OSINT tools, these three services will likely never provide the same results. This is why it is so important to exhaust all options in order to acquire the most data possible.

Blacklight (themarkup.org/blacklight)

You may want to know about any malicious activity embedded into a target website. Blacklight displays this data. When querying cnn.com, I am notified of the following invasive activity.

34 Advertisement Trackers
52 Third-Party Cookies
Notifications to Facebook
Interactions with Adobe, Google, and others

Domain Analytics

Domain analytics are commonly installed on websites in order to track usage information. This data often identifies the city and state from where a visitor is; details about the web browser the person is using; and keywords that were searched to find the site. Only the owner of the website can view this analytic data. Analytics search services determine the specific number assigned to the analytics of a website. If the owner of this website uses analytics to monitor other websites, the analytic number will probably be the same. These services will now conduct a reverse search of this analytic number to find other websites with the same number. In other words, it will search a website and find other websites that the same owner may maintain. Additionally, it will try to identify user specific advertisements stored on one site that are visible on others. It will reverse search this to identify even more websites that are associated with each other. None of this relies on Whois data. A couple of examples should simplify the process.

Spy On Web (spyonweb.com)

Spy On Web is one of many sites that will search a domain name and identify the web server IP address and location. It also conducts a Whois query which will give you registration information of a website. More importantly, it identifies and cross-references website analytic data that it locates on a target domain. A search

Order 0138471223

for the website phonelosers.org reveals a "Google AdSense" ID of ca-pub-3941709854725695. It further identifies five domains that are using the same Google AdSense account for online advertising. This identifies an association between the target website and these new websites. We now know that whoever maintains phonelosers.org places ads on the page. We also know that those same ads and affiliate number is present on five domains. This means that our target likely maintains all of the following domains.

www.notla.com
www.oldpeoplearefunny.com
www.phonelosers.com
www.phonelosers.org
www.signhacker.com

Analyze ID (analyzeid.com)

While Spy On Web is a strong overall analysis tool, there are additional options that should be checked for reverse analytics. Analyze ID performs the same type of query and attempts to locate any other domains that share the same analytics or advertisement user numbers as your target. This will provide new websites related to your target. During a search of phonelosers.org, it identified the Google AdSense ID previously mentioned. It also revealed an Amazon Affiliate ID of phonelosersof-20 and an Amazon product ID of 1452876169. These are likely present because the target sells books through his websites and receives compensation from Amazon through customer purchases. These new pieces of information can be very valuable. Clicking the Amazon Affiliate ID presents the five domains that possess that same code. The Amazon product ID displays the seven websites that also advertise a specific product. When you encounter this type of data, consider the following Google search. It identifies the actual product that the target is selling within embedded ads. This search reveals that the product is the target's self-published book titled Phone Losers of America.

Amazon product 1452876169

Analyze ID also identified a Clickbank affiliate ID that was associated with five additional domains, one of which was unique to any other results. Spy On Web and Analyze ID have provided us several new pieces of information about our target. If you ever see an ID that starts with "UA-", this is likely an identifier for Google to monitor viewers of the website. Searching that number within these tools will also identify related websites. We should visit each website and analyze the content for any relevant evidence.

Hacker Target (hackertarget.com/reverse-analytics-search)

This service stands out due to their immediate availability of historic analytics IDs across multiple subdomains. Consider a search of cnn.com. Hacker Target displayed dozens of associated Google IDs, such as the following.

cnnworldlive.cnn.com,UA-289507
center.cnn.com,UA-96063853
games.cnn.com,UA-74063330

We could search these IDs through this or other sites to track additional associated domains.

DNSLytics (dnslytics.com/reverse-analytics)

The final analytics option I offer is DNSLytics. Enter the analytics ID found with any of the previous techniques and you may find associations not present within other options. The two direct URL options follow.

https://dnslytics.com/reverse-analytics/inteltechniques.com
https://dnslytics.com/reverse-adsense/inteltechniques.com

SSL Certificates

Most target websites will possess an SSL certificate which allows for a secure connection through HTTPS. The history of these certificates can be quite lucrative. Consider my own domain through a free service called **CRT.sh** (crt.sh) through a direct URL of https://crt.sh/?q=inteltechniques.com. Most of the data identifies various certificate updates which do not provide any valuable information. However, through the history you can see that I once purchased a group SSL certificate from GoDaddy. Clicking on any of these entries displays the other domains I secured with this option, including dates of activity, as follows. This identifies other domains to investigate.

inteltechniques.com
privacy-training.com
yourcomputernerds.com
computercrimeinfo.com

OSINT.sh (osint.sh)

This multiple-use service offers a lot to digest. I find the following direct URLs most beneficial.

https://osint.sh/subdomain: Display all subdomains of a target domain
https://osint.sh/stack: Display all technologies in use by a target domain
https://osint.sh/email: Display all email addresses publicly associated with a target domain
https://osint.sh/ssl: Display all SSL certificates associated with a target domain
https://osint.sh/whoishistory: Display historic registrations associated with a target domain
https://osint.sh/analytics: Display all domains associated with a Google Analytics ID
https://osint.sh/adsense: Display all domains associated with a Google AdSense ID
https://osint.sh/domain: Display all domains associated with keywords
https://osint.sh/reversewhois: Display all domains publicly associated with an email address
https://osint.sh/ocr: Extract text found within a document stored at a target domain URL

Website Source Code

Every website possesses valuable information "under the hood" within its source code. We should always search within this code and understand the various technologies used to create the site. For this, I rely on the following services.

Nerdy Data (search.nerdydata.com)

Nerdy Data is a search engine that indexes the source code of websites. I use this to locate websites which steal the source code of my search tools and present them as their own. In one example, I searched "function doSearch01(Search01)", which is the JavaScript I use within the search tools. Nerdy Data immediately identified a competitor's website which was selling access to my scripts.

If you have located a Google Analytics ID, AdSense ID, or Amazon ID of a website using the previous methods, you should consider searching this number through Nerdy Data. A search of our target's Google AdSense number revealed five domains that possess the same data. The search of the Amazon number revealed three domains. If this service presents more results than you can manage, consider using their free file download option to generate a csv spreadsheet.

Real World Application: While investigating an "anonymous" website that displayed photo evidence of a reported felony, I discovered that the registration information was intentionally inaccurate. A search of the website on these services identified a Google Analytics ID and an additional website that possessed the same number. That additional website was the personal blog of the suspect. An arrest was made the same day.

Order 0138471223

Built With (builtwith.com)

A quick analysis of a target website may identify the technologies used to build and maintain it. Many pages that are built in an environment such as WordPress or Tumblr often contain obvious evidence of these technologies. If you notice the YouTube logo within an embedded video, you will know that the creator of the site likely has an account within the video service. However, the presence of various services is not always obvious. Built With takes the guesswork out of this important discovery.

Entering the domain of phonelosers.org into the Built With search immediately identifies the web server operating system (Linux), email provider (DreamHost), web framework (PHP, WordPress), WordPress plugins, website analytics, video services, mailing list provider, blog environment, and website code functions. While much of this is geek speak that may not add value to your investigation, some of it will assist in additional search options through other networks. Another option for this type of search is **Stats Crop** (statscrop.com).

Subdomains

A subdomain is a domain that is a part of another domain. For example, if a domain offered an online store as part of their website example.com/shop.html, they might use a subdomain of shop.example.com. A subdomain finder is a tool which performs an advanced scan over the specified domain and tries to find as many subdomains as possible. This often discloses interesting evidence otherwise not visible within the main page. I rely on several tools.

PentestTools (https://pentest-tools.com/information-gathering/find-subdomains-of-domain)

This unique tool performs several tasks that will attempt to locate hidden pages on a domain. First it performs a DNS zone transfer which will often fail. It will then use a list of numerous common subdomain names and attempt to identify any that are present. If any are located, it will note the IP address assigned to that subdomain and will scan all 254 IP addresses in that range. In other words, it will attempt to identify new areas of a website that may not be visible from within the home page. The following example may help to clarify.

The website at phonelosers.org is a blog that appears to have no further content to be analyzed. Searching for it on Pentest-Tools provides additional intelligence. It identifies the following subdomains present the server.

webmail.phonelosers.org	ssh.phonelosers.org	ftp.phonelosers.org
www.phonelosers.org	mail.phonelosers.org	

We now know that this domain possesses a webmail server, SSH connection, FTP server, and mail server. This method has helped me locate "hidden" pages which contain several forum messages from users of the site.

Columbus Project (columbus.elmasy.com)

This new service emerged in late 2022, and is the fastest of all present options. The web-based search should suit most users, but they also offer an API via Terminal with the following command, replacing "cnn.com" with your target domain.

curl -H "Accept: text/plain" "https://columbus.elmasy.com/lookup/cnn.com"

I find myself using the API via terminal when I have numerous domains to query and I want to easily document the results. I can add " > cnn.com.txt" at the end of the command to output a text file with all found subdomains. If these options do not provide the results you need, consider **SubDomain Finder** (subdomainfinder.c99.nl) and **DNS Dumpster** (dnsdumpster.com). These services rely on Host Records from the domain registrar to display potential subdomains. While searching a target domain related to a stalking investigation, it displayed a blog hidden from the main domain. This presented more information than I could easily digest.

Robots.txt

Practically every professional website has a robots.txt file at the "root" of the website. This file is not visible from any of the web pages at the site. It is present in order to provide instructions to search engines that crawl the website looking for keywords. These instructions identify files and folders within the website that should not be indexed by the search engine. Most engines comply with this request, and do not index the areas listed. Locating this file is relatively easy. The easiest way to view the file is to open it through a web browser. Type the website of interest, and include "robots.txt" after a forward slash (/). The file for CNN can be found at the following address, with partial contents underneath.

http://www.cnn.com/robots.txt

Disallow: /cnnbeta
Disallow: /development
Disallow: /partners

The file identifies online folders which include a new beta website, temporary development page, and list of their partners. Much of this content would not be found in a search engine because of the "Disallow" setting. These Disallow instructions are telling the search engines to avoid scanning the folders "cnnbeta", "development", and "partners". It is likely that there is sensitive information in these directories that should not be available on Google or Bing. You can now type these directories after the domain name of your target to identify additional information. Based on the robots.txt file in this demonstration, typing the following addresses directly into a browser may generate interesting results.

https://www.cnn.com/cnnbeta
https://www.cnn.com/development
https://www.cnn.com/partners

Most robots.txt files will not identify a secret area of a website that will display passwords, raunchy photos, or incriminating evidence. Instead, they usually provide insight into which areas of the site are considered sensitive by the owner. If you have a target website and have exhausted every other search method, you should also visit this file. It may direct you toward a new set of queries to find data otherwise ignored by search engines.

If this technique produces no results, you can conduct a Google or Bing query to identify any files. A search of "site:cnn.com robots ext:txt" on either search engine identifies robots.txt files from the entire website. We can also query the Wayback Machine to display changes of this file over time at the following URL structure.

https://web.archive.org/web/*/cnn.com/robots.txt

Search Engine Marketing Tools

The ultimate goal of most commercial websites is to generate income. These sites exist to bring in new customers, sell products, and provide the public face to a company. This has created a huge community of services that aim to help companies reach customers. Search Engine Optimization (SEO) applies various techniques affecting the visibility of a website or a web page in a search engine's results. In general, the higher ranked on the search results page and more frequently a site appears in the search results list, the more visitors it will receive. Search Engine Marketing (SEM) websites provide details valuable to those responsible for optimizing their own websites. SEM services usually provide overall ranking of a website; its keywords that are often searched; backlinks; and referrals from other websites. SEO specialists use this data to determine potential advertisement relationships and to study their competition. Online investigators can use this to collect important details that are never visible on the target websites. Three individual services will provide easily digestible data on any domain. I will use my own domain for each example in order to compare the data. Only the free versions will be discussed.

Order 0138471223

Similar Web (similarweb.com)

Similar Web is usually the most comprehensive of the free options. However, some of these details usually contradict other services. Much of this data is "guessed" based on many factors. A search of inteltechniques.com produced the following partial information about the domain.

- The majority of the traffic is from the USA, followed by UK, DE, FR.
- There is no paid search or advertisements on search engines.
- There are 56 websites that possess links to the target, and 15 are visible.
- "Buscador" led more people to the site than any other search term followed by OSINT.
- Over 50,000 people visit the site monthly.
- There are five main online competitors to the target, and the largest is onstrat.com.
- 71% of the visitors navigated directly to the domain without a search engine.
- 12% of the traffic was referrals from other websites and search engines.
- The referrals included my other website (computercrimeinfo.com).
- 3% of the traffic to this site originated from social networks Facebook and Twitter.
- Similar websites include onstrat.com and automatingosint.com.

Moon Search (moonsearch.com)

This service replaced Alexa for me when it was shut down in 2022. It provides a recent screen capture of the target domain, plus its ranking, backlinks, IP address, server technologies, and analytics identifiers. The data will probably be redundant at this point, but the service should be queried. Additional websites which provide a similar service to Moon Search include **Search Metrics** (suite.searchmetrics.com), **SpyFu** (spyfu.com), and **Majestic** (majestic.com).

Shared Count (sharedcount.com)

This website provides one simple yet unique service. It searches your target domain and identifies its popularity on social networks such as Facebook and Twitter. A search of labnol.org produced the following results. This information would lead me to focus on Pinterest and Facebook first. It tells me that several people are talking about the website on these services.

Facebook Likes: 348	Facebook Total: 1034
Facebook Shares: 538	Twitter Tweets: 0
Facebook Comments: 148	Pinterest Pinned: 1

Reddit Domains (reddit.com)

Reddit was discussed previously as a very popular online community. The primary purpose of the service is to share links to online websites, photos, videos, and comments of interest. If your target website has ever been posted on Reddit, you can retrieve a listing of the incidents. This is done through a specific address typed directly into your browser. If your target website was phonelosers.org, you would navigate to the exact URL of reddit.com/domain/phonelosers.org. This example produced 16 Reddit posts mentioning this domain. These could be analyzed to document the discussions and usernames related to these posts.

Small SEO Tools: Backlinks (smallseotools.com/backlink-checker)

After you have determined the popularity of a website on social networks, you may want to identify any websites that have a link to your target domain. This will often identify associates and people with similar interests of the subject of your investigation. There are several online services that offer a check of any "backlinks" to a specific website. Lately, I have had the best success with the backlink checker at Small SEO Tools. A search of my own website, inteltechniques.com, produces 264 websites that have a link to mine. These results include pages within

my own websites that have a link to inteltechniques.com, so this number can be somewhat misleading. Several of the results disclosed websites owned by friends and colleagues that would be of interest if I were your target.

Host.io Backlinks (host.io)

In 2021, I discovered this service which seems to offer many additional backlinks which were not present within the previous option. The following direct URL displays 45 domains which are linking to my website.

https://host.io/backlinks/inteltechniques.com

Host.io Redirects (host.io)

This option displays any URLs which are forwarding their traffic to your target site. The following direct URL displays my own results. You can quickly identify three domains which I own that are forwarding visitors to my main site. Searching for historic records of these domains should reveal outdated websites and details.

https://host.io/redirects/inteltechniques.com

A summary of all details about a domain stored with Host.io can be found via the following direct URL.

https://host.io/inteltechniques.com

Small SEO Tools: Plagiarism Checker (smallseotools.com/plagiarism-checker)

If you have identified a web page of interest, you should make sure the content is original. On more than one occasion, I have been contacted by an investigator that had been notified of a violent threat on a person's blog. I was asked to track down the subject before something bad happened. A quick search of the content identified it as lyrics to a song. One of many options for this type of query is the plagiarism checker at Small SEO Tools. You can use this tool by copying any questionable text from a website and paste it into this free tool. It will analyze the text and display other websites that possess the same words. This service uses Google to identify anything of interest. The benefit of using this tool instead of Google directly is that it will structure several queries based on the supplied content and return variations of the found text. Clicking the results will open the Google search page that found the text. Another option for this type of search is **Copy Scape** (copyscape.com).

Visual Site Mapper (visualsitemapper.com)

When researching a domain, I am always looking for a visual representation to give me an idea of how massive the website is. Conducting a "site" search on Google helps, but you are at the mercy of Google's indexing, which is not always accurate or recent. This service analyzes the domain in real time, looking for linked pages within that domain. It provides an interactive graph that shows whether a domain has a lot of internal links that you may have missed. Highlighting any page will display the internal pages that connect to the selected page. This helps identify pages that are most "linked" within a domain, and may lead a researcher toward those important pages. This visual representation helps me digest the magnitude of a target website.

XML Sitemaps (xml-sitemaps.com)

This service "crawls" a domain and creates an XML text file of all public pages. Scanning my own site and blog displayed direct URLs of 493 unique pages. Exporting the XML file provided documentation of the process. This is a great companion to visual site mappers, as the text can be easily imported into reporting systems. This often presents previously unknown content.

Order 0138471223

Threat Data

This represents a new category for this chapter and I am quite embarrassed it took me so long to realize the value of the content. I use the term "Threat Data" to encompass the top four websites which monitor for malicious content. For a network security analyst, this data might identify potentially malicious sites which should be blacklisted within internal networks. For us OSINT researchers, the data represented here can provide a unique glimpse into our target domain. Instead of explaining every facet of these services, I will only focus on the new evidence received after a query of various domains. All of these services are available in your search tools.

VirusTotal (virustotal.com)

This option displays the most useful information in regard to OSINT, and this is likely the most popular threat data service of the four. Much of the details presented here are redundant to the previous options, so let's focus only on the unique data. The "Details" menu provides the most public data. The Whois and DNS records should be similar to other sites. The "Categories" area provides the general topics of the target site. For mine, it displays "Information Technology". This can be useful to know a small detail about sites which have disappeared. The "HTTPS Certificate" section can become interesting very quickly. The "Subject Alternative Name" portion of this section identifies additional domains and subdomains associated with the SSL certificate of your target site. When I search cnn.com, I receive dozens of additional URLs which could prove to be valuable. Below is a partial view.

dev.client.appletv.cnn.com,dev.cnnmoney.ch,dev.content.cnnmoney.ch,dev.hypatia.api.cnn.io

I now know that cnn.io is directly associated with the target, which should then be investigated. The "Relations" tab identifies many new subdomains such as customad.cnn.com, go.cnn.com, and store.cnn.com. The "Files" section displays unique content from practically any other resource. It identifies files downloaded **from** the target site for analysis and files which have a reference **to** the target site.

Let's analyze my own site as an example. Figure 24.06 displays two files which are present on my site. Both of these files have been analyzed by VirusTotal from either user submission or automated scanning. The first column displays the date of the scan and the second column identifies whether any virus databases detected the files as malicious (none out of 60 tested positive for a virus). The final two columns identify the file type and name. If I were to remove these files today, this evidence would stick around.

Figure 24.07 displays the result in the "Files Referring" section. These are files and programs, which are not present on my site, that refer to my domain. All of these display positive results for being malicious. These are basically files stored on other websites which mention my domain. If you were investigating me, you should try to find these files for further analysis. The fourth file is a virus disguised as a digital copy of this book, attempting to fool would-be downloaders. If your target is mentioned within off-site files, you can learn a lot from analysis. Always use a virtual machine without network access if you plan to download or open anything found here.

2018-11-14	0 / 59	ZIP	ffmpeg.zip
2018-11-25	0 / 60	PDF	workbook.pdf

Figure 24.06: A VirusTotal result identifying files from a domain.

Scanned	Detections	Type	Name
2019-08-29	1 / 48	Win32 EXE	OSCAR.exe
2019-08-04	1 / 60	ZIP	Doxing eBooks.zip
2019-05-14	3 / 73	Win32 EXE	HawkEye.exe
2019-05-09	1 / 60	Office Open XML Document	Open Source Intelligence Techniques
2019-02-11	1 / 60	ZIP	Hacklog2.zip
2019-01-15	2 / 58	PDF	Hacklog. Web Hacking - vol. 2.pdf
2019-01-06	1 / 60	ZIP	Doxing eBooks.zip

Figure 24.07: A VirusTotal result identifying files referring to a domain.

Finally, the "Community" tab can be a treasure of details if anything exists about your target domain. This is where members of the VirusTotal community can leave comments or experiences in reference to the target site. While there are no comments currently on my profile, I have seen helpful details on target "hacking" related sites. These included owner details and new domains created for illegal phishing purposes. Most sites will not have any comments, but this option should be checked while browsing through the other sections.

Threat Intelligence (threatintelligenceplatform.com)

This service replicates many of the features already presented. However, I usually look to three specific sections in the domain report. The "Connected Domains" area identifies any external domains which are linked from your source. This can often display hidden links to third-party services otherwise unknown from previous queries. On my domain, you see a link to the icon service I use because I gave attribution within the footer of my page. In previous investigations, I have found additional domains owned by the suspect. From there, I focus on the "Potentially dangerous content" and "Malware detection" sections. Both of these offer a historical view into any malicious content hosted on the target domain. This can include suspicious files or phishing campaigns. While recently investigating a domain which currently possessed no content, this service confirmed the presence of a phishing page designed to steal credentials.

Threat Crowd (threatcrowd.org)

This service provides a unique view of the domains associated with your target. Figure 24.08 displays a partial result for my own domain. It displays my server IP, primary domain, and additional domains which were once associated with my account. The upper-right domain was a small website created for a friend which was hosted as demonstration for a short amount of time.

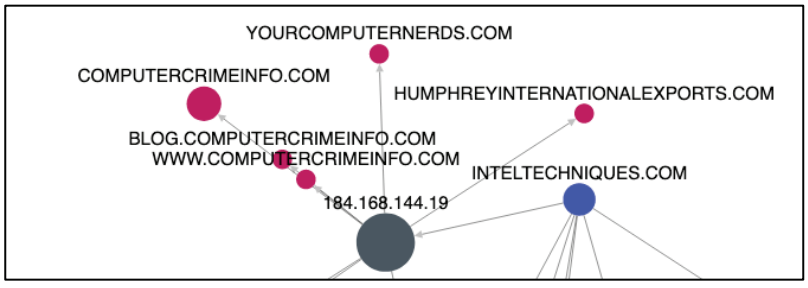


Figure 24.08: A Threat Crowd domain report.

Censys (censys.io)

Censys provides a detailed summary of the basics, which is quite redundant at this point. However, there are three key pieces of information I access here on most domains. I previously mentioned the importance of checking the Subject Alternative Names of a domain's SSL certificate. Most services conduct scans of the entire

Order 0138471223

internet in order to retrieve this data. The moment a certificate is issued, it is provided in real-time to Censys. Censys thus does not need to rely on internet scans to discover certificates, and more importantly Subject Alternative Names. Therefore, I always click the "Details" button on the summary page and look for any interesting data by searching "alt_name" within the results. Next, I have relied on the HTTP Body text information stored within the "Details" page of the HTTP and HTTPS sections. This is basically the HTML code which makes the web page display properly within a browser. It is the same data you would see by viewing the source code of a target page. If the target website should disappear, this entire section of code could be copied; pasted into a text file; saved with an html extension; and opened within a browser to reveal the overall look and structure. I prefer to capture this data in the event of a modified or removed target web page.

Finally, I believe Censys has the overall best data about the security certificates associated with a domain. It provides hyperlinks to every certificate within the chain and extreme details about each. Much of this data is not valuable to an OSINT report, but I prefer to collect a screen capture for potential later analysis or comparison to live data. Overall, threat data is often considered minor bits of information designated only for the digital security community. OSINT practitioners should also be aware of the content available within these sites. While the majority of details are not immediately useful, small nuggets of valuable information which cannot be found anywhere else awaits you.

WordPress Data

Many websites displayed at domains are WordPress blogs. These can be customized in any way to have the appearance of a traditional website. You may want to identify any vulnerabilities which exist within the WordPress installation which may disclose interesting details about the site. The following three providers display the basics without the need to install any software. Please note the terms of service when you use these options. A search of my own domain revealed the version of WordPress; blog IP address; hosting provider; title of the blog; any blacklist entries; installed plugins; custom themes; login usernames; linked websites; and overall security of the site.

https://gf.dev/wordpress-security-scanner
https://hackertarget.com/wordpress-security-scan/

Domain Reputation

In the chapter about email addresses, I explained the importance of investigating an address's reputation for sending spam or delivering malicious content. We can do the same for domains with the following URLs.

https://www.spam.org/search?type=domain&convert_block=1&group_ips=1&data=inteltechniques.com
https://spamdb.org/blacklists?q=inteltechniques.com
https://www.mywot.com/en/scorecard/inteltechniques.com

Data Breaches and Leaks

Similar to email addresses and usernames, domains can possess valuable breach data. You should already be familiar with breach data providers, so I will focus only on methodology here.

Dehashed (dehashed.com)

This service and search options function identical to the email queries. Provide the domain, and the results display the breaches which possess at least one email address matching the provided data. The URL query structure is https://dehashed.com/search?query="inteltechniques.com".

IntelX (intelx.io)

This service presents partial Pastebin files which include your target domain. A direct query URL is https://intelx.io/?s=inteltechniques.com. A free trial is required to see all results.

Leakpeek (leakpeek.com)

This service requires a free account to search domains, and a direct URL is not available.

Phonebook (phonebook.cz)

This option searches a domain for any email addresses which exist within publicly available breaches. A search of my own domain found four active addresses.

Shortened URLs

Social networking sites, such as Twitter, have made the popularity of shortened URL services soar. When people post a link to something they want their friends to see, they do not want the link to take up unnecessary space. These services create a new URL, and simply point anyone to the original source when clicked. As an example, I converted a URL to a blog post from "https://inteltechniques.com/blog/2019/08/03/book-release-extreme-privacy/" to "https://bit.ly/32Up8h7".

You have likely seen these during your own investigations, and many people pay them little attention. There is actually a lot of information behind the scenes of these links that can reveal valuable information associated with your investigation. For a demonstration, I created the following shortened links, all of which forward to my home page. After, I will explain how to access the hidden data behind each service.

bitly.com/29A4U1U
http://tiny.cc/v973ez
goo.gl/Ew9rlh
bit.do/cbvNx

Bitly allows access to metadata by including a "+" after the URL. In our scenario, the direct URL would be bitly.com/29A4U1U+. In this example, the results only identified that 21 people have clicked on my link. However, creating a free account reveals much more detail. After logging in, I can see any websites that referred the user to the link and extremely generic location data, such as the country of the user. This is a good start.

Tiny.cc adds a "~" to the end of a link to display metadata. In our example, the direct URL would be tiny.cc/v973ez~. The results on this page identify the number of times the URL was clicked, the number of unique visits, the operating systems of those that clicked the link, and the browsers used. This service also displays generic location data, such as the country of the user.

Google gives us the same detail as above. It also uses the "+" at the end, and our direct demo URL would be goo.gl/Ew9rlh+. This demo notified me that 18 people have clicked my link from 7 different countries. They are mostly Windows users with the Chrome browser.

Bit.do provides the most extensive data. They use a "-" after the URL, and our direct demo address would be bit.do/cbvNx-. The results identify all of the details listed previously, plus the actual IP addresses of each visit. This type of service can be used in many ways. If you are investigating a viral Twitter post with a shortened URL, you may be able to learn more about the popularity and viewers. You could also use this offensively. During covert investigations, you could forward a shortened URL from Bit.do and possibly obtain the IP address being used by the suspect. If you are investigating a shortened URL link that was not mentioned, consider using the catch-all service at **CheckShortURL** (checkshorturl.com).

Order 0138471223

UNFURL (dfir.blog/unfurl/)

This service appeared in late 2022, and I find it very valuable. You can enter any full internet address to gather behind-the-scenes details by expanding the data associated within the URL. This could be to identify details behind a Twitter post, Instagram photo, or shortened URL. As one of many examples, I often find Twitter links posted in various places. Clicking these will forward me to the intended URL target, but I want more details. The link of https://t.co/QPs812NVAW contains the data presented within Figure 24.09. I can see all forwarding details, timestamps, and unique identifiers without visiting the site.

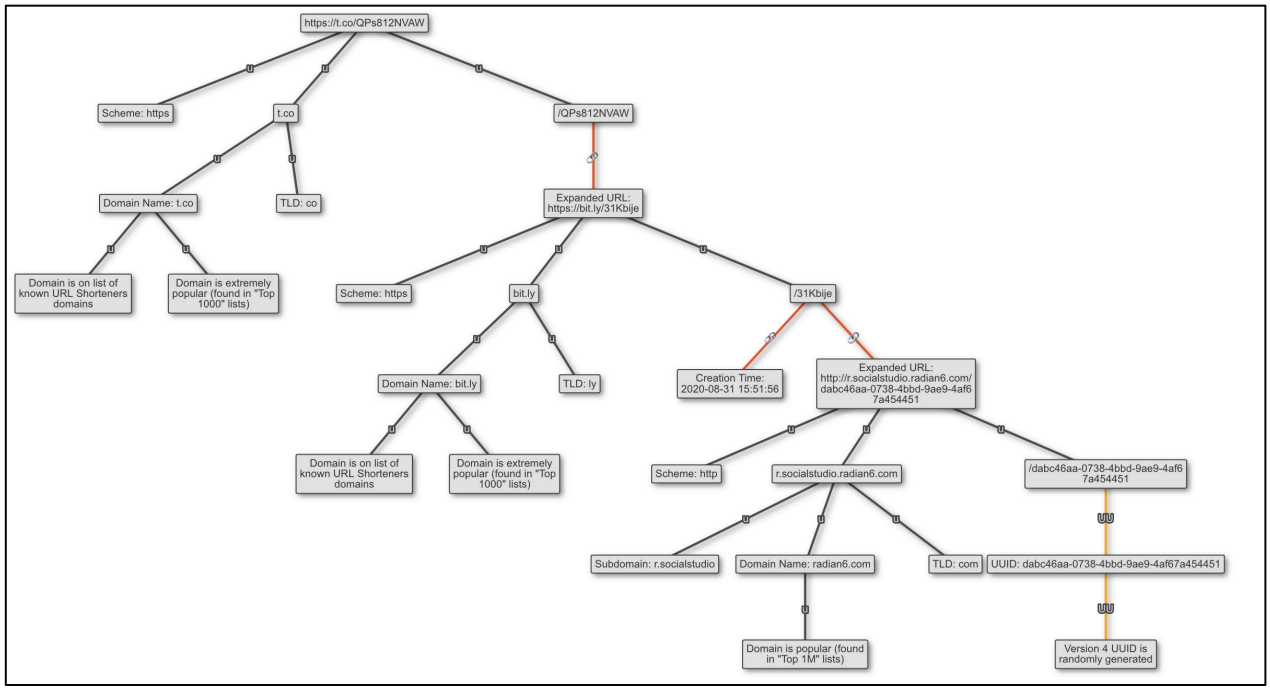


Figure 24.09: A UNFURL result.

I also find this very beneficial for the investigation of magnet Torrent links. Any time I find a Torrent link, I copy and paste the entire URL into this tool. It typically displays the web seed, address tracker, source, file details, and domains. I encourage you to submit any URLs or domains of interest to this tool. You will soon discover the possibilities as related to your types of investigations.

Cloudflare

While identifying web hosts and IP addresses behind your target domain, you are likely to encounter sites hiding behind Cloudflare. This company provides security for websites which often prevents online attacks and outages. They also help keep web host and owner details hidden when investigating criminal websites. There is no magic solution to uncover the owner of a website behind Cloudflare's protection, but we do have a few investigative options, as follows.

Historical: Use the previous methods to search historical domain registration records. While Whois sites may show Cloudflare today, historical registration records may show the actual web host used prior to Cloudflare. The result may or may not be the current host.

Censys (censys.io): Most websites possess Secure Socket Layer (SSL) certificates. Searching a domain name through the Censys "Certificates" search may identify historical SSL ownership records. The direct URL of "https://censys.io/certificates?q=inteltechniques.com" displays my own domain. The results identify my SSL host authority as Sectigo in 2020, Comodo (Namecheap) in 2018, and GoDaddy in 2016. You now know my

domain host history. This area of Censys can be beneficial to any domain regardless of Cloudflare protection, which is a feature often overlooked. We can also query a specific SSL certificate to potentially see other associated domains. The previous search identified "7583b0cb25632de96575dd0f00ff99fed81b9069" as the SSL certificate which I possessed in 2016. Searching this within Censys under the "Certificate" menu provides the domains of computercrimeinfo.com, inteltechniques.com, privacy-training.com. We now have other domains to investigate. Some of them may reveal the current domain registration provider and web host.

Third-Party Tracking: Some websites will hide their domain registration and host behind various protection services, but continue to use analytics, tracking, and Google services. Consider the previous methods of searching Analytics, and any other unique identifiers across other sites owned by the same target.

Advanced DNS

I mentioned **Domains App** (dmns.app) previously as a way to query stored historical screen captures. We can also use this resource to see much more DNS details than the services explained at the beginning of this chapter. This can often include an email address within the DMARC data which is not visible elsewhere. Consider an example for michaelbazzell.com. We can query with the following direct URL.

https://dmns.app/domains/michaelbazzell.com/dns-records

The results identify the usual DNS suspects, including my web host and server IP address. However, there is new data in the last sections.

"protonmail-verification=f84a8f78b21c92a4493fe5d9d5cb1150385846e9"
"v=spf1 include:_spf.protonmail.ch mx ~all"
"v=DMARC1; p=none; rua=mailto:donotspamme@michaelbazzell.com"

We now know that all email for that domain is handled within a Proton Mail account. We also see a new email address in the final DMARC section. For many domains which apply this extra level of email security and verification, you will find a legitimate email address which may have escaped your other analysis. As I write this, I learned of a personal email address included within results for a domain which is part of a current investigation. I have included this option within the search tool explained next.

IntelTechniques Domain Tool

Similar to the previous custom search tools mentioned here, I have created a page for easier domain searching. While it does not possess every service discussed in this chapter, it can automate queries across the most beneficial options. Each box allows entry of a target domain name. The "Submit All" option will open several tabs within your browser that present each query listed on the page. The final section provides easy access to shortened URL metadata. The online version at **https://inteltechniques.com/tools/Domain.html** is updated often.

Order 0138471223

CHAPTER TWENTY-FIVE
IP ADDRESSES

IP addresses are often obtained from an internet investigation, email message, or connection over the internet. When legal process is served to online content providers, a list of IP addresses used to log in to the account is usually presented as part of the return of information. Serving legal orders to identify and obtain IP addresses is outside the scope of this book. However, several techniques for collecting a target's IP address using OSINT are explained in this chapter. The previous instruction assumed that you were researching a domain name. These names, associated with websites, simply forward you to a numerical address that actually hosts the content. This is referred to as an Internet Protocol (IP) address.

The way that you encounter IP addresses as a target of your research will vary widely. Law enforcement may receive an IP address of an offender after submitting a subpoena to an internet provider. Any online researcher may locate an IP address while researching a domain with the previous methods. While only one website can be on a domain, multiple domains can be hosted on one IP address. The following resources represent only a fraction of available utilities. Note that many of the domain name resources mentioned in the previous chapters also allow for query by an IP address. Let's begin with the basic resources at ViewDNS.

ViewDNS Reverse IP (viewdns.info/reverseip)

This page was previously used to translate a domain name into an IP address. It will also display additional domains hosted on an individual IP address. This service identified 134 domains hosted on 104.28.10.123. These included domains from websites all over the world without a common theme. This indicates that he uses a shared server, which is very common. If I would have seen only a few domains on the server, that may indicate that he is also associated with those specific domains.

ViewDNS IP Location (viewdns.info/iplocation)

This utility cross-references an IP address with publicly available location data connected to the server hosting any domains associated with the IP address. A search of 54.208.51.71 revealed the following information.

City: Ashburn
Zip Code: 20147
Region Name: Virginia
Country Code: US
Country Name: United States

ViewDNS Port Scan (viewdns.info/portscan)

This online port scanner looks for common ports that may be open. An open port indicates that a service is running on the web server that may allow public connection. A search of 54.208.51.71 revealed that ports 21, 53, 80, and 443 are open to outside connections. Port 21 is for FTP connections, 53 is for DNS settings, 80 is for web pages, and port 443 is for secure web pages.

ViewDNS IP Whois (viewdns.info/whois)

This service was used earlier to display registration information about an individual domain. Entering an IP address will attempt to identify details about any domain registrations associated with the address. A search of 54.208.51.71 revealed it to belong to Amazon and provided the public registration details.

ViewDNS IP Traceroute (viewdns.info/traceroute)

This tool identifies the path that ViewDNS took from their servers to the target IP address. This can identify IP addresses of servers that were contacted while you tried to establish communication with the address. These will occasionally identify associated networks, routers, and servers. Additional IP addresses can be searched for further details. The numbers after the IP addresses indicate the number of milliseconds that each "hop" took.

ViewDNS Reverse DNS (viewdns.info/reversedns)

This simply finds the reverse DNS entry for a given IP, which is usually the host name.

Similar to the Domain chapter, ViewDNS allows query of IP addresses via URL, as follows.

https://viewdns.info/reverseip/?host=70.39.110.82&t=1
https://viewdns.info/iplocation/?ip=70.39.110.82
https://viewdns.info/portscan/?host=70.39.110.82
https://viewdns.info/whois/?domain=70.39.110.82
https://viewdns.info/traceroute/?domain=70.39.110.82
https://viewdns.info/reversedns/?ip=70.39.110.82

For redundancy, I also recommend the following UltraTools direct IP address query URLs.

https://www.ultratools.com/tools/ipWhoisLookupResult?ipAddress=70.39.110.82
https://www.ultratools.com/tools/geoIpResult?ipAddress=70.39.110.82
https://www.ultratools.com/tools/pingResult?hostName=70.39.110.82

Bing IP (bing.com)

Once you have identified an IP address of your target, you can search for websites hosted on that IP address. A specific search on Bing will present any other websites on that server. If your target is stored with a large host, such as GoDaddy, there will not be much intelligence provided. It will only list websites that share a server, but are not necessarily associated with each other. If the user is hosting the website on an individual web server, this search will display all other websites that the user hosts. This search only works on Bing and must have "ip:" before the IP address. An example of a proper search on Bing would look like ip:54.208.51.71. The results of this search identify every website hosted by a specific local website design company. The direct URL follows.

https://www.bing.com/search?q=ip%3A54.208.51.71

IPLocation (iplocation.net)

IPLocation offers unlimited free IP address searches, and queries five unique services within the same search results. The results are the most comprehensive I have seen for a free website. While GPS coordinates of an IP address are available, this most often returns to the provider of the internet service. This usually does not identify the exact location where the IP address is being used. The country, region, and city information should be accurate. If an organization name is presented in the results, this indicates that the address returns to the identified company. The exception here is when an internet service provider is identified. This only indicates that the IP address belongs to the specified provider. Most results translate an IP address into information including business name, general location, and internet service provider. This can be used to determine if the IP address that a target is using belongs to a business providing free wireless internet. If you see "Starbucks", "Barnes & Noble", or other popular internet cafés listed in the results, this can be important intelligence about the target. This can also confirm if an IP address is associated with a VPN service. The direct URL query follows.

https://www.iplocation.net/ip-lookup?query=70.39.110.83&submit=IP+Lookup

That's Them (thatsthem.com/reverse-ip-lookup)

The previous resources rely on conventional IP address data, which is sourced from various registration documentation and scanning of servers. Very little information is sensitive or personal in nature. That's Them enters into an environment that is a lot more invasive. This service, mentioned previously during person, email, and telephone search, collects marketing data from many sources to populate its database. This often includes IP address information. These details could have been obtained during an online purchase or website registration. Regardless of the source, the results can be quite beneficial. At the time of this writing, I searched an IP address associated with a business email that I received. The result identified a person's name, home address, company, email address, and age range. All appeared accurate. This tool will work best when searching static business IP addresses, and not traditional home IP addresses that can change often. While I get no results much more often than positive results, this resource should be in everyone's arsenal. The direct URL query follows.

https://thatsthem.com/ip/70.39.110.82

I Know What You Download (iknowwhatyoudownload.com)

While discussing invasive websites, this resource might be the most personal of all. This service monitors online torrents (ways to download large files which often violate copyright laws) and discloses the files associated with any collected IP addresses. I searched the previous IP address collected from an associate, and received an immediate hit. Figure 25.01 displays the result. It identifies that the target IP address was downloading two specific movies on December 28, 2017 at 9:53 pm. Clicking on the movie title presents every IP address captured that also downloaded the same file. Again, this will work best with IP addresses that rarely change, such as a business, organization, or public Wi-Fi network. I have used this to determine the files being downloaded from the network with which I was currently connected. On one occasion, this revealed an employee that was downloading enormous amounts of pornography on his employer's network. He should have used a VPN, which would have masked his online activity from me. In order to see the power of this type of service, try searching a known VPN address such as an address provided by Private Internet Access (PIA) 173.244.48.163. While I know that no one reading this book has ever downloaded pirated content, this should serve as a reminder why VPNs are essential. The direct URL query follows.

https://iknowwhatyoudownload.com/en/peer/?ip=70.39.110.82

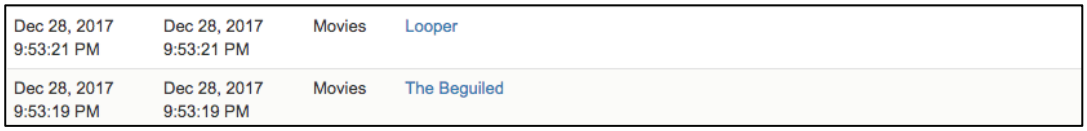


Figure 25.01: A search result from I Know What You Download.

Exonerator (metrics.torproject.org/exonerator.html)

The Onion Router (Tor) was explained in Chapter Three. It is a network that provides anonymity by issuing IP addresses to users that connect to servers in other countries. If you possess an IP address of your target, but cannot locate any valuable information using the previous techniques, it is possible that the address was part of the Tor network and there is no relevant data to be located. Exonerator is a tool that will verify the usage of an IP address on the Tor network. Provide the IP address and a date of usage, and the service will display whether it was used as a Tor connection. While a date is required, you could provide the current date if your target time frame is unknown. Most IP addresses are typically always or never a part of the Tor network.

Wigle (wigle.net)

Wigle is a crowd-sourced database of wireless access points. Users in all areas of the country conduct scans of wireless devices in their area; identify details of each device; and submit this data to Wigle in order to map the found devices on the site. This allows anyone to browse an area for wireless access points or search an address to locate specific devices. Additionally, you can search for either a specific router name or MAC address and locate any matching devices. The results include links that will display the results on an interactive map. Most of the world has been covered. In order to take advantage of the search features, you will need to register for a free account. Generic or misleading information can be used that does not identify you.

There are many investigative uses for this service. You can identify the wireless access points in the immediate area of a target's home. As an example, a search of the address of a gas station revealed a map of it with the associated routers. In this view, I can identify the router names including potential sensitive information. It displays wireless router SSID's of AltonBPStore, tankers_network, Big Toe, and others. Clicking View and then Search in the upper left of the page presents a detailed query engine. A search of tankers_network, as identified previously in the map view, displays details of the wireless access point. It has a MAC address of 00:1F:C6:FC:1B:3F, WPA encryption, was first seen in 2011, and operates on channel 11.

An investigator could also search by the target's name. This may identify routers that have the target's name within the SSID. A search of "Bazzell" identifies seven access points that probably belong to relatives with my last name. These results identify the router name, MAC address, dates, encryption method, channel, and location of the device. This can easily lead an investigator to the home of a target.

Many internet users will use the same name for their wireless router as they use for their online screen name. Assume that your target's username was "Hacker21224". A search on Wigle for "Hacker21224" as a router name might produce applicable results. These could identify the router's MAC address, encryption type, and GPS coordinates. A search on Google Maps of the supplied GPS coordinates will immediately identify the home address, a satellite view of the neighborhood, and a street view of the house of the target. All of this intelligence can be obtained from a simple username. These results would not appear on any standard search engines.

If you find Wigle valuable to your investigations, I recommend you create a free account. While logged in, you can use their advanced search (wigle.net/search) or submit direct queries via URL as follows.

https://wigle.net/search?ssid=bazzell
https://wigle.net/search#fullSearch?postalCode=62002

Shodan (shodan.io)

Shodan is a search engine that lets you find specific computers (routers, servers, etc.) using a variety of filters. General search engines, such as Google and Bing, are great for finding websites; however, they do not search for computers or devices. Shodan indexes "banners", which are metadata that a device sends back to a client. This can be information about the server software, what options the service supports, or a welcome message. Devices that are commonly identified through Shodan include servers, routers, online storage devices, surveillance cameras, webcams, and VOIP systems. Network security professionals use this site to identify vulnerabilities on their systems. Criminals use it to illegally access networks and alter devices. We will use it to locate specific systems near a target location. In order to take advantage of Shodan's full search capabilities, you must create a free account. Only a name and email address is required. The following example will identify how to locate live public surveillance cameras based on location. The target for this search is Mount Pleasant, Utah. The following search on Shodan produced 9,684 results.

country:US city:"Mount Pleasant"

There are two flaws with this search. First, you may receive results from other cities named Mount Pleasant. Second, you will likely receive too many results to analyze effectively. A search of "geo:39.55,-111.45" will focus only on the specific GPS location of interest (Lat=39.55, Long= -111.45). There were 238 results for this search. This is much more manageable and all of the results will be devices in the target area. Adding more specific search criteria will filter the results further. A search of "geo:39.55,-111.45 netcam" identified only one device. The result displays this device as a "Netcam". It also identifies the internet service provider as "Central Utah Telephone" indicating the user has a DSL connection.

To connect to the device, you would click on the IP address identified as 63.78.117.229. Clicking through each of these options may be time consuming. You can add a search term to filter your results. Replicating this search for a GPS location in a large city will produce many results. Clicking the IP address will take you to the page that will connect to each device. You must be careful here. Some devices will require a username and password for access. You could try "admin" / "admin" or "guest" / "guest", but you may be breaking the law. This could be considered computer intrusion. However, many of the webcam and netcam results will not prompt you for a password and connect you to the device automatically. There is likely no law violation when connecting to a device that does not prompt you for credentials. Your local laws may prohibit this activity.

Shodan Maps (maps.shodan.io) allows you to conduct any of these searches based on location alone while **Shodan Images** (images.shodan.io) displays collected webcam captures from open devices. Figure 25.02 displays a home using an automated lighting and climate control system in Missouri located with Shodan Maps. These two options are premium services and require a modest fee. All Shodan features allow input of the following types of information for filtering.

City: Name of the city (ex. City:"San Diego")
Country: 2-letter country code (ex. Country:US)
GPS: Latitude and longitude (ex. Geo:50.23,20.06)
OS: Operating system (ex. Os:Linux)
IP Address: Range (ex. Net:18.7.7.0/24)
Keyword: (ex. Webcam)

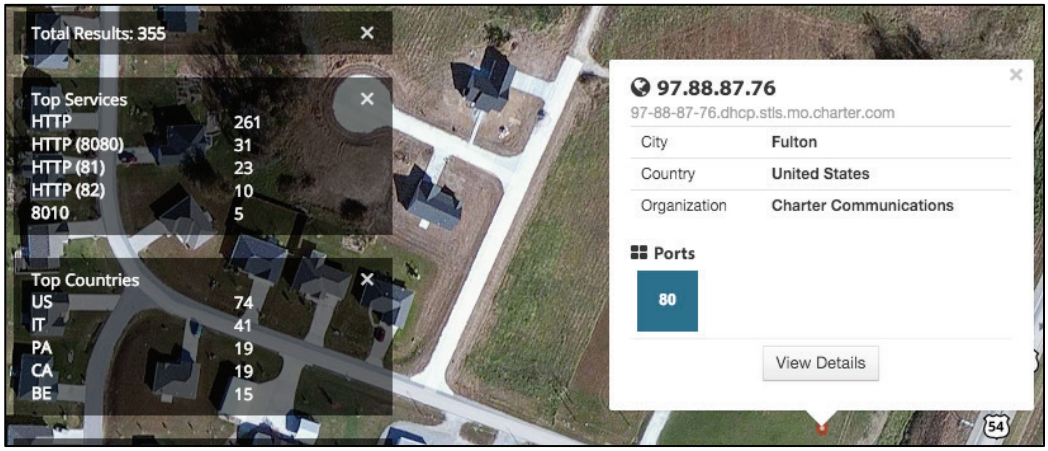


Figure 25.02: A Shodan Maps search result.

Shodan Beta (beta.shodan.io) offers complete details of a specified IP address. This can be queried with the following three URLs.

https://beta.shodan.io/host/70.39.81.131
https://beta.shodan.io/host/70.39.81.131/raw
https://beta.shodan.io/host/70.39.81.131/history

The first presents the standard view while the second offers text-only details which may be more valuable to a report. The final option looks at all available details about the domain throughout several scans over time. Each entry displays a date in the format of 2020-04-26T07:49:17 (April 26, 2020 at 7:49 am).

Zoom Eye (zoomeye.org)

This Shodan competitor provides a similar service, often with unique results. A direct URL query follows.

https://www.zoomeye.org/searchResult?q=70.39.110.82

Threat Crowd (threatcrowd.org)

As mentioned in the previous chapter, Threat Crowd is a system for finding and researching artifacts relating to cyber threats. Searching an IP address can reveal an association to malicious software being spread over the internet. A positive result will display the type of malware, associated domain names, dates of discovery, and any comments by other researchers. Most readers that actually need this type of service likely already know more about it than me. However, it should be a consideration when investigating suspicious IP addresses. A URL query follows.

https://www.threatcrowd.org/ip.php?ip=70.39.110.82

Censys (censys.io)

Similarly, Censys is a search engine that enables researchers to ask questions about the hosts and networks that comprise the internet. Censys collects data on hosts and websites through daily scans of the internet, in turn maintaining a database of how hosts and websites are configured. Researchers can interact with this data through a search interface. As an example, a search of 173.189.238.211 reveals it to be associated with a Schneider Electric BMX P34 2020 device through a Windstream provided internet connection, located near Kansas City, Kansas. A URL query follows.

https://censys.io/ipv4/70.39.110.82

Ipv6 Addresses

The IP addresses previously mentioned were all version four (Ipv4), such as 192.168.1.1. Due to limited availability, many providers are switching to Ipv6, which allows many more addresses. A typical example may appear as 2001:0db8:85a3:0000:0000:8a2e:0370:7334. While many of the utilities mentioned here are adapting to this input, we should query these types of addresses through designated Ipv6 engines. I have tested dozens, and I find the following two to work best, both of which are included in the search tool.

https://www.ultratools.com/tools/ipv6InfoResult?ipAddress=2001:db8::8a2e:370:7334
https://www.ultratools.com/tools/ping6?ipAddress=2001:db8::8a2e:370:7334

Email Headers

I no longer teach email header analysis in my live courses. The vast majority of users rely on web-based email such as Gmail or Yahoo. These services do not disclose the IP address of an individual user within the email headers. The only email headers that I have encountered over the past three years that contained valuable IP addresses were business users that sent emails within a desktop client such as Outlook. If you would like to analyze an email header in order to identify the IP address and sender information, you have two options. You can look through a few sites and teach yourself how to read this confusing data, or you can use an automated service.

IP2Location (ip2location.com/free/email-tracer) provides a large text box into which an entire email header can be copied for analysis. The response includes the IP address and location of the sender; interactive map identifying the originating location; internet service provider; and links to additional information from an IP search. Anyone wanting more information from an email threat should start here. An alternative site that conducts similar actions is **MX Toolbox** (mxtoolbox.com/EmailHeaders.aspx).

Obtaining a Target's IP Address

You may want to know the IP address of the person you are researching as provided by their internet service provider. This address could be used to verify an approximate location of the person; to provide law enforcement details that would be needed for a court order; or to determine if multiple email addresses belong to the same subject. All of those scenarios will be explained here while I explain the various services that can be used.

IP Logger (iplogger.org)

For many years, this was my favorite option for identifying the IP address of a target. There are many options now, and most of them will be explained here. This specific technique involves some trickery and the need to contact the target from a covert account. For this demonstration, assume your target has a Facebook page that he checks regularly. You can send him a private message that includes "bait" in the form of an online link. A detailed set of instructions should explain the processes. The main website presents several options, but only the "URL & Image Shortener" service will be explained.

Link: You can generate a URL which will redirect to any website that you provide. IP Logger will save the IP address of each user who clicked the link. In the box provided, enter any address that you want the target to see when clicking on a link. This could be something generic such as cnn.com. After submitting, you will receive a series of links. This page also serves as the log of visitors, and I recommend documenting it. In an example, I received the following link at the beginning of this list.

http://www.iplogger.org/3ySz.jpg

Although the link appears to be a jpg image, clicking this link or typing it into a browser forwards the target to cnn.com. This action collects his or her IP address, operating system, and browser details. These details, along with the date and time of capture, can be viewed at the link generated previously. A URL shortening service such as Bitly (bit.ly) would make the link look less suspicious.

Image: You can provide a digital image to this service, and it will create a tracker out of it for placement onto a website, forum, or email message. I provided an image that is present on my website at inteltechniques.com/img/bh2016.png. This presented a page similar to the previous example. I was provided the following links.

http://www.iplogger.org/23fq.jpg <img src="http://www.iplogger.org/23fq.jpg">

The first link forwards to the image that I provided. During this process, the IP address, operating system, and browser details are collected and stored on the page that stored the links. The second link could be inserted directly into a web page or email message. Upon loading either, the image is presented and collects the data.

Real World Application: I was once communicating with an unknown subject on a web forum about hacking and stolen credit card numbers. I wanted to find out his IP address in order to discover his true identity with a court order. I told the hacker I had an image of a freshly stolen debit card that I was willing to share. He requested proof, so I created an IP Logger link based on a generic online image, and embedded that link into the web forum where we were communicating. Within a few moments, I visited the log for this image and discovered his IP address in Newark, New Jersey.

Canary Tokens (canarytokens.org)

A newer option for IP identification is Canary Tokens. It offers redundant functionality as the previously mentioned product, but may be more useful to you. Ultimately, you should familiarize yourself with all options and choose which works best for you. Lately, I have found Canary Tokens to be the superior option of all. It allows creation of a PDF or DOCX file that contains a tracker, and is the most user-friendly of the services. After choosing a tracking option, it walks you through the process. I maintain a few Canary Token files at the following address. They are used as traps for people that conduct Google searches attempting to find my home address. Opening any of these alerts me to your IP address and general location. At the time of this writing, the most recent opening of one of these documents occurred only two days prior. The culprit lives in Matawan, New Jersey, possesses MCI as an internet provider, and had recently downloaded an Xbox 360 game through a torrent.

https://inteltechniques.com/canary

Always remember that technologies such as VPNs, Tor, and other forms of IP masking may create inaccurate results. Always use caution when sending these types of trackers, and make sure you are not violating any laws or internal policies. Due to the heavy usage of VPNs within the communities in which I investigate, I find these services slowly becoming less useful.

LinkBait (github.com/AmIJesse/LinkBait)

There is a glaring problem with all of these public IP logging services. They are well-known and may be blocked by email providers. Gmail typically blocks any domains associated with either IP Logger or Canary Tokens. A tech-savvy target may recognize these tactics which could jeopardize your investigation. This is why I rely on my own self-hosted option made by my colleague Jesse. The link above includes all files required to host your own IP Logger on your website. You simply need to copy the PHP file into a web-accessible directory on your Apache web server, including shared web hosts. I renamed the file to "index.php" and placed it on my website at https://inteltechniques.com/site/index.php. If you visited that page by clicking a link I had sent to you, a text file would have been generated in that folder. The title would have included the date and your IP address. The following represents a partial view of the content submitted by your computer. This page is not currently live because I do not want to log visitors to my site.

```
70.39.126.131
Mozilla/5.0 (Macintosh; Intel Mac OS X 10.14; rv:81.0) Gecko/20100101
Firefox/81.0
touch: false / mic: Found / gpu: AMD Radeon Pro 560X OpenGL Engine
Screen Height: 1080
Language: en-US
Fonts: 109 fonts: American Typewriter...
Detected OS   = Mac OS X  [generic] [fuzzy]
MTU           = 1392
Network link  = OpenVPN UDP bs128 SHA1 lzo
MTU not 1500, VPN probable.
"country":"United States"
"regionName":"California","city":"Los Angeles","isp":"Sharktech"
"timezone":"Denver"
browserVersion: 5.0 (Macintosh) / Screen Width: 1920
System Time: 2020-10-14 16:2:6
router: https://192.168.1.1
browser: Mozilla / platform: MacIntel
Discord: Not Running
Logins: Google Services
```

I would then know that you clicked the link from a Mac computer through Firefox. I would have details about your hardware and know all of the fonts installed to the operating system. I would know that you are connected to PIA VPN from a west coast server, but your system time is set to Mountain. I would also know you are logged in to a Google product. If you would like to see what details your computer would submit to this script, I have a test page available at the following URL.

https://inteltechniques.com/logger/

This page does not collect your details or store them on my server and nothing is logged. It simply generates the same queries through JavaScript through your own browser and displays them within your screen. Theoretically, any website could replicate these tactics and collect data about your visit. This is why investigators should always use caution and practice good operational security. I dive deep into this topic in my book ***Extreme Privacy***.

GetNotify (getnotify.com)

GetNotify tracks the opening of email messages and presents the connection information of the target. This service is completely free and does not require Gmail as your email provider. You will need to create an account through the Get Notify website and you will be limited to five email messages per day. After you have registered the email address you will be using, you can send emails from that account as usual. However, you will need to add ".getnotify.com" after each email recipient. Instead of sending an email message to the valid account of Michael@inteltechniques.com, you would send the message to a modified email address of Michael@inteltechniques.com.getnotify.com. This will force the email message to go through Get Notify's servers and route the message to the valid address. When your target reads the email message, Get Notify will track the user's IP address, geographical location, and notify you whether your message was viewed for a length of time or deleted right away.

Get Notify works by adding a small invisible tracking image in your outgoing emails. When your email recipient opens your message, this image gets downloaded from a Get Notify server. Get Notify will know exactly when your sent email was opened and it notifies you through an email that your sent message was read by the recipient. You can also view log files within your online account. The tracking image inserted by Get Notify is invisible to the recipient. Optionally, you can specify your own images to be used as tracking images by going to the preferences section after signing in to GetNotify.com. Your recipient will not see ".getnotify.com" at the end of his or her email address. If you want to send a single email to multiple recipients, you should add ".getnotify.com" at the end of every email address.

There are countless scenarios that may make these techniques beneficial to your online research. While I used it for law enforcement, especially in tracking down stolen goods on Craigslist, civilians can use it for many different things. Private investigators have used it on dating websites while hunting cheating spouses. Singles have used it to verify that the potential mate they have been chatting with is local and not in another state or country. The possibilities are endless.

IntelTechniques IP Addresses Tool

Similar to the domain tool mentioned previously, this page automates some of the most common IP address searches. The first box accepts any IP address. Clicking the Populate All button will insert this address into all the search options where manual queries can be conducted. The final option will open several tabs within your browser. The online version at **https://inteltechniques.com/tools/IP.html** is updated often.

CHAPTER TWENTY-SIX
GOVERNMENT & BUSINESS RECORDS

Open source government and business information has never been easier to obtain. A combination of a more transparent government, cheaper digital storage costs, and marketing data leaks has placed more information online than ever before. There is no standard method of searching this data. One county may handle the queries much differently than another county, and business records vary by state. The following resources and techniques should get you started in the United States.

County General Records (www.blackbookonline.info/USA-Counties.aspx)

Counties all over America have digitized the majority of their public records and allow unlimited access over the internet. Searching for your county's website will likely present many information options. This can become overwhelming and it can be easy to get lost within the pages of the site. My initial preference when just beginning an investigation is to use Black Book Online's free county public records page. It allows you to drill down from state to county. The resulting page isolates all available records for viewing. As an example, I chose Illinois and then Madison County as my target. I was presented with the following databases, each linking directly to the source.

Coroner Reports
Delinquent Tax Sale
Government Expenditures
Property Tax Search
Public Employee Salaries
Recorded Documents
Registered Lobbyists
Press Releases
Voter Registration Verification
Voter Registration Addresses
Unclaimed Property
Crime Map
Building Contractors
Building Permits
Foreclosed Properties

County Court Records (www.blackbookonline.info/USA-County-Court-Records.aspx)

A Google search of your county of interest should identify whether an online court records database is available. As an example, St. Clair County in Illinois possesses a website that has their entire civil and criminal court records online (co.st-clair.il.us/departments/circuit-clerk/courts). Searching only a last name will present profiles with full name, date of birth, physical identifiers, case history, fines, pending appearances, and more. Navigating the website will expose charged crimes even if they were dismissed. This can be extremely useful in civil litigation. There are several websites that help connect you to publicly available county government records, such as Black Book Online. It allows you to drill down to your local records. The main page will prompt for the state desired. The result will be a list of links that access each county's court information. Some rural areas are not online, but an occasional search should be done to see if they have been added. Repeating my previous search of Madison County, Illinois revealed the following court related databases.

Circuit Court Complete Docket
Circuit Court Attorney Docket
Family and Civil Pro Se Dockets
Felony State's Attorney Jury Trials
Traffic, Misdemeanor, DUI Docket
Traffic Citations
Crash Reports
Police Blotter
Daily Crime Log
Jail Inmate Search

If the Black Book Online options do not provide optimal results, please consider **Public Records Online** (publicrecords.onlinesearches.com) or a new robust online service which appeared in 2021 called **Netronline** (publicrecords.netronline.com). This option seems to receive more updates of both content and design than Black Book Online. Ultimately, all of these simply connect you to the source of the records. Use the option which works best in your area.

Order 0138471223

PACER (pacer.gov)

PACER is an acronym for Public Access to Court Electronic Records. It is an electronic public access service of United States federal court documents. It allows users to obtain case and docket information from the United States district courts, United States courts of appeals, and United States bankruptcy courts. It holds more than 800 million documents. PACER charges $0.10 per page. The cost to access a single document is capped at $3.00, the equivalent of 30 pages. The cap does not apply to name searches, reports that are not case-specific, and transcripts of federal court proceedings. Account creation is free and if your usage does not exceed $15 in a quarter, the fees are waived. I have possessed an account for several years and have never been billed for my minimal usage. PACER has been criticized for being hard to use and for demanding fees for records which are in the public domain. In reaction, non-profit projects have begun to make such documents available online for free.

RECAP (courtlistener.com/recap)

RECAP (PACER backwards) allows users to automatically search for free copies during a search in PACER, and to help build up a free alternative database at the Internet Archive. It is an extension for the Firefox and Chrome browsers. Each PACER document is first checked if it has already been uploaded by another user to the Internet Archive. If no free version exists and the user purchases the document from PACER, it will automatically upload a copy to the Internet Archive's PACER database. While the browser extension assists greatly with searching, a search page exists on RECAP at the address above.

UniCourt (unicourt.com)

This court document search site appeared in 2018 and currently possesses an impressive database. The search options are straightforward, and results appear quickly, but you are limited to three searches without a paid membership. In my experience, clearing your browser's cache and selecting a new VPN IP address seems to reset this restriction. This is a premium site with full court documents hidden from guests, but we can obtain a fair amount of free information. Let's conduct an actual example.

Searching "Facebook" within the official site provides numerous results. However, after clicking through three case summaries, you will likely receive a notification to purchase a monthly premium membership. This may be acceptable if you plan to use the service heavily, but there are many additional free resources available. Consider conducting your queries through Google instead. Since the case summaries are publicly available, Google appears to have indexed most, if not all, of the pages. The following Google search produced over 10,000 results.

site:unicourt.com "facebook"

The first result connected to the following static URL.

https://unicourt.com/case/ca-la23-adam-blumenkranz-vs-facebook-123515

This page returned the following case details without a subscription or registration.

Case Summary	Case Type	State
Case Number	Jurisdiction	Defendant Names
Filing Date	Judge Name	Plaintiff Names
Update Date	Courthouse	Respondent Names
Case Status	County	Docket Entries

The documents area is restricted, but the docket entries are public. In this case, the details provide dates and actions taken within the litigation. Below is a partial example. UniCourt is now a staple within my investigations, but I rely on Google to find a direct link to the data.

03/14/2018 Case Management Statement; Filed by FACEBOOK, INC. (Defendant)
03/14/2018 Reply Filed by FACEBOOK, INC.; MARK ZUCKERBERG (Defendant)
02/21/2018 NOTICE OF CONTINUANCE; Filed by Attorney for Defendant
02/21/2018 Continuance of Hearing and Order; Filed by FACEBOOK, INC.

Judy Records (judyrecords.com)

This service is (was) so good that they were sued for providing too much access. While the search function is currently disabled, there is hope that it will return. When functioning, the search feature accepts traditional search operators. By placing my name in quotes, I immediately identified an Illinois Supreme Court case in which I was mentioned. The site is completely free with no business model. I hope it comes back. A direct URL query follows.

https://www.judyrecords.com/getSearchResults/?search="michael bazzell"

Trellis (trellis.law)

While not as abundant as Judy Records, Trellis does offer value for those seeking court records. If you want to query via URL, you must add each state indexed by the service, as demonstrated below. A free account is required to see full court docket information.

https://trellis.law/cases/bazzell?state=az&state=ca&state=ct&state=de&state=fl&state=ga&state=il&state=ma&state=nv&state=nj&state=ny&state=oh&state=pa&state=ri&state=tx&state=wa

FOIA Search (foia.gov/search.html)

The Freedom Of Information Act (FOIA) allows us to demand government records from various entities. Filing a specific request exceeds the scope of this book, but you should know that a lot of public information is already available online. This site appears to rely on Bing to index government websites, so I assumed that this resource was unnecessary. However, my attempts to replicate some of the results within Google were unsuccessful. As an example, I searched for "Darrell Bazzell" on the FOIA site and received a specific result from a 2001 committee session. I replicated this search within Google and could not retrieve the same entry. Therefore, this service should be available within your arsenal of tools. A static submission URL is as follows, and present within the search tool.

https://search.foia.gov/search?affiliate=foia.gov&query=osint

MuckRock (muckrock.com)

This service buries government entities with FOIA requests and then publishes all responses. A direct URL search query structure follows.

https://www.muckrock.com/foi/list/?q=inteltechniques

Open Corporates (opencorporates.com)

Practically every state offers a searchable database of all businesses created or registered within the state. This will usually identify the owner(s), board members, and other associated subjects. **Dun & Bradstreet** (dnb.com) offers a business search within the home page, but many registered companies do not participate with them. In my experience, the best overall business lookup entity is Open Corporates. This free service indexes all 50 states in the U.S. plus dozens of additional countries. The records usually identify corporate officers' names, addresses, and other contact details. The basic search option allows queries by business name, entity registration number, or officer name. Clicking the advanced option allows query by physical address, but requires you to create a free

account. This website is superior to targeted Google queries, because it indexes and scrapes data directly from government websites. This can assist with identifying historical results that no longer appear within the original source. I visit this resource every time I encounter a business name during my research, or identify a target that would likely be associated with an organization. Open Corporates allows search submission via URL as follows.

https://opencorporates.com/companies?q=inteltechniques
https://opencorporates.com/officers?q=bazzell

AIHIT (aihitdata.com)

This service is unique in that it uses artificial intelligence (AI) in order to populate and update business records. I do not know how much of this is legitimate process and not marketing hype, but I consistently find unique data here. The default search options query business names and registration numbers, and usually provide the same results as the previous options. However, the "More Fields" area allows entry of an email address, telephone number, or name of an individual. I have used this option to search personal email addresses and retrieve associated businesses. Fortunately, AIHIT allows search submission via URL as follows.

https://www.aihitdata.com/search/companies?k=email@gmail.com
https://www.aihitdata.com/search/companies?t=Michael Bazzell
https://www.aihitdata.com/search/companies?c=inteltechniques

OCCRP Aleph (aleph.occrp.org)

This self-described "global archive of research material for investigative reporting" includes numerous business records, transcripts, and other data sets. All can be filtered to display email addresses, phone numbers, names, and specific file types. Queries are free, but results might be limited if you are not signed in with a free account. The query URL structure follows.

https://aleph.occrp.org/search?q=michael%20bazzell
https://aleph.occrp.org/search?q=inteltechniques

Open Payrolls (openpayrolls.com)

Open Payrolls might be the largest searchable nationwide government salary database consisting of nearly 85 million salary records from over 14,800 employers. It allows you to locate employee salaries for federal agencies, states, counties, cities, universities, colleges, and K-12 schools. A direct query URL follows.

https://openpayrolls.com/search/michael-bazzell

US Staff (bearsofficialsstore.com)

This odd domain possesses millions of profiles of people associated with government and corporate employment. The data appears to have been scraped from LinkedIn and search queries rely on a Google Custom Search Engine. Searching a company typically reveals all employees. My gut says this site will not be around long.

https://bearsofficialsstore.com/search/?q=michael+bazzell
https://bearsofficialsstore.com/search/?q=inteltechniques

EINTaxID (eintaxid.com)

There is no official way to query a tax ID (EIN) for a company, but this service attempts to uncover those assigned to large companies.

Little Sis (littlesis.org)

This grassroots watchdog network connects the dots between the world's most powerful people and organizations. Searching a name or company can reveal details associated with donations, political support, board members, and other relationships. The query URL follows.

https://littlesis.org/search?q=michael bazzell
https://littlesis.org/search?q=inteltechniques

SSN Validator (ssnvalidator.com)

A simple way to verify if a social security number is valid is at SSN Validator. This does not provide the personal information attached to the number, only verification that the number is valid. A typical response will include the state that issued the number, the year issued, verification that the number was assigned, and confirmation of death if applicable.

Social Security Death Index (genealogybank.com/explore/ssdi/all)

This public index of death records is stored on a genealogy site. The only required information is the first and last name. The results will identify birth year, death year, state of last residence, and state of SSN issue.

Legacy (legacy.com/search)

There are many websites that search for death-related information such as social security indexes and ancestry records. A leader in this area is Legacy. This site indexes online obituaries and memorials from approximately 80 percent of all online newspapers. The search on this site is straightforward and results can identify family members and locations.

Asset Locator (www.blackbookonline.info/assetsearch.aspx)

Black Book Online's Asset Locator is the most comprehensive list of sources for the search of real estate, judgments, bankruptcies, tax liens, and unclaimed funds. This page will allow you to select the type of asset you are researching and the state of the target. This will then create a new page with all the options for that state. It will provide direct links to the sites for a search of the target. This often includes online databases of public employee salaries, vehicle registrations, property tax records, and dozens of other categories.

Voter Registration Records

The elections of 2016 and 2020 caused a lot of controversy in regard to the use and collection of voter registration data. While these personal details are public record, many people did not believe it was appropriate for politicians to use this personal data as a part of their campaign strategies. Regardless of your opinion on these matters, much of the voter registration details are available online. The most beneficial sites I have found are **Voter Records** (voterrecords.com) and **VoterRef** (voteref.com). You can search both by name or browse by state. Any results will identify full name, home address, mailing address, gender, party affiliation, age, and relatives. Currently, databases are available for Alaska, Arkansas, Colorado, Connecticut, Delaware, Florida, Michigan, Nevada, North Carolina, Ohio, Oklahoma, Rhode Island, Utah, and Washington.

Vehicles

Many people assume that information related to vehicle registration and licensing is only available to law enforcement through internal networks. While a full driver's license search and complete license plate query is not publicly available, a surprising portion of related data is online for anyone to view. The following methods will display all publicly available details.

Department of Transportation (vpic.nhtsa.dot.gov)

The DOT has a website which provides free access to vehicle identification number (VIN) data. All information on this website is public information, and the data comes from vehicle manufacturers. You can search by VIN within their database to find detailed vehicle information. I submitted a unique VIN and received the following response.

Year: 2010	Trim Level: TDI
Make: VOLKSWAGEN	Style: SEDAN 4-DR
Model: JETTA	Manufactured: Mexico
VIN: 3VWRL7AJ6AM13xxxx	Weight: 4,500 lbs

The following options also allow you to enter any VIN and retrieve the year, make, and model of the vehicle associated. The first option will often display estimated mileage based on service records.

Vin decoderz (vindecoderz.com)
Check That VIN (checkthatvin.com)
Search Quarry (searchquarry.com)
FaxVin (faxvin.com)
Vehicle History (vehiclehistory.com)
VinCheck (vincheck.info)

NICB VIN Check (nicb.org/vincheck)

While the previous searches will identify details about vehicles, they will not display any information about theft or salvage records. The National Insurance Crime Bureau (NICB) allows a search of any VIN and will display two unique pieces of information. The VINCheck Theft Record will identify vehicles that have been reported stolen, while the VINCheck Total Loss Records identifies VINs that belong to salvaged vehicles.

Cycle VIN (cyclevin.com)

VINs from motorcycles may not be searchable on standard VIN engines due to the number of characters in them. Cycle VIN will display a year and make, as well as any indication that the VIN exists in its proprietary database. If it does, $25 will obtain title and mileage information. I only use this as a free resource for verifying motorcycle VINs to the correct year and make.

Vehicle Registration

Several free services identify the year, make, and model of a vehicle after supplying the VIN. However, it is more likely that we know the license plate registration details rather than a VIN. Fortunately, we have many options for researching these plates. The following services provide a search option based on the vehicle registration plate and state. Results are hit-or-miss, and rarely include a name, but many will identify the VIN for further research.

Auto Check (autocheck.com)
Records Finder (recordsfinder.com/plate)
CarFax (carfax.com/vehicle-history-reports)
Search Quarry (searchquarry.com/vehicle_records)
Free Background Search (freebackgroundcheck.org)
Carvana (carvana.com/sellyourcar/getoffer/vehicle)
VinCheck (vincheck.info/free-license-plate-lookup/)
Vehicle History (vehiclehistory.com/license-plate-search)
Find By Plate (findbyplate.com)

While the previous license plate search websites offer a straight-forward query option, they are limited to the data available within publicly-traded vehicle databases. California plates often reveal accurate data, but states such as South Dakota and Montana are not as generous with the sharing of their own registrations. Because of this, we may want to query services which pay a fee in order to access premium data sets. My favorite of these is **Kelley Blue Book** (kbb.com/instant-cash-offer). The premise of this site is to identify the value of your own vehicle and potentially receive an offer to purchase it from KBB. That may have a benefit to you, but I prefer to harness the OSINT capabilities of this service.

As a test, I submitted a vehicle's registration number which was displayed on a television show playing in the background while I wrote this section. The result correctly identified the vehicle as a 2010 Dodge Avenger. This example was a California plate, so I felt like I was cheating a bit. Instead, I conducted a search of the license plate "MRBIG" within South Dakota and Wyoming, which are both known to protect vehicle registration data from public view. The results appear below.

2008 MAZDA MX-5 Miata

VIN: JM1NC26FX80142016

2019 Ram 2500 Crew Cab

VIN: 3C6UR5DL7KG684611

While only a small piece of information, this works in conjunction with other search techniques. After exhausting all of these searches, you should be able to obtain the VIN, make, model, year, engine, and style of the vehicle. These options will not typically provide the name of the owner.

O'Reilly Auto Parts (oreillyauto.com)

If the previous search options fail to identify the VIN, year, make, and model of a vehicle based on the license plate, try O'Reilly. As a service to potential customers, it allows you to enter your license plate in order to identify applicable parts for your vehicle. Select the "Shop by Vehicle" in the upper-right and enter the license plate and state. I provided the state of California and the plate of "HACKER". I received the following response.

2007 GMC Sierra 1500 SLE
V8,6.0
5967,364,Electronic
SFI,GAS
FI,MFI

When I searched the same plate on VinCheck, I received a VIN of 3GTEK13Y87G527460. I now have a decent beginning to my investigation of a license plate.

Progressive (progressive.com)

While not an official vehicle search, the insurance provider Progressive offers an interesting piece of information. I first learned about this technique from S.L., a member of my online OSINT forum. When you view the home page at progressive.com, you are prompted to request a free insurance quote. If you provide the zip code and address of any target, you receive a summary of the year, make, and model of all vehicles registered at that address. You can supply any random data besides the physical address and receive the results. This was likely designed to make the quote process more efficient and accurate, but investigators should appreciate the free utility.

Marine Traffic and Boat Information

There is an abundance of details available about global marine traffic within ownership records and real-time monitoring. Marine Traffic (marinetraffic.com) provides an interactive map that displays the current location of all registered ships and boats. Clicking on any vessel provides the name, speed, collection time, and destination. Boat Info World (boatinfoworld.com) allows the search of a boat name and provides the following details.

Boat Name	Lloyd's Registry Number	Vessel Build Year
Boat Owner	Call Sign	Ship Builder
Record Date	Coast Guard Vessel ID	Hull Shape
Registered Address	Service Type	Propulsion Type
Hull ID	Boat's Length	
Hailing Port	Boat's Gross Tons	

Aircraft Information

Monitoring aircraft during flight and searching historical ownership records is relatively easy. Commercial planes constantly announce their location with automated reporting systems and tail numbers act similarly to a vehicle's registration plate. Today, this information is publicly available on multiple websites. Plane Finder (planefinder.net) displays an interactive global map identifying all known aircraft currently in flight. Hovering over a selection displays the carrier, flight number, originating departure, destination, speed, and altitude. Historical ownership records are available on multiple websites and none are completely accurate. I recommend Black Book Online's aviation page (www.blackbookonline.info/Aviation-Public-Records.aspx). At the time of this writing, it provided direct links to the following databases.

Aircraft N-Number Search	Certified Pilots
Aircraft Ownership Search	Cockpit Voice Recorder Database
Airline Certificates	Flight Tracker
Airport Profiles	Military Aviation Crash Reports

Campaign Contributions

Any contributions to political campaigns are public record. Searching this is now easy thanks to three separate websites. These sites will search using information as minimal as a last name. Including the full name and year will provide many details about the target. This includes occupation, the recipient of the contribution, the amount, the type of contribution, and a link to the official filing that contains the information. After an initial search is conducted, you will receive additional search tabs that will allow you to filter by zip code, occupation, and year. Melissa Data allows you to search a zip code and identify all political donations for a specified year. The results from these sites may be redundant, but often contain unique data.

Open Secrets (opensecrets.org)
Money Line (politicalmoneyline.com)
Melissa Data (melissadata.com/v2/lookups/fec/index)

Selective Service Verification (sss.gov/Home/Verification)

This website requires a last name, social security number, and date of birth of the target. The result will identify the person's full name, selective service number, and date of registration.

High Programmer (highprogrammer.com/cgi-bin/uniqueid)

Most states use some type of algorithm to create a driver's license number for a person. Often, this number is generated from the person's name, sex, and date of birth. After you have determined your target's middle initial and date of birth from the previous websites mentioned, you can use this data to identify the target's driver's license number. High Programmer will automate this process for the following states.

Florida	Michigan	New York
Illinois	Minnesota	Washington
Maryland	New Hampshire	Wisconsin

BinDB (www.bindb.com/bin-database.html)

While not technically government data, I felt that this option fits best in this chapter. This website will allow you to enter the first six digits of any credit card number and identify the brand, issuing bank, card type, card level, country, bank website, and customer care line.

Real World Application: While working in the homicide division, I often identified credit or debit card numbers of my victims. If the actual card was located, I did not need this service. However, if only the number was located, this service helped to identify the financial institution and a contact number. In one specific investigation, I had learned that the victim had eaten at a local restaurant the evening prior to her suspicious death. Visiting the restaurant allowed me to acquire the debit card that she used for payment. Searching this number through BinDB identified the issuing bank. Calling presented an automated self-service feature for members of that bank. Entering the newly found debit card number and the zip code of the victim allowed me to access the previous 30 days of charges to her account. This identified an unknown ATM withdrawal on the day of her killing. Retrieving video from that ATM machine displayed a passenger in her vehicle. This initiated a new investigation which eventually led to the killer.

Criminal Information

If a target has a criminal past, there is probably evidence of this on the internet. County court searches will identify most of this information, but this requires a separate search on each county's website. There are a handful of services that attempt to locate nationwide information by name. **Family Watch Dog** (familywatchdog.us) is one of the leading sites in identifying public criminal information about sex offenders. The main page includes a "Find Offender" tab. You can search here by address or name. The name search only requires a last name to display results. This will identify registered sex offenders that match the criteria specified. This will include a photograph of the target and details of the offense.

Both federal and state prisons offer prisoner details online. The amount of detail will vary by state, but most will include photographs of the target and details of the crime. In most states, this information is maintained in public view after the target is released, if the subject is still on probation or parole. Federal prisoners can be located at www.bop.gov/inmateloc. A first and last name is required for a search. Each state maintains its own database of prisoner information. Conducting a search on Google of "Inmate locator" plus the state of interest should present official search options for that state.

VINELink (vinelink.com) is an online portal to VINE, a victim notification network. VINE has been providing victims and concerned citizens with information for decades, allowing individuals to access reliable information about custody status changes and criminal case information. After choosing the state of interest, you can select from the following options.

Find an Offender: Get info and register to be notified of custody status changes.
Find an Offender Court Case: Get info and register to be notified of offender court dates.
Find Sex Offender Registry Status: Get info about sex offender registry status changes.
Find a Protective Order: Get info and register to be notified of protective order status changes.

Warning: Every city, county, state, and country reports criminal matters uniquely. While you may find something of interest through traditional OSINT resources, absence of online criminal information does not indicate that a crime was not committed. More often than not, I cannot locate online information about a known crime or criminal. These resources should be considered secondary to a detailed inquiry through the courts or law enforcement.

IntelTechniques Business & Government Tool (https://inteltechniques.com/tools/Business.html)

While minimal, this tool should assist with replicating some of the searches mentioned within this chapter.

CHAPTER TWENTY-SEVEN
VIRTUAL CURRENCIES

In simplest terms, virtual currencies can be spent for goods and services without connection to a person or bank account. It has no physical presence, and is mostly used online as digital payment. Bitcoin is virtual currency. A bitcoin address, which is an identifier you use to send bitcoins to another person, appears similar to a long string of random characters. In our demo, we will use 12t9YDPgwueZ9NyMgw519p7AA8isjr6SMw, which is the real address that was used to collect ransom from victims after malicious software had taken over their computers. Think of a Bitcoin address as an email address. That address stores their "virtual" money.

Blockchain (blockchain.info)

This website allows search of a Bitcoin address and displays the number of transactions, total amount of Bitcoin received ($), final balance, and a complete transaction history. We can track every incoming and outgoing payment. This will almost never be associated with any real names, but it provides a great level of detail about the account. We learn that this account has received 19.12688736 Bitcoin worth $ 287,391.14 USD at the time of this writing.

Bitcoin Who's Who (bitcoinwhoswho.com)

Our next stop is a service that provides a bit more analysis about the suspect account. We immediately learn that it is a suspect ransomware account, and that the address has appeared on various news outlet websites. Furthermore, we see transaction IP addresses, which are likely behind VPNs. Overall, I use Blockchain for transaction details and Bitcoin Who's Who to get a better idea of why I might care about the account.

BlockChair (blockchair.com)

This service is very similar to Blockchain, but I find it has better representation across multiple virtual currencies. Additionally, we can query each currency via URL, which will assist in our tools. Let's start with a search of a Bitcoin address at the following URL.

https://blockchair.com/bitcoin/address/1EzwoHtiXB4iFwedPr49iywjZn2nnekhoj

The results are typical, and include balance and transaction data. The power of BlockChair is the ability to search Bitcoin, Ethereum, Ripple, Bitcoin Cash, Litecoin, Bitcoin SV, Dash, Dogecoin and Groestlcoin. We will use the following URLs for each, replacing "xxx" with the target address.

https://blockchair.com/bitcoin/address/xxx
https://blockchair.com/ripple/address/xxx
https://blockchair.com/bitcoin-cash/transaction/xxx
https://blockchair.com/litecoin/address/xxx
https://blockchair.com/bitcoin-sv/address/xxx
https://blockchair.com/dash/address/xxx
https://blockchair.com/dogecoin/address/xxx
https://blockchair.com/groestlcoin/address/xxx

Bitcoin Abuse (bitcoinabuse.com)

This service focuses on one feature. It notifies you if others have reported a target virtual currency address as associated with malicious activity. This often provides valuable information about an investigation. Consider an actual report located at the following URL.

https://www.bitcoinabuse.com/reports/1KUKcwCv64cXQZa4csaA1cF3PPTio6Yt2t

The results include a summary of the activity and the email addresses sending malicious email.

Sep 21, 2019	sextortion	peter6389dd@excite.co.uk	"Hacked computer email"
Sep 21, 2019	ransomware	addntfogjnfi@activeware.com	Claims to hack computer

Wallet Explorer (walletexplorer.com)

The previous utilities examined an individual virtual currency account, such as a Bitcoin address. Many people possess numerous addresses and store them all within a virtual wallet. This is where Wallet Explorer can be extremely beneficial. While researching one of our target Bitcoin addresses within this free service, the results identified a wallet of "00037fd441" which contained the target address. Clicking on the link to this wallet revealed multiple new transactions from additional Bitcoin addresses previously unknown. This step is vital in order to track all transactions associated with your suspect. The following URLs search an address and a wallet.

https://www.walletexplorer.com/address/1EzwoHtiXB4iFwedPr49iywjZn2nnekhoj
https://www.walletexplorer.com/wallet/00037fd441938ba4

Virtual Currency APIs

In order to create the custom search tool presented at the end of this chapter, I needed a very simple way to query virtual currency addresses for various tasks. Many of the websites which allow searching of Bitcoin addresses do not permit submission via URL. Instead, I will take advantage of various Application Programming Interfaces (APIs) which allow us to query directly and receive a text-only result. The following URLs are used within the tool, with an explanation of each. Each display of "xxx" is where the virtual currency address or amount would be inserted.

Validation: The following URL provides an indication whether a provided address is valid or invalid. This is a great first search to make sure you have a proper address. The response will be the Unix date the address was first seen within the network. An error notifies you of a bad address.

https://blockexplorer.com/api/addr/xxx

Value: The following URL presents the current value of one Bitcoin.

https://blockchain.info/q/24hrprice

Received: This URL displays the total amount of Bitcoin received by a specific address. It is important to note that this amount will be in "Satoshi". A Satoshi is equal to 0.00000001 Bitcoin. Put another way, one bitcoin contains 100 million Satoshis. This unit of measurement is popular because a single Bitcoin is currently worth approximately $19,000. The Satoshi is a more precise number. In a moment, we will convert Satoshi to USD.

https://blockchain.info/q/getreceivedbyaddress/xxx

Sent: This URL displays the total amount of Bitcoin sent by a specific address. It is important to note that this amount is also presented in "Satoshi" (0.00000001 Bitcoin).

https://blockchain.info/q/getsentbyaddress/xxx

Balance: This utility displays the current balance of an address in "Satoshi".

https://blockchain.info/q/addressbalance/xxx

BTC > USD Value: The following URL will always display the current value of any amount of Bitcoin in USD. This price fluctuates hourly. Replace "xxx" with your value of Bitcoin.

https://api.exchangerate.host/convert?from=BTC&to=USD&amount=xxx

USD > BTC Value: The following URL will always display the current Bitcoin value of any amount of USD. This price fluctuates hourly. Replace "xxx" with your value of USD.

https://blockchain.info/tobtc?currency=USD&value=xxx

Satoshi > USD Value: The following URL will always display the current value of any amount of Satoshi in USD. This price fluctuates hourly. Replace "xxx" with your value of Satoshi.

https://api.exchangerate.host/convert?from=SAT&to=USD&amount=xxx

USD > Satoshi Value: The following URL will always display the current Satoshi value of any amount of USD. This price fluctuates hourly. Replace "xxx" with your value of USD.

https://api.exchangerate.host/convert?from=USD&to=SAT&amount=xxx

Summary: This URL displays a brief summary of a Bitcoin address including total received, total sent, balance, total transactions, first transaction, and most recent transaction. Replace "xxx" with your target address.

https://chain.api.btc.com/v3/address/xxx

First seen: This Blockchain query displays the date which a virtual currency address transaction was first seen within the public blockchain. Note that this result will appear in Unix time format, but our tools will allow you to convert this to traditional time format. Replace "xxx" with your virtual currency address.

https://blockchain.info/q/addressfirstseen/xxx

Investigation Summary

Now that you understand the details available about a virtual currency address, let's run through a typical investigation. Assume you are investigating a Bitcoin address of 1EzwoHtiXB4iFwedPr49iywjZn2nnekhoj. It was used as part of an extortion email, and you have been tasked to find any information about the address. First you input the address into the search tool. The following information would be presented after each of the options.

BTC Validation: Valid (The address is a proper format)
1 BTC Price: $19,978.23 (The current value of one Bitcoin)
Satoshi Received: 716409285544 (The total amount of received currency)
Satoshi Sent: 716371585974 (The total amount of sent currency)
Satoshi Balance: 37699570 (The total amount of the current balance)
Satoshi > USD: (Used to convert Satoshi to USD as follows)
    Received: $136,925,336.62
    Sent: $136,914,950.52
    Balance: $7,205.23
Summary:
    "address": "1EzwoHtiXB4iFwedPr49iywjZn2nnekhoj",
    "received": 716409285544,
    "sent": 716371585974,
    "balance": 37699570,

"tx_count": 3534,
"unconfirmed_tx_count": 0, "unconfirmed_received": 0, "unconfirmed_sent": 0,
"unspent_tx_count": 3,
"first_tx": "6cc1542feb7abcff6364e0d31fc75097e0ecf7dae897ad6de6a2c1c5a1261316",
"last_tx": "e11525fe2e057fb19ec741ddcb972ec994f70348646368d960446a92c4d76dad"

Creation Date: 1331482301 (Unix time when the address was first seen)
Date Conversion: Mar-11-2012 10:11:41 (Time in UTC)
Blockchain: A detailed list of all transactions.
BitcoinAbuse: One report of malicious activity and a new email address to research.
BitcoinWhosWho: Links to online references to the address on Reddit.
WalletExplorer Address: Transaction details and wallet ID of 00037fd441.
WalletExplorer Wallet: Several pages of additional Bitcoin addresses within the suspect's wallet.

You could now repeat the process with the new Bitcoin addresses with hopes of identifying more email addresses. The email address search tool may identify further information about your target. While this is all very time consuming, the tools should simplify the queries.

Scam Search (scamsearch.io)

This free service was previously explained as a resource for searching email addresses, usernames, and telephone numbers to identify association with online scams. It can also be used to query virtual currency addresses. Searching "1FVuyuSN41aa3JN9sn8qkuD2PmaMEMHHnc" reveals email addresses, IP addresses, and locations associated with an online extortion suspect.

IntelTechniques Virtual Currency Tool

This tool simplifies the various techniques explained in this chapter. Each option, including API requests, open in a new browser tab. Figure 26.01 displays the tool.

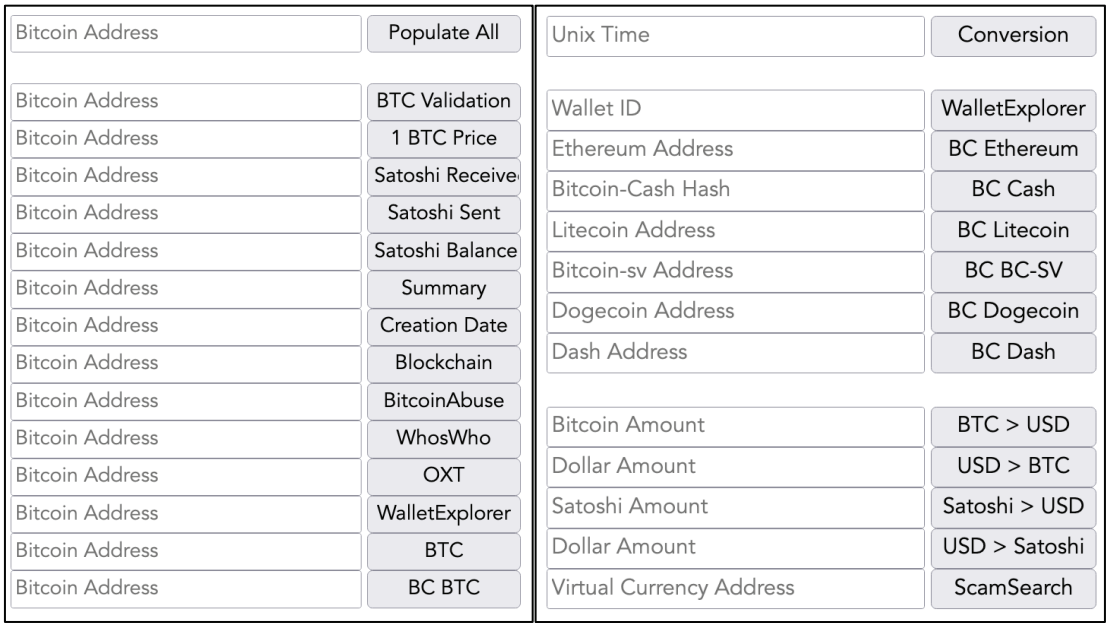


Figure 26.01: The IntelTechniques Virtual Currency Tool.

CHAPTER TWENTY-EIGHT
APPLICATION PROGRAMMING INTERFACES (APIs)

The last few editions of this book eliminated the chapter devoted to Application Programming Interfaces (APIs). Instead, I included API options within the chapters which were appropriate for a specific service. As one example, I offered reverse caller ID API terminal commands within the chapter about telephone numbers. That made sense at the time. Over the past year, I have seen API access become more complex and some services have eliminated the possibility to access their API within a traditional web browser. Therefore, I have moved any API service which either requires an API key, or mandates that access be gained through a non-browser option such as Terminal, within this chapter. I also created a new API search tool page which can be found in both the offline tools previously explained and online at **https://inteltechniques.com/tools/API.html**. **At the end of this chapter, I explain how to access a script which has all of the code presented within this chapter for easy usage (or copying and pasting). Please don't try to retype all of these commands manually.** We have a lot to discuss. However, we should probably back up and understand the API.

An application programming interface (API) enables companies to offer their data and functionality to external third-party developers and within their own products. This allows customers (you) to communicate directly with a company's data without navigating to a web page, logging in, and clicking buttons. It also allows companies to deliver fancy content without the user needing to understand how databases and other technology works. When you are on Facebook and click a person's profile, Facebook is acquiring that data through a series of internal APIs and servers. This is true for most online services which have a search function. Think of the API as the protocol which delivers the text which gets turned into pretty pages in your browser. This is a very basic explanation, but enough to get us to some demonstrations.

I have already mentioned APIs and offered a few techniques previously in this book. Much earlier, I explained how you could use the Pushshift API to access deleted Reddit posts. The result was raw text data within your browser. Accessing their API was very easy if we knew the exact URL to access their data. Since Pushshift does not require a private API key, I left that technique in the place where it was most useful. Everything within this chapter will require an API key, which is similar to a password issued only to you. You will need to provide that key every time you request data from a service. At the end of this chapter, I will offer a custom script which can be modified to include your own API keys. Afterward, you will only need to supply the target data you are researching, and never enter the API key again. First, let's work through the process manually. We will start with People Data Labs.

People Data Labs (peopledatalabs.com)

I am a bit hesitant to present this resource because it seems too good to be true. This "people data" collection company offers 1,000 free queries of their premium data sets to anyone, and they accept masked email addresses such as 33mail, Anonaddy, and Simple Login. We have seen other similar companies offer the same type of deal, such as Full Contact and Pipl, only to convert to a paid model without any type of trial. People Data Labs may be planning the same marketing strategy, but let's take advantage of the resource while it is available. You must create a free trial account at the website under the option of "Get API Key". I recommend providing an email address associated with a personal domain for best results. You want to appear as legitimate as possible. Your new account will provide a unique API key for your usage within the "Dashboard" of the page. Mine appeared similar to the following.

```
5c0ck097aa376bb7741a1022pl2222e3d45chs
```

Since this service does not provide a traditional search option, we must create URLs for our queries. This URL can be entered into a web browser or script. We will keep things simple for now. The following URL submits a query for "sean@peopledatalabs.com" in any browser.

```
https://api.peopledatalabs.com/v5/person/enrich?pretty=true&api_key=5c0ck097aa
376bb7741a1022pl2222e3d45chs&email=sean@peopledatalabs.com
```

The results are presented as text in JSON format. The following is a partial summary. The full result included four pages of details. I found the following most beneficial.

```
"full_name": "sean thorne",
"birth_year": "1990",
"linkedin_url": "linkedin.com/in/seanthorne",
"linkedin_id": "145991517",
"facebook_url": "facebook.com/deseanthorne",
"facebook_id": "1089351304",
"twitter_url": "twitter.com/seanthorne5",
"work_email": "sean@peopledatalabs.com",
"mobile_phone": "+14155688415",
"email address": "sthorne@uoregon.edu",
"email address": "sean@hallspot.com",
"education": "university of oregon",
```

While this is an obviously staged demo, the results are impressive. I have been able to convert a personal email address of a target into a full resume with social network profiles and cellular telephone number. We can also use that same API key to query by a telephone number. The following queries a U.S. number of 222-555-1212 with my test API key.

```
https://api.peopledatalabs.com/v5/person/enrich?pretty=true&api_key=5c0ck097aa
376bb7741a1022pl2222e3d45chs&phone=+12225551212
```

The results are once again impressive. As an example, I provided a private direct office number of a previous government colleague. The full result would fill many pages, but I have provided a partial summary of the best parts below. I replaced any sensitive details with "REDACTED".

```
"full_name": "emily REDACTED",
"gender": "female",
"linkedin_url": "linkedin.com/in/REDACTED",
"linkedin_username": "REDACTED",
"linkedin_id": "REDACTED",
"facebook_url": "REDACTED",
"personal_emails": "REDACTED@charter.net"
"industry": "law enforcement",
"job_title": "police officer",
"job_last_updated": "2022-08-29",
"location_metro": "st. louis, missouri",
"location_street_address": "REDACTED drive",
"location_postal_code": "REDACTED",
"location_geo": "REDACTED",
"location_last_updated": "2022-08-31",
"company": "illinois law enforcement training and standards board",
"title": "instructor",
"school": "southern illinois university edwardsville",
```

Within seconds, I identified the name, gender, LinkedIn profile, Facebook profile, personal email, profession, location, home address, secondary employment, and education details associated with my target telephone number. I even know when she last updated these details. I cannot overstate the value of this free trial. We can also query by social network profile, such as a Twitter, Facebook, or LinkedIn username. The following URL structure should be used with your own API key.

Twitter: `https://api.peopledatalabs.com/v5/person/enrich?pretty=true&api_key=5c0ck097aa376bb7741a1022pl2222e3d45chs&profile=www.twitter.com/inteltechniques`

Facebook: `https://api.peopledatalabs.com/v5/person/enrich?pretty=true&api_key=5c0ck097aa376bb7741a1022pl2222e3d45chs&profile=www.facebook.com/inteltechniques`

LinkedIn: `https://api.peopledatalabs.com/v5/person/enrich?pretty=true&api_key=5c0ck097aa376bb7741a1022pl2222e3d45chs&profile=www.linkedin.com/inteltechniques`

Let's take a look at the Twitter result for my own name. It immediately identified the three locations which have been present within my Twitter profile over the past decade, as follows. I am not aware of any other service which could replicate this data going back to 2009.

```
"st. louis, missouri, united states",
"washington, district of columbia, united states",
"new york, new york, united states"
```

I submitted a query for our previous Facebook target (facebook.com/zuck) and received the following partial result.

```
"full_name": "mark zuckerberg",
"birth_year": "1984",
"linkedin_url": "linkedin.com/in/mark-zuckerberg-a513498b",
"facebook_url": "facebook.com/zuck",
"mobile_phone": "+16506447386",
"emails": "mzuckerb@fas.harvard.edu",
"education": "harvard university",
```

This is powerful data, especially considering a social network profile was the starting point. All of these queries can be completed within your web browser, but I hope you will consider the script which I will explain momentarily. Let's move on to reverse caller ID options.

Twilio (twilio.com)

I previously explained my usage of the Twilio Lookup service as my favorite telephone number query option. However, it has one annoyance. Every time I want to search a number, I must log in to Twilio; confirm my 2FA; navigate to the Lookup option; and receive my results embedded into their web portal. That is not extremely difficult, but it does slow me down. That is why I only rely on their API for my everyday queries. Assuming my Twilio Account SID was 11111; my Twilio API key was 22222; and my target U.S. number was 202-555-1212, the following command copied and pasted into Terminal would conduct my query.

```
curl -X GET 'https://lookups.twilio.com/v1/PhoneNumbers/+12025551212?
Type=caller-name&Type=carrier' \
-u 11111:22222 | python3 -mjson.tool
```

There is a lot to unpack here. Let's first understand where we get the information required. Once you have a Twilio trial established, the home page of your portal should display your "Account SID" and "Auth Token". The Auth Token is the same as an API key for our purposes. Next, let's work through that command, which is actually two lines, but prints as three for the book formatting.

`curl`: This is our command which will download data from a URL.
`-X`: This specifies a request method for communication with the server.
`GET`: This is the method we want to use (GET).
`'https://lookups.twilio.com/v1/PhoneNumbers/`: This is the URL for Twilio's Lookup API.
`+12025551212`: This is our target telephone number.

`?Type=caller-name&Type=carrier':` This tells Twilio we want both the name and carrier data.
`\:` This inserts a new line without executing just yet.
`-u 11111:22222:` This is our Twilio Account SID and Auth Token.
`| python3 -mjson.tool:` This displays the date in an easier format to read.

The following is a redacted result when I searched one of my Google Voice numbers which has been leaked throughout various services.

```
   "caller_name": "MICHAEL BAZZELL",
   "caller_type": "CONSUMER",
   "country_code": "US",
   "carrier": {
        "mobile_country_code": "311",
        "mobile_network_code": "910",
        "name": "Google (Grand Central) - Level3 - SVR",
        "type": "voip",
```

Once again, I find this impressive. Not only did Twilio identify the name of the VOIP account owner, it also identified the carrier (Google). I use this lookup tool every day. Twilio also offers additional Lookup services, which may provide additional value to your investigations. These each withdraw an additional small fee from your trial balance, and some require a verified or "upgraded" account to function. I don't use these often, but you may find enough value to justify transitioning to a paid account.

First, you must "Activate" any desired additional telephone query service. While in your Twilio portal (or console), click "Explore Products" and scroll down to the "Add-Ons" option under "Marketplace". Here you will find many options, and I explain only those which I have used and find valuable. Clicking the "Activate" option will display a notification about pricing per search. Once you have activated an add-on, you will see a new tab called "Configure" within the specific product. This tab presents the "Unique Name" for that product, which we will use within our queries. Assuming my Twilio Account SID was 11111; my Twilio API key was 22222; and my target U.S. number was 202-555-1212, the following commands would conduct my queries with each of the extended add-on services. I offer a brief explanation before each command and a typical result after.

Twilio Phone Validation (Identify connected or disconnected numbers with line types):

```
curl -X GET 'https://lookups.twilio.com/v1/PhoneNumbers/+12025551212?AddOns=
real_phone_validation_rpv_turbo' \
-u 11111:22222 | python3 -mjson.tool
```

Twilio Phone Validation Result:

```
   "status": "connected",
   "iscell": "N",
   "carrier": "Level 3 Communications, LLC",
```

Twilio OpenCNAM (Alternative Caller ID Provider):

```
curl -X GET 'https://lookups.twilio.com/v1/PhoneNumbers/+12025551212?AddOns=
telo_opencnam' \
-u 11111:22222 | python3 -mjson.tool
```

Twilio OpenCNAM Result:

```
   "name": "MICHAEL BAZZELL",
   "number": "+16185551212"
```

Twilio DataAxle (Business Data):

```
curl -X GET 'https://lookups.twilio.com/v1/PhoneNumbers/+12025551212?AddOns=
dataaxle_bizinfo' \
-u 11111:22222| python3 -mjson.tool
```

Twilio DataAxle Result:

```
   "website": "https://inteltechniques.com",
   "city": "Los Angeles",
   "name": "IntelTechniques",
   "facebook_url": "https://www.facebook.com/inteltechniques",
   "phone": "(618) 555-1212",
   "state": "CA",
   "street": "1700 Main",
   "first_name": "Michael",
   "last_name": "Smith",
   "titles": "Senior Vice President"
```

Twilio Ekata Search (Name and Address):

```
curl -X GET 'https://lookups.twilio.com/v1/PhoneNumbers/+12025551212?AddOns=
ekata_reverse_phone' \
-u 11111:22222| python3 -mjson.tool
```

Twilio Ekata Search Result:

```
   "phone_number": "+16184633505",
   "alternate phone_number": "+16184631967",
   "line_type": "FixedVOIP",
   "is_commercial": true,
   "name": "Jody Allen O'Guinn",
   "relation": "PotentialOwner",
   "line_type": "FixedVOIP",
   "carrier": "Charter Fiberlink",
   "street_line_1": "1700 E Broadway",
   "postal_code": "62002",
   "latitude": 38.88788,
   "longitude": -90.16033,
   "accuracy": "RoofTop"
```

Twilio Ekata Validate (Identify Fraud and Risk Groups):

```
curl -X GET 'https://lookups.twilio.com/v1/PhoneNumbers/+12025551212?AddOns=
ekata_phone_valid' \
-u 11111:22222| python3 -mjson.tool
```

Twilio Ekata Validate Result:

```
   "warnings": [],
   "country_calling_code": "1",
   "is_valid": true,
   "line_type": "Premium",
   "country_code": "US",
   "carrier": "Bandwidth",
```

Twilio NoMoRobo (Identify Spam Callers):

```
curl -X GET 'https://lookups.twilio.com/v1/PhoneNumbers/+12025551212?AddOns=
nomorobo_spamscore' \
-u 11111:22222| python3 -mjson.tool
```

Twilio NoMoRobo Result:

```
   "score": 1,
   "neighbor_score": 1
```

Twilio TrueSpam (Identify Spam Callers):

```
curl -X GET 'https://lookups.twilio.com/v1/PhoneNumbers/+12025551212?AddOns=
truecnam_truespam' \
-u 11111:22222| python3 -mjson.tool
```

Twilio TrueSpam Result:

```
   "spam_score": 91,
   "spam_score_match": 1
```

There are additional options within the Twilio Add-ons catalog, but I have not found them to have much value for OSINT investigations. You should peruse the entire catalog for any new options which may have appeared since publication of this book. I feel obligated to remind readers that I rarely use any Twilio Add-ons. I find the basic name and carrier queries to suffice for most of my needs, and they don't eat away at my free trial balance as bad as the extended options.

Telnyx (telnyx.com)

If I could only search two telephone number services, this would be my second pick after Twilio. I found this service in late 2020 and I am impressed. The website offers a trial account with $5.00 in free queries. However, they are extremely scrutinous of new accounts. They will likely ask you how you will be using their service, and I encourage you to revisit the Twilio registration hint previously given. Once you do obtain a trial, it could last several years with minimal usage. Unfortunately, there is no way to submit a single URL as a query within a web browser. You must submit a request within Terminal, similar to Twilio, which includes several pieces of data. The following is the submission structure.

```
curl -X GET \
      --header "Content-Type: application/json" \
      --header "Accept: application/json" \
      --header "Authorization: Bearer XXXX" \
      "https://api.telnyx.com/v2/number_lookup/+12025551212?type=carrier&type=
caller-name"
```

You could do this manually, replacing "XXXX" with your Telnyx API key (visible within your portal) and "2025551212" with your target number, but that is complicated. Instead, consider using my custom script, as explained at the end of this chapter.

CID Name (cidname.com)

Similar to Twilio and Telnyx, you must create a free trial account in order to use this service. You must provide an email address and telephone number, and I have used temporary addresses and Google Voice to bypass this restriction. Navigate to this website and register for a free account. Upon completion, you should receive an

email with an API license key that is valid for approximately 100 free successful searches. You will not be charged for empty results. If you do not receive the API key via email, log in to your account and retrieve it from the portal.

You are now ready to submit requests for caller ID information. To do this, you must formulate an API request in your browser or Terminal which includes your authentication key and target telephone number to search. The following is an example.

```
https://dip.cidname.com/2025551212/XXXXXXXXX&output=raw&reply=none
```

This queries the domain (cidname.com), our authentication key (XXXXXXXXX), and our target number (2025551212). I recommend saving the address of your first query as a bookmark or favorite if you plan to use this through a browser instead of our automated script presented in a moment. I currently only recommend CID Name if you are unable to create a Twilio or Telnyx account, or if you simply want another service to verify the confidence in a previous result.

Caller ID Summary

Before reverse caller ID lookups, this information would have required a subpoena. In one recent scenario, a cellular telephone number searched on one of these services revealed the name "Jennifer S" in the result. During an interview with this subject, she disclosed that "Jennifer S" is how she identifies her account on the telephone bill that she shares with other family members. She was unaware that this data was sent to the receiving number. On many searches, the full name will be present. This should explain why you may be noticing a caller's name on your caller ID display when he or she is calling from a cellular number.

There are many caller ID options available on the internet and I encourage you to investigate any companies which have surfaced since this research. Most services offer a free trial, even if it is not advertised. Adding a few dollars to each service may provide more queries than you will ever need. Take advantage of the free trials to determine which services work best in your investigations.

In 2019, I purchased premium memberships through these services. A $10 purchase provided over 1,000 queries at each provider, and I still have credits today. I updated my API keys within my custom script (explained soon), which I rely on every day. Spending a few minutes modifying your own script within your Linux VM will pay off ten-fold within the first week. I often avoid all telephone search websites when I can quickly identify a number's owner without opening a browser.

Overall, reverse caller ID services can tell us more about a target telephone number than the standard people search engines. In many cases, you can immediately obtain data that would have required a subpoena just a few years prior. Always utilize all of the services in order to gauge the confidence in the results. If this is overkill for your needs, there are other web-based search engines that are easier to use.

Removed Caller ID Services

In previous editions of this book, I explained caller ID lookup features of services including EveryoneAPI, BulkCNAM, OpenCNAM, CallerIDService, and Service Objects. I no longer present these options for various reasons. EveryoneAPI and OpenCNAM were acquired by Neustar, and no longer offer free trials of the service. BulkCNAM became Bulk Solutions and suffers the same fate. CallerIDService appears to have shut down completely. Service Objects only offers a one-week trial now, and the results are not great. In my experience, numbers which are not identified through Twilio or Telnyx would also not be present within these removed services. Instead of maintaining numerous accounts, I find it better to simply focus on the most reliable and easiest options.

Full Contact (fullcontact.com)

This service was present in the first several editions of this book but removed from the previous two versions. In previous years, one could access their API from any web browser. However, in 2019 they terminated their original API and transitioned all of their customers to a more complex option. Since my previous editions of this book focused on browser-based access, I stopped including this service within the content. However, we are now ready to tackle the new API format, and the script I keep teasing can execute our queries on our behalf. Therefore, let's revisit Full Contact.

Full Contact was one of the first "enrichment" APIs I ever tested. I provided an email address and this service gave me all sorts of data associated with my target. Since then, they have expanded to email, telephone, and Twitter queries, and we will test each. However, the first step is to obtain a trial API key. You will need to send a request and specifically ask to test their "Enrich" product, and it might help to explain that you are hoping to match email addresses and telephone numbers within your "leads" to personal identifiers. I did this recently and was granted a trial API key which allowed for hundreds of queries without a time restriction. By the time you attempt this, they may offer a different type of trial. Let's take a look at each query and an example result. You would replace "XXXXX" with your own trial API key.

Full Contact Email:

```
curl -X POST https://api.fullcontact.com/v3/person.enrich \
-H 'Authorization: Bearer XXXXX' \
-H "Content-Type: application/json" \
-d '{
  "emails": [
    "lorangb@gmail.com"
  ]
}' | python3 -mjson.tool
```

Full Contact Email Result:

```
   "fullName": "Bart Lorang",
   "ageRange": "40-49",
   "gender": "Male",
   "title": "Co-Founder",
   "organization": "FullContact",
   "website": "http://bartlorang.com/",
   "age": 43
   "education": "University of Denver, Daniels College of Business",
   "degree": "EMBA",
   "year": 2011
   "updated": "2022-10-16"
```

Full Contact Telephone:

```
curl -X POST https://api.fullcontact.com/v3/person.enrich \
-H 'Authorization: Bearer XXXXX' \
-H "Content-Type: application/json" \
-d '{
  "phones": [
    "+12025551212"
  ]
}' | python3 -mjson.tool
```

Full Contact Telephone Result:

```
   "fullName": "John Smith",
   "gender": "Male",
   "title": "Expense Accounting Manager",
   "organization": "Facebook",
   "linkedin": "https://www.linkedin.com/in/john-smith-5934",
       "emails": "john@fb.com,
       "phones": "2025551212",
       "education": "University of Maryland Baltimore County",
```

Full Contact Twitter:

```
curl -X POST https://api.fullcontact.com/v3/person.enrich \
-H 'Authorization: Bearer IMKhL7kBWzwOxosioS84zrlFGH7UeDaP' \
-H "Content-Type: application/json" \
-d '{
  "profiles": [
    {
      "service": "twitter",
      "username": "inteltechniques"
    }
  ]
}' | python3 -mjson.tool
```

Full Contact Twitter Result:

```
   "fullName": "Michael Bazzell",
   "gender": "Male",
   "twitter": "https://twitter.com/IntelTechniques",
   "website": "http://inteltechniques.com/",
   "updated": "2022-07-02"
```

Results from Full Contact will vary. If your target possesses a strong social network presence, expect greater results. If you are researching a burner email or number, expect empty results. I don't use this service often, but it has been valuable when I have. I am still conservatively accessing a free trial obtained in 2021. I find this service best when trying to obtain a phone number as associated with an email address or vice versa. I don't think I would recommend paying for their service purely for OSINT usage.

Breach Directory (breachdirectory.org)

This service allows query of email address, username, password, domain, or hash value against popular known data breaches. We have already discussed these types of services in order to identify email addresses which have been compromised, but this service allows much more customization of our searches. I dedicate an entire section with multiple chapters toward breach data in a moment, but let's focus only on publicly available content within this API for now. First, you will need to obtain a free API key from Beach Directory's Rapid API repository at https://rapidapi.com/rohan-patra/api/breachdirectory. Once you have that, you would replace "XXXXX" in the following examples with your own key. The results for each are using the test data within each query. After the examples, I will offer some OSINT considerations and further investigation.

Breach Directory Email (Identifies Partial Passwords and Full Password Hashes)**:**

```
wget --quiet \
      --method GET \
      --header 'X-RapidAPI-Key: XXXXX' \
      --header 'X-RapidAPI-Host: breachdirectory.p.rapidapi.com' \
      --output-document \
      - 'https://breachdirectory.p.rapidapi.com/?func=auto&term=123@gmail.com'
```

Breach Directory Email Result:

```
   "success": true,
   "has_password": false,
   "sources": "Ledger.com"
   "has_password": true,
      "password": "bart****",
      "sha1": "4a161083c434dd9aa89c0ba91ec37143a47662fd",
      "hash": "jMgujSG0GO4Gtu3gnM2KIfuVMphiLznm",
    "sources":
      "MyHeritage.com",
      "Paxful.com"
    "has_password": true,
      "password": "pala*****",
      "sha1": "2234e9ff8c40dc43cd2ab13c4fc7efca12baf0bb",
      "hash": "sUi6A0IvcpETYOOJnFCAWOmVLI1ifjnz7Q==",
    "sources":
      "MyFitnessPal.com",
      "Dropbox.com",
      "Collection 1",
      "Myspace.com"
```

Breach Directory Username (Identifies Partial Passwords and Full Password Hashes)**:**

```
"Username")
echo "Username: "
read data
wget --quiet \
      --method GET \
      --header 'X-RapidAPI-Key: XXXXX' \
      --header 'X-RapidAPI-Host: breachdirectory.p.rapidapi.com' \
      --output-document \
      - 'https://breachdirectory.p.rapidapi.com/?func=auto&term=badguy143'
```

Breach Directory Username Result:

```
   "success": true,
   "found": 2,
   "has_password": true,
     "password": "pala*****",
     "sha1": "2234e9ff8c40dc43cd2ab13c4fc7efca12baf0bb",
     "hash": "sUi6A0IvcpETYOOJnFCAWOmVLI1ifjnz7Q==",
   "sources": "MyFitnessPal.com"
   "has_password": true,
     "password": "p0ba**",
     "sha1": "f0b261eb28eb14fb9a493c9857eb4d107a1ad0a9",
     "hash": "zSR\/X0xA4lioFowltKCVw+nEIo1yOg==",
   "sources": "Collection 1"
```

Breach Directory Password (Identifies Number of Identical Passwords Found):

```
wget --quiet \
      --method GET \
      --header 'X-RapidAPI-Key: XXXXX' \
      --header 'X-RapidAPI-Host: breachdirectory.p.rapidapi.com' \
      --output-document \
      - 'https://breachdirectory.p.rapidapi.com/?func=password&term=password1'
```

Breach Directory Password Result:

```
   "found": 587
```

Breach Directory Domain:

```
wget --quiet \
      --method GET \
      --header 'X-RapidAPI-Key: XXXXX' \
      --header 'X-RapidAPI-Host: breachdirectory.p.rapidapi.com' \
      --output-document \
      - 'https://breachdirectory.p.rapidapi.com/?func=domain&term=cnn.com'
```

Breach Directory Domain Result:

(This option was not working at the time of writing, but is expected to return.)

Breach Directory Hash:

```
wget --quiet \
      --method GET \
      --header 'X-RapidAPI-Key: XXXXX' \
      --header 'X-RapidAPI-Host: breachdirectory.p.rapidapi.com' \
      --output-document \
      -
'https://breachdirectory.p.rapidapi.com/?func=dehash&term=482C811DA5D5B4BC6D49
7FFA98491E38'
```

Breach Directory Hash Result:

```
   "found": true
   "password": password123
```

Let's dig deeper into these results. In the first example with an email address, you saw the following.

```
      "password": "pala*****",
      "sha1": "2234e9ff8c40dc43cd2ab13c4fc7efca12baf0bb",
```

This service masks the password, but includes a SHA1 hash of the entire password. Hashes are explained in great detail in the next section, but this is a good opportunity to expose the entire password. If you were to enter "2234e9ff8c40dc43cd2ab13c4fc7efca12baf0bb" into the hash tool which was added to your VM during the previous chapters, the result would display the full unredacted password. If we were to enter that password into the "Password" option above, the service would tell us the total number of accounts which include that compromised password. For obvious reasons, I will not display all of that here. We also see the "sources" of this data. We now know that our target possessed accounts at Myspace, MyHeritage, and others. We will dive much further down the data breach rabbit hole in the next section.

Historical Whois Information

In the chapter about domains, I emphasized the value of historical domain registration. People who currently hide their identity behind "whoisguard" protections may not have always done so. I explained a few free options which can display limited historical records. If you want to dive deeper, an API can disclose details unavailable anywhere else. I present options for two services, one of which offers a free trial.

WhoisXMLAPI (whois.whoisxmlapi.com)

In order to use this service, you must first sign up for a free account and confirm your email address. Sign in to the new account and click your profile in the upper-right corner. Select the "My Products" option and make note of your API key. Mine was similar to "at_0vPfsSUdf1ZpiCxc5". You should now have 500 credits toward historical domain registration queries. The command is as follows.

```
curl 'https://whois-history.whoisxmlapi.com/api/v1?apiKey=at_0vPfsSUdf1ZpiCxc5
&domainName=inteltechniques.com&mode=purchase' | python3 -mjson.tool
```

You could also access the URL directly within a browser at the following.

```
https://whois-history.whoisxmlapi.com/api/v1?apiKey=at_0vPfsSUdf1ZpiCxc5
&domainName=inteltechniques.com&mode=purchase
```

This command queries the service; identifies the API key to be charged; provides the domain name to query (inteltechniques.com); and outputs the data in JSON format for easy reading. When conducted against my site, I received several pages worth of details. Most were notifications that I use a whois masking service, and that my true details were unavailable. However, the following was of most interest. In 2013, I did not immediately pay for the whois protection, and WhoisXMLAPI grabbed the following copy of my registration details.

```
"registrantContact": {
    "name": "Michael Bazzell",
    "street": "1700 E Broadway",
    "city": "Alton",
    "state": "Illinois",
    "postalCode": "62002",
    "country": "UNITED STATES",
    "email": "info@inteltechniques.com",
    "telephone": "16184628253",
```

Fortunately, that is not a sensitive physical address, the email does not forward to me, and the telephone number is not mine. However, you would have uncovered the true owner of my site if I were doing something bad. This service provides free credits to numerous additional domain-related services, but I find them to be no more valuable than the free web-based options previously discussed. The historical API is the only option I access.

Whoxy (whoxy.com)

Whoxy allows a free demo at https://www.whoxy.com/whois-history/demo.php, but you will be rate-limited if it detects abuse. They do not offer a free trial of their API, but the fees are minimal. The current price is $2.00 for 400 queries. The following is the URL structure.

https://api.whoxy.com/?key=XXXX&history=inteltechniques.com | python3 -mjson.tool

API Search Tools

We have discussed a lot of options within this chapter, and you will likely find more API services throughout your online investigations. I present two search options which should aid in the execution of the examples presented within this chapter. The first is my online browser-based API Search Tool located on my website at **https://inteltechniques.com/tools/API.html**. It is also replicated within the offline search tools previously explained. It allows you to manually enter your API key and target details for some of the services discussed. It only works with services which allow execution via URL, so it is limited. This may be sufficient for your usage. I confess I never use it. I would rather hard-code my API keys into a script which is always available to me locally on my computer. I explain that next.

API Script

You may feel overwhelmed with these options. I can relate. For many years, I stored bookmarks of all URLs and APIs within my browser, then modified the content of each URL with my new target telephone number or other data. This always felt sloppy and often wasted credits. In 2022, I finally created my own search script which can be executed from within the Linux OSINT machine previously explained. Within that build, locate the api.sh file within your Documents/scripts folder. A shortcut to this file should already be present within your OSINT build from Chapters Four and Five. If using macOS or Windows, use the lessons in Chapter Six of the book to replicate this script for your own needs. Next, you must open the api.sh file and modify the content. You must provide your own API keys issued by any services which you wish to search. Replace any instance of "XXXX" within the script with your own API keys, as previously explained. The following is a sample for the first two entries, displaying placeholder text of "XXXX" for each API key which needs to be entered.

```
$opt1 )
data=$(zenity --entry --title "PDL Email" --text "Email Address")
mkdir ~/Documents/API/
cd ~/Documents/API/
curl
'https://api.peopledatalabs.com/v5/person/enrich?pretty=true&api_key=XXXX&emai
l='$data'' > $data-PDL.txt
open ~/Documents/API/
exit;;
$opt2 )
data=$(zenity --entry --title "PDL Phone" --text "Phone Number")
mkdir ~/Documents/API/
cd ~/Documents/API/
curl
'https://api.peopledatalabs.com/v5/person/enrich?pretty=true&api_key=XXXX&phon
e='$data'' > $data-PDL.txt
open ~/Documents/API/
exit;;
```

You can execute the script, which should now be in your "Activities" menu and application dock labeled as "API Tool". Select the service you desire, then enter the target data when prompted. The results will be saved to a text file which will be placed in the "API" folder within your "Documents" folder. The folder should open upon completion. As a final reminder, you must replace each occurrence of "XXXX" within my script with your actual license keys or credentials if you want to use this technique. There are unlimited API services out there which provide valuable data unavailable within any web browser. When you find the next amazing API, hopefully you now have the ability to add it to your custom scripts.

CHAPTER TWENTY-NINE
ADVANCED LINUX TOOLS

The first section of this book focused heavily on a Linux virtual machine, and I explained numerous applications which assist our investigations. Those programs demanded a single piece of information, which allowed me to present scripts to automate the processes. As an example, the Instagram tools prompt you for a username and then execute the proper commands in order to simplify the process. This chapter has many similarities but also some key differences. It contains some advanced applications which would be impossible to automate with simple scripts and applications which require the lessons explained within previous chapters. Some of these programs require heavy user input with features which must be tackled within Terminal while others present automation similar to the options presented within the first section of this book.

Before we begin, there are two very important details. First, I will not display the full code from within the scripts and shortcuts of each program. You can easily navigate to the "scripts" and "shortcuts" folders within the "vm-files" archive which you previously downloaded if you want to view the entire code. Displaying the lengthy text within this chapter might be considered redundant and a waste of paper. I only focus on the benefits of each tool here.

Next, every application mentioned within this chapter is already present within your VM which possesses an automated script to help you along the usage. If you completed all of the steps within Chapter Five and created a custom OSINT virtual machine, you are ready to start using each of these applications. I encourage you to boot to a CLONE of your "OSINT Original" VM and test the lessons as you progress through this chapter. Your previous work will pay off here and allow you to jump right in. I still present the manual steps for installation, configuration, and usage for each application in order for you to understand the process. None of these steps need to be replicated if you simply want to launch each program within your custom VM.

Some readers skip this chapter until they become more comfortable within Linux. However, I believe those with any skill level can replicate all tutorials. Most applications will feel similar to the automation previously discussed in Section One. Let's start with the automated tasks.

Internet Archive Tool

I previously explained the power of the Internet Archive while searching for online content which has since been removed. Browsing a target domain on the Wayback Machine website can be very fruitful, but automating the collection of data can be more beneficial. The "Archive Tool" script titled "archives.sh" in the custom OSINT VM we made in Chapters Four and Five takes advantage of a Python script called "waybackpy", which was previously installed. Launching the script presents a single domain entry window which accepts a domain or specific URL. Executing the script conducts the following tasks, using pretendradio.org as a target.

Make a directory in the Documents folder for data and enter it:

```
mkdir ~/Documents/waybackpy
mkdir ~/Documents/waybackpy/pretendradio.org
cd ~/Documents/waybackpy/pretendradio.org
```

Download all known URLs indexed by Internet Archive into a text file:

```
waybackpy --url "https://pretendradio.org" --known_urls
```

Download the oldest known archive URL into a text file:

```
waybackpy --url "https://pretendradio.org" --oldest >> pretendradio.txt
```

Append the file with the newest archive URL:

```
waybackpy --url "https://pretendradio.org" --newest >> pretendradio.txt
```

Append the file with URLs from the past ten years:

```
waybackpy --url "https://pretendradio.org" --near --year 2010 >>
pretendradio.txt
waybackpy --url "https://pretendradio.org" --near --year 2011 >>
pretendradio.txt
waybackpy --url "https://pretendradio.org" --near --year 2012 >>
pretendradio.txt
waybackpy --url "https://pretendradio.org" --near --year 2013 >>
pretendradio.txt
waybackpy --url "https://pretendradio.org" --near --year 2014 >>
pretendradio.txt
waybackpy --url "https://pretendradio.org" --near --year 2015 >>
pretendradio.txt
waybackpy --url "https://pretendradio.org" --near --year 2016 >>
pretendradio.txt
waybackpy --url "https://pretendradio.org" --near --year 2017 >>
pretendradio.txt
waybackpy --url "https://pretendradio.org" --near --year 2018 >>
pretendradio.txt
waybackpy --url "https://pretendradio.org" --near --year 2019 >>
pretendradio.txt
waybackpy --url "https://pretendradio.org" --near --year 2020 >>
pretendradio.txt
```

Remove duplicates and sort by date:

```
sort -u -i pretendradio.txt -o pretendradio.sorted.txt
```

Generate screen captures of all unique links with only one thread (slower):

```
webscreenshot -r chrome -i pretendradio.sorted.txt -w 1
```

Download source code of the oldest and newest archives:

```
waybackpy --url "https://pretendradio.org" -o > oldest.html
waybackpy --url "https://pretendradio.org" -n > newest.html
```

The result is a "pretendradio.org" folder within Documents/waybackpy which includes a text file of over 500 URLs (with duplicates) of potential evidence on the target domain; a text file of five known archives of the target website; screen captures of all five archives; and source code of the newest and oldest archive. Figure 29.01 displays a result which identifies the way the target website appeared in April and October of 2017. Imagine you have a target URL which has hundreds of archives throughout the past decade. This is a quick way to see annual changes to the site and document the evidence. When the tool completes, you will be presented with the folder containing all of the data.

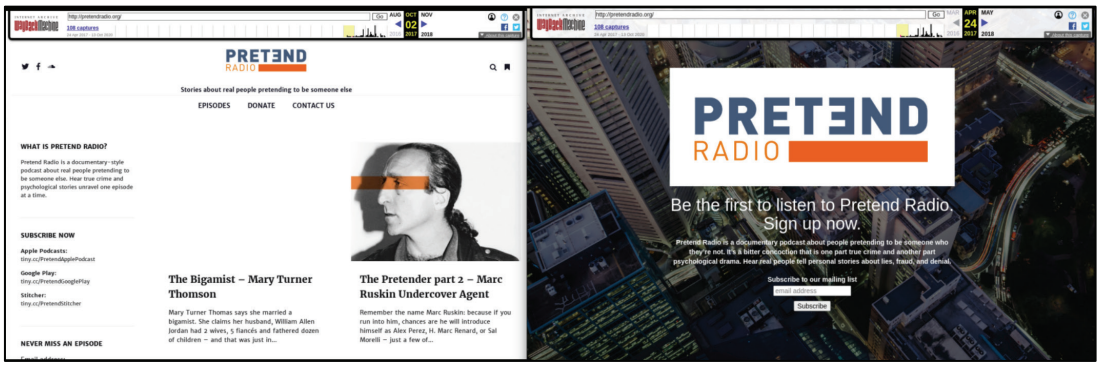


Figure 29.01: Screen captures from the automated Internet Archive Tool.

Now that we have screen captures of these pages, we should also extract the text. The automated script included with your custom VM conducts the following at the end of the cycle.

```
wget -i $url.sorted.txt
```

This places several files within the "waybackpy" folder within your "Documents" folder. In this example, they included the following.

```
index.html
index.html.1
index.html.2
index.html.3
```

Each of these is an archived home page of the target website from a different date. Double-clicking them will load the pages within a web browser which should display the target site similar to the way it previously appeared. However, a keyword search can also be beneficial. You can use the embedded search function within the Files application which is presented after successful completion of the script. Figure 29.02 displays my search of these files for the word "prison". The results identify files of interest which I need to further investigate.

In summary, this tool compiles a list of URLs related to the target website on The Internet Archive; creates screen captures of those pages; and extracts the text in order to quickly search.

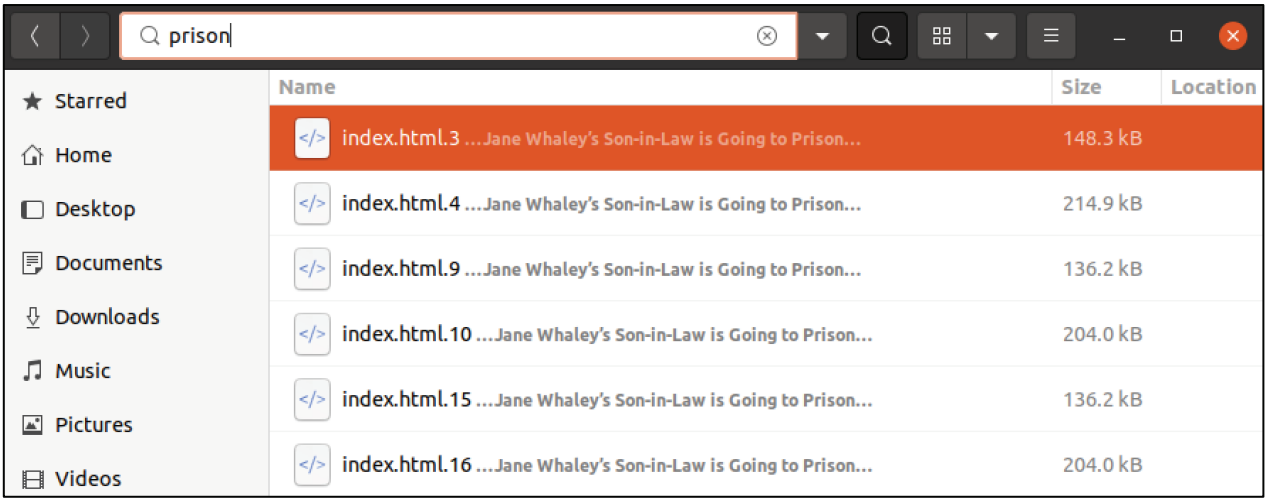


Figure 29.02: Search results from a keyword search within archived web pages.

Change Detection (github.com/dgtlmoon/changedetection.io)

In the chapter about domains, I explained several online services which would monitor a URL for changes. These are beneficial, but there are some scenarios where they might be inappropriate. You might have a sensitive URL which you do not want to share with an online service, or a site with very minor changes which is not being picked up by the free services. This is where Change Detection can help. It is locally installed and available only to you. The following configures the application.

```
mkdir ~/Downloads/Programs/changedetection
cd ~/Downloads/Programs/changedetection
python3 -m venv changedetectionEnvironment
source changedetectionEnvironment/bin/activate
sudo pip install changedetection.io
deactivate
```

We can now test with the following entries into Terminal.

```
mkdir ~/Documents/ChangeDetection
changedetection.io -d ~/Documents/ChangeDetection -p 5000 & firefox
http://127.0.0.1:5000
```

The first command makes sure we have a directory created for results and the second command launches the server and Firefox simultaneously. Figure 29.03 displays an example. The first option was created for my home page. I can click the "Recheck" button to fetch a new live copy. Clicking the "Diff" button shows me any changes. Figure 29.04 identifies two paragraphs which were modified between visits. This tool provides the same service as the online options, if not better, but gives you full control within your Linux machine.

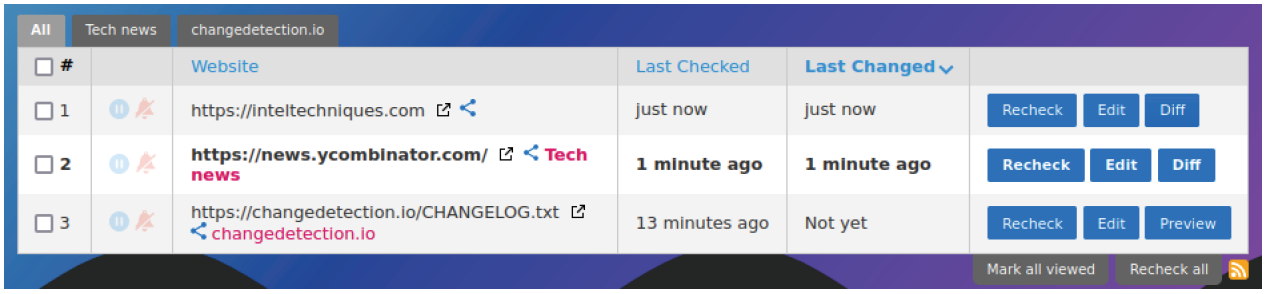


Figure 29.03: Change Detection in action.

```
      Welcome to the new home for both sites. With simpler navigation and a cleaner interface, you should be able
to easily browse through our services and resources. Consider checking out our blog and podcast from the menu
above, or visit our resources page for our latest projects, such as the new online search tools. Browse through our
services page to learn about the various privacy remedies we offer. Join our online video training and become a
certified Open Source Intelligence Professional (OSIP) around your schedule! Finally, we offer updated books about
OSINT and Privacy in our book store.
      TEST HERE Welcome to the new home for both sites. With simpler navigation and a cleaner interface, you
should be able to easily browse through our services and resources. Consider checking out our blog and podcast from
the menu above, or visit our resources page for our latest projects, such as the new online search tools. Browse
through our services page to learn about the various privacy remedies we offer. Join our online video training and
become a certified Open Source Intelligence Professional (OSIP) around your schedule! Finally, we offer updated
books about OSINT and Privacy in our book store.
```

Figure 29.04: Change Detection modifications to a target URL.

Archive Box (github.com/ArchiveBox/ArchiveBox)

I previously explained how we could use online services to document the current state of a website; see previous versions through archived sources; and be notified of changes when they occur. In 2022, I began using Archive Box as an all-in-one solution. We can install and initialize the application with the following, which you may have already accomplished in Chapter Five.

```
mkdir ~/Downloads/Programs/archivebox
cd ~/Downloads/Programs/archivebox
python3 -m venv archiveboxEnvironment
source archiveboxEnvironment/bin/activate
sudo pip install archivebox
deactivate
mkdir ~/Documents/archivebox
cd ~/Documents/archivebox
archivebox init
```

Once installed, change into our directory; add our target website; launch the server; and open the database within our browser. The "Archive Tool" already within your Linux virtual machine automates this for you, but we should take a look at the manual steps. Note that you must enter the full URL, including "https://".

```
cd ~/Documents/archivebox
archivebox add 'https://notla.com'
archivebox server 0.0.0.0:8000
firefox http://0.0.0.0:8000/
```

Figure 29.05 displays results as they are being fetched. Archive Box captured the target with SingleFile through Chrome; generated a PDF and screenshot; performed a WGET of the live page; extracted the page from Archive.org; fetched all HTML code; and downloaded any media files. This is an impressive tool. Combined with the automated script in your Linux VM, this service should be used often. Knowing the manual approach is important, but automated tools allow us to work faster and more efficiently.

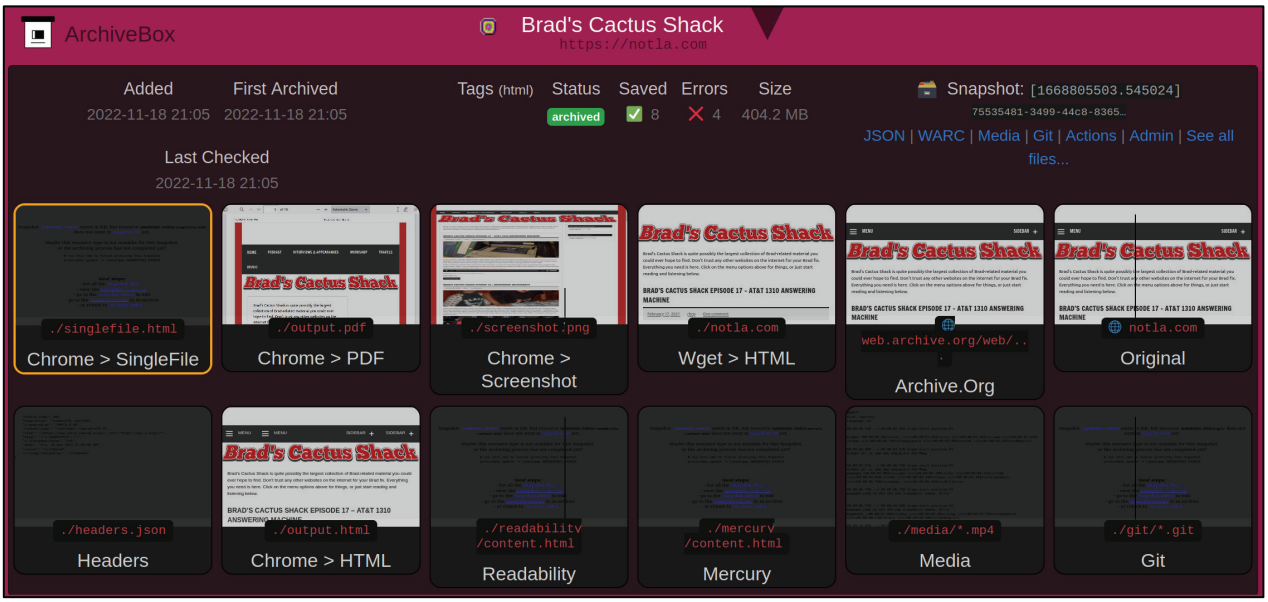


Figure 29.05: Archive Box.

Spiderfoot (https://github.com/smicallef/spiderfoot.git)

This program introduces more valuable utilities than any other single resource within this book. This will take some effort to install and configure, but the benefits justify the work. I installed the application into the custom Linux VM with the following steps.

```
cd ~/Downloads/Programs
git clone https://github.com/smicallef/spiderfoot.git
cd spiderfoot
python3 -m venv spiderfootEnvironment
source spiderfootEnvironment/bin/activate
sudo pip install -r requirements.txt
deactivate
```

If we are inside the Spiderfoot directory (~/Downloads/Programs/spiderfoot), we can launch the Spiderfoot service with the following command.

```
python3 ./sf.py -l 127.0.0.1:5001
```

If executing manually, you would need to launch Firefox and connect to http://127.0.0.1:5001. We can replicate this from Terminal with the following.

```
firefox  http://127.0.0.1:5001
```

I have already prepared a script titled "Spiderfoot.sh" which can be launched from your Dock or Applications menu within your custom OSINT VM. Let's take a look at the interface and conduct an example query. After launching the Spiderfoot program within Firefox, click the "New Scan" option. Provide a name for your session (IntelTechniques) and a domain (inteltechniques.com). Choose your desired "Use case" and click "Run scan".

I chose "All" in order to test the features, but this can be intrusive toward your target site. Choose the level of access appropriate for your investigation. The scan will launch and may take a while to complete, possibly hours. The amount of data acquired will be substantial, and I will only focus on a few areas of interest. The default screen displays the current progress and a log file. The "Browse" button in the upper left allows you to start peering into the data found about your target. Below are the sections of interest to my own site and a description of the results displayed.

- Account on external site (Four online profiles connected to my brand)
- Human Name (Identification of my full name and three associates)
- Leak site content (56 Pastebin files referencing my domain)
- Similar domain (Two domains with similar spelling, attempting to hijack traffic)
- Web technology (Pages on my site which rely on PHP, and my version)

The "Graph" button displayed a detailed chart of connections from my domain to external sources. I cannot overstate that I am only presenting a handful of interesting nuggets. This application scours a domain, IP address, or email address for hundreds of data points which may provide value. Clicking the "Scans" button provides options to stop, re-run, or delete a scan result. It also provides a status summary of each current scan, and you can execute numerous scans simultaneously.

Recon-ng

Recon-ng is a full-featured web reconnaissance framework written in Python. Complete with independent modules, database interaction, built-in convenience functions, interactive help, and command completion, Recon-ng provides a powerful environment in which OSINT research can be conducted quickly and thoroughly. This utility provides automation to many of the redundant tasks that OSINT examiners find themselves

performing on a daily basis. I offer a warning before proceeding. This is a technically complicated portion of this book. Please don't let that scare you off, we will approach each step slowly. First, we need to install Recon-ng into our OSINT Original virtual machine. Type the following into Terminal.

```
cd ~/Downloads/Programs
git clone https://github.com/lanmaster53/recon-ng.git
cd recon-ng
python3 -m venv recon-ngEnvironment
source recon-ngEnvironment/bin/activate
sudo pip install -r REQUIREMENTS
deactivate
```

This should install Recon-ng, and typing `./recon-ng` from within this directory launches the application. This screen will display the current version, and I am writing this based on version 5.1.1. You can also launch the "Recon-NG" icon in your Dock which eliminates the need to navigate to ~/Downloads/Programs/recon-ng every time you want to use the application. The following command would be used to update your version of Recon-ng, which is already included in the "updates.sh" script previously mentioned.

```
cd ~/Downloads/Programs/recon-ng
git pull https://github.com/lanmaster53/recon-ng.git
```

Recon-ng does not possess many online tutorials. The guides that I have found are mostly an index of commands with little explanation. Instead of trying to summarize how the program functions, I will walk you through actual usage and explain as we go. I will start with the basics and then conduct numerous actual searches. In lieu of screen captures, I will include all text input and output in **`9 pt Terminal font`**. Upon executing Recon-ng, you will be notified that no modules are installed. This is normal, and we will add them as we need them. At this prompt, let's begin with the help command. Typing **`help`** reveals the following commands and explanations.

```
back            Exits the current context
dashboard       Displays a summary of activity
db              Interfaces with the workspace's database
exit            Exits the framework
help            Displays this menu
index           Creates a module index (dev only)
keys            Manages third party resource credentials
marketplace     Interfaces with the module marketplace
modules         Interfaces with installed modules
options         Manages the current context options
pdb             Starts a Python Debugger session (dev only)
script          Records and executes command scripts
shell           Executes shell commands
show            Shows various framework items
snapshots       Manages workspace snapshots
spool           Spools output to a file
workspaces      Manages workspaces
```

Typing **`marketplace search`** will reveal the current functions available. Think of the marketplace similar to a list of utilities within Recon-ng, and each option as a "resource". Just like Bing is a website resource that we can use through a web browser, "bing_domain_web" is a specific resource that we can use in Recon-ng. At any time, you can type **`marketplace info`** into Recon-ng to receive details about a specific item. As an example, typing **`marketplace info virustotal`** displays the following description.

```
path          | recon/netblocks-hosts/virustotal
name          | Virustotal domains extractor
author        | USSC (thanks @jevalenciap)
version       | 1.0
last_updated  | 2019-06-24
description   | Harvests domains from the Virustotal by using the report API.
```

```
required_keys | ['virustotal_api']
dependencies  | []
files         | []
status        | not installed
```

This provides the detailed description, and whether the utility requires an API key or other dependencies. It also confirms we have not installed the module. We will execute this option later in the chapter. For now, we must set up our first investigation. Before we can conduct any research within this program, we must create a workspace. A workspace is a container that will isolate your work from one investigation to another. Think of a workspace as a case file. You may have a stack of cases on your desk, each with its own folder. All of your work on a case stays within the folder associated. Workspaces are similar. You should create a new workspace for each investigation. They can be deleted later or preserved for additional work. You can type **`workspaces list`** at any time to see the currently used workspaces. For now, we will create a new workspace titled OSINT by executing the command of **`workspaces create OSINT`**.

After creation, you will automatically begin using the new workspace. If you have created more than one workspace, such as one titled OSINT2, you can switch to it by typing **`workspaces load OSINT2`**. You might have a workspace for every target suspect or a single workspace for an entire case. Each situation will be unique. Now that you have a space created, we can begin. Let's start with a very simple yet powerful query, using the Profiler module. First, we must install the module with the following command within Recon-ng.

```
marketplace install profiler
```

The module is now installed, but is not loaded. The following loads the module.

```
modules load profiler
```

Now that the module is loaded, we can add any input desired. Since this module queries usernames, we will add our target of "inteltechniques" with the following command. Note that SOURCE is uppercase, which is required.

```
options set SOURCE inteltechniques
```

We should test our input with the following command.

```
input
```

The response should now be the following.

```
+-----------------+
|  Module Inputs  |
+-----------------+
| inteltechniques |
+-----------------+
```

Finally, we can launch the module with the following command.

```
run
```

This script should query the username of inteltechniques against numerous online services. It does not present any results when complete, but did locate and store valuable data. To view the results, type the following.

```
show profiles
```

The results appear similar to the following.

```
-------------------------------------------------------------------------------------------+
1     | inteltechniques | Gravatar | http://en.gravatar.com/profiles/inteltechniques | images   |
2     | inteltechniques | reddit   | https://www.reddit.com/user/inteltechniques     | news     |
3     | inteltechniques | Twitter  | https://twitter.com/inteltechniques             | social   |
-------------------------------------------------------------------------------------------+
```

In just a few seconds, we queried dozens of online services and immediately received only the three which contained the presence of our target username. This demonstrates the ability to save a substantial amount of time by using Recon-ng. If you were tasked to locate online profiles of ten suspects, this could be completed in a few minutes. Let's repeat the process, but with another username, with the following commands.

```
options set SOURCE humanhacker
run
show profiles
```

The following result displays the additional online profiles collected during this second query. Recon-ng will continue to store target data as you receive it. This is one of the more powerful features of the application.

```
-------------------------------------------------------------------------------------------+
1  | inteltechniques | Gravatar  | http://en.gravatar.com/profiles/inteltechniques   | images     |
2  | inteltechniques | reddit    | https://www.reddit.com/user/inteltechniques       | news       |
3  | inteltechniques | Twitter   | https://twitter.com/inteltechniques               | social     |
4  | humanhacker     | Blogspot  | http://humanhacker.blogspot.com                   | blog       |
5  | humanhacker     | Disqus    | https://disqus.com/by/humanhacker/                | discussion |
6  | humanhacker     | Flipboard | https://flipboard.com/@humanhacker                | tech       |
7  | humanhacker     | GitHub    | https://api.github.com/users/humanhacker          | coding     |
8  | humanhacker     | Instagram | https://www.instagram.com/humanhacker/            | social     |
9  | humanhacker     | Kongregate| http://www.kongregate.com/accounts/humanhacker    | gaming     |
10 | humanhacker     | Kik       | https://kik.me/humanhacker                        | social     |
11 | humanhacker     | Medium    | https://medium.com/@humanhacker/latest            | news       |
12 | humanhacker     | Technet   | https://social.technet.microsoft.com/humanhacker/ | tech       |
13 | humanhacker     | Minecraft | https://namemc.com/name/humanhacker               | gaming     |
14 | humanhacker     | Pornhub   | https://www.pornhub.com/users/humanhacker         | XXX PORN   |
15 | humanhacker     | scratch   | https://scratch.mit.edu/users/humanhacker/        | coding     |
16 | humanhacker     | reddit    | https://www.reddit.com/user/humanhacker           | news       |
17 | humanhacker     | Twitch.tv | https://passport.twitch.tv/usernames/humanhacker  | gaming     |
18 | humanhacker     | Twitter   | https://twitter.com/humanhacker                   | social     |
19 | humanhacker     | Xbox      | https://www.xboxgamertag.com/search/humanhacker/  | gaming     |
-------------------------------------------------------------------------------------------+
```

Let's conduct another example within a different module. First, we must leave our current module by typing **back**. This returns us to our workspace. Next, install four additional modules with the following commands.

```
marketplace install bing_domain_web
marketplace install google_site_web
marketplace install brute_suffix
marketplace install pgp_search
```

We will use these in order and target the website cnn.com. First, we will load the bing_domain_web option with the command of **modules load bing_domain_web**. Next, we will set our source with **options set SOURCE cnn.com** and execute the script with **run**. This command queries the Bing search engine for hosts connected to the domain cnn.com. The result identified over 70 unique hosts, including the following.

```
[*] [host] internationaldesk.blogs.cnn.com (<blank>)
[*] [host] crossfire.blogs.cnn.com (<blank>)
[*] [host] reliablesources.blogs.cnn.com (<blank>)
```

```
[*] [host] lightyears.blogs.cnn.com (<blank>)
[*] [host] commercial.cnn.com (<blank>)
[*] [host] collection.cnn.com (<blank>)
```

We can replicate this type of search on Google to make sure we are not missing any hosts that could be valuable by typing **back**, then **modules load google_site_web**, then **options set SOURCE cnn.com**, and finally **run**. This notifies us **38 total (15 new) hosts found**, which indicates that Bing found more hosts than Google, and Google found 15 hosts that we did not have in our collection from Bing. Since Recon-ng can parse out duplicates, we should have a list of unique hosts with a combined effort from both Google and Bing. Typing **show hosts** will display all of them. Below is a small portion.

```
news.blogs.cnn.com
m.cnn.com
buzz.money.cnn.com
thechart.blogs.cnn.com
globalpublicsquare.blogs.cnn.com
tech.fortune.cnn.com
```

Next, let's type **back** to leave the current module and then **modules load brute_suffix** to load our next demo. Since there is no domain set as our source for this module, we will add one with **options set SOURCE social-engineer.org**. There are many top-level domains (TLDs) aside from .com and .org. Executing **run** will scour the various TLDs such as .net, .tv, and others. After completion, typing **show domains** will display our updated set of target addresses ready for further searching. In this example, I was notified that additional domains were located, including the following.

```
social-engineer.be          social-engineer.dev
social-engineer.ch          social-engineer.info
social-engineer.com         social-engineer.me
social-engineer.de          social-engineer.us
```

These are all new leads that should be analyzed later. We could now repeat our previous module execution of **bing_domain_web** and **google_site_web** to likely grow our list of hosts substantially. This is a good time to pause and consider what is happening here. As we find data, Recon-ng stores it within our workspace. Every time we conduct a new search, or repeat a previous search, all of the new data is appended. This prevents us from documenting everything that we locate because Recon-ng is keeping good notes for us. This can allow us to collect an amount of data otherwise impossible to manage manually. Let's move on to individual contacts.

Typing **show contacts** will display any contacts stored within the current workspace. You likely do not have any, so let's add some. First, type **back** to make sure you are out of the previous module. Next, load another module with **modules load pgp_search**. This will scan all of the stored domains that we have located and search for any email addresses associated with public PGP keys within those domains. We have not set a source for this module, but you likely already have some ready for you. In a previous example, you searched social-engineer.org within other top-level domains and received numerous results. If you type **input** within this module, you should see those same domains listed. This is because Recon-ng is constantly storing found data and making it available for future use. If we type **run**, this list will be searched, but no results will be found. Note that this list does not possess our target domain of social-engineer.org, and only the additional names previously found. Therefore, you may wish to remove these sources, and then add a fresh new source, with the following commands.

```
options unset SOURCE
options set SOURCE cnn.com
```

Typing **run** and striking enter executes the process, while submitting **show contacts** afterward displays the results. The following is the partial output with new email addresses identified. Each of these addresses are now stored in your workspace, ready for the next round of research.

```
| 2     | barsuk      |             |             | barsuk@cnn.com
| 3     | Tristan     |             | Helmich     | tristan.helmich@cnn.com
| 4     | Paul        | P           | Murphy      | paul.p.murphy@cnn.com
```

Let's reflect on how this can be beneficial. Assume that you are investigating numerous websites. Recon-ng provides dozens of utilities which automate queries and provides immediate information. Magnify this by tens or hundreds of domains, profiles, or other target data, and you have an easy way to replicate several hours of work. In another scenario, you are investigating a list of potential email addresses connected to a case. Entering these into Recon-ng allows you to execute your searches across all accounts. The effort to check one address is the same to check thousands. This impressive capability is only a small fraction of what can be done with this application.

This seems like a good time to back away, create a report, and start a new set of actions. The following commands will back out of our current module; install the reporting feature; instruct Recon-ng that we want to use the reporting tool; mandate a graphical html (web) template be used; set the "Customer" as IntelTechniques; set the "Creator" as M.Bazzell; and execute the process.

```
back
marketplace install html
modules load html
options set CUSTOMER IntelTechniques
options set CREATOR M.Bazzell
run
```

Note the output after the final command. It identifies that the report is complete, and provides the storage location. Since I am running Recon-ng from my OSINT virtual machine, the default location is ~/.recon-ng/workspaces/OSINT/results.html. Therefore, I can open the home folder on my desktop; double-click the ".recon-ng" folder; double-click the "workspaces" folder; double-click the "OSINT" folder; and then open the "results" file. Please note you must have "Show Hidden Files" option enabled from within the preferences menu of the Files application. Note that the Domains, Hosts, and Contacts sections are not expanded, but contain a lot of information. At the bottom of this file, the "Created by", date, and time clearly identify these report details.

Hopefully this demonstration explained the usage of Recon-ng. Executing **exit** in the window closes everything, but removes nothing. Before our next example, let's delete our previous work and start fresh. Note that deleting a workspace removes all associated data and reports. Make sure you have exported your evidence if needed. First, relaunch Recon-ng. The following commands display the current workspaces; delete the OSINT workspace; and create a new workspace titled location.

```
workspaces list
workspaces remove OSINT
workspaces create location
```

This chapter explains only a small portion of the capabilities of Recon-ng. Please consider revisiting the modules listed at the beginning and experiment with the execution of each. Overall, it would be very difficult to break the application, and any errors received are harmless. You will receive best results by requesting API keys from the services which require them. The "Info" screen of each Recon-ng module displays any requirements within the "Required Keys" field. Many API keys are free and open new possibilities. Overall, an entire book could be written about this application alone. The goal of this section was simply to familiarize you with the program and demonstrate the power of automated queries.

If you would like more information about Recon-ng, please visit the official Github page at https://github.com/lanmaster53/recon-ng. From there, you can join a dedicated group in order to participate in discussions about errors, features, and overall usage. While Recon-ng has not been updated in some time, and many believe the project has been abandoned, the tools still perform well for specific tasks.

SECTION III
LEAKS, BREACHES, LOGS, & RANSOMWARE

The previous edition of this book had a single chapter devoted to data breaches and leaks. It was possibly the most controversial chapter, and contained the content which was most discussed and debated. Every day, I am asked for further clarification, tutorials, and resources surrounding this type of data. This new section, which contains four chapters dedicated to these topics, attempts to expand on the original teachings. This section is probably the biggest change since the previous edition, and hopefully the most anticipated.

The techniques that you will read about in this section are for educational use only. Many, if not all, of the methods described here could violate organizational policy if executed. While the section will only discuss publicly available data, possession could violate your security clearance or be determined illegal by state or federal law. Distribution (not necessarily possession) of the types of content that will be discussed here may be illegal in most cases. However, I will explain ways that you can apply these practices in a legal way, and keep yourself employed. Overall, please do not take any action from this instruction unless you have verified with your organization's legal counsel or supervisory personnel that you have the authority to do so. I am not a lawyer.

In previous chapters, I discussed the online services Have I Been Pwned (HIBP) and Dehashed as two resources for compromised email search. These services possess huge databases of publicly available content which were stolen from the host companies and distributed over the internet. In order to emphasize the concern, I will repeat that **this data was originally stolen by criminals**. When you search an email address on these services and are informed that it was compromised within the LinkedIn data breach, this means that a partial copy of this stolen data resides within these services. HIBP and others are often applauded as a great resource to monitor your own accounts for any reported compromises. Well, what if we created our own unredacted collection of this data? Is there additional value? I believe so.

This is where things get tricky. While you can find copies of thousands of stolen databases all over the internet, what are the legalities of downloading and possessing the data? First, let me say that I am not an attorney and I offer no legal advice. I have spoken with many attorneys and prosecutors about this, and the feedback was similar from each. If the data is publicly available, possession alone would likely be legal. This is similar to viewing an email stolen from Hillary Clinton posted on WikiLeaks or an internal document stolen from Google posted on a blog. If you do not violate the laws and policies applicable in your city, county, state, or country when you view this publicly available, yet stolen, data, then you could likely get away with viewing stolen credentials existing in the various database leaks online.

What matters most is that you never take any illegal action with the data that you possess. In a moment, I will explain how to access valid email addresses and passwords of billions of accounts. Using this data as an investigation technique is one extreme, but attempting to use this data to access someone's account is another. **There is no situation where casually gaining access to a target's account is acceptable**, unless you have a valid search warrant or court order to do so. Since many of you are in law enforcement, this chapter may identify new strategies for you when you have the legal authority to access an account. We will start with some very basic legal "leaked" data in just a moment.

Before proceeding, we should discuss some definitions for these four types of data and legitimate uses for each. We can then tackle the hardware and software requirements for the chapters within this section. First, let's define the scope. There are many lengthy definitions and explanations of this type of data, but let's keep things simple. For the purposes of this section, the following apply.

Leaks: This type of data is accidentally released, or "leaked", through public online resources. It could be public voter data which was legally acquired and then shared online; a marketing database of public information which was archived through a third-party site; a misconfigured database which is publicly leaking all details if you know

where to look; or a plethora of other possibilities. The common theme is that no one intentionally compromised a secure server and stole anything. Of the four types of data, this is the most likely to be legally accessed.

Breaches: This type of data takes us into the criminal world. Breaches occur when a criminal compromises a source of data illegally and copies the content. Examples include the Twitter and LinkedIn breaches where people's email addresses and passwords were stolen and published online. There are now tens of thousands of data breaches floating around, which have introduced billions of plain-text passwords for the majority of people online. While accessing or downloading this publicly-available data might not get you into legal trouble, further publishing it might. Also, your organization's policies might prevent you from this access. Possession and research of this data is extremely common today, even by us law-abiding internet citizens. More warnings are to come.

Logs: This data, sometimes referred to as "Stealer Logs" or "Password Logs" is illegally obtained from infected machines. A criminal tricks a victim into installing a virus which extracts data and sends it to an accessible third-party server. This often includes stored passwords, session cookies, documents, photos, and other sensitive information. This may sound like a rarity, but tens of thousands of victim's logs are uploaded every day. This type of data can expose the passwords of people who are not within any known data breaches, and the content is much more likely to be accurate.

Ransomware: This type of data is also illegally acquired. Criminals install a virus within a company's network; infect many machines; make a copy of all data; encrypt all files on the network; and hold the decrypted copies hostage. They then offer to decrypt all data for a ransom, and threaten to publish the stolen content if payment is not received. Every day, new private company data is published online for anyone to see. This often includes passwords, sensitive documents, and copies of passports.

There are many reasons for interest in leaks, breaches, logs, and ransomware. I always split them into the sides of "offense" and "defense". Let's discuss a few of both, beginning with offense.

Identify Home Addresses: Some people do a very good job of keeping their names and home addresses off of the internet. You may strike out with people search websites. However, they often slip-up when ordering products to their home or registering to vote. Breach and leak data will often disclose true home addresses when no other online resource was successful.

Identify Alias Account Holders: In a later chapter, I will explain how I uncover the true owners of many "burner" email addresses due to common recycling of passwords. When my suspect uses a junk email account, and it appears within a data breach, I may see his password is "badguy432!@". When I discover an additional email address which had a password of "badguy432%$", I call that a lead. A surprising number of suspects will use the same password across their real account and those used for criminal purposes. Once their information appears within multiple data breaches, we can put together the pieces. I will provide many examples later.

Associate Social Network Accounts to People: Every day, my company uses breach data to connect a social network account to a real person. When a service has a data breach, the otherwise hidden email addresses and cellular telephone numbers are often disclosed along with the username for the product. This can immediately uncover the account holder's true identity. When used in conjunction with the previous two examples, we often solve entire cases from breach data alone.

The following are only a few defense considerations.

Identify Stolen Assets: If you are employed by a company which has suffered a data breach, you may want to track down the stolen data. This helps understand the magnitude of the breach better than any claims or demands from the criminal. I believe every victim of a data breach should investigate the incident and obtain the data being shared publicly. This is one of our most requested services by clients.

Aid Disclosure Notifications: When a company is hit with a cyber-attack, there are many laws and regulations which require notification to those impacted by the stolen data. Knowing the exact data being shared within criminal groups can help identify the proper disclosure requirements for impacted victims.

Defend Clients From Future Attacks: I rely on our daily update of leaks, breaches, logs, and ransomware in order to protect my clients. Every week, we are able to notify a potential victim of newly exposed information which could be devastating to them if abused. In many cases, we can intercept this data before the criminals begin to use it. We can then help the client block access before bad things happen.

These are only a few benefits to this type of data. I query our leaks, breaches, logs, and ransomware data every day. My staff uses them to quickly identify new leads and to protect our clients. I cannot overstate the value of this data. Next, let's have a conversation about hardware. If you begin acquiring this type of data, it can become overwhelming very quickly. Every day, our systems ingest anywhere from 20 to 500 GB of data. Storing and processing this amount of data can be both time consuming and resource intensive. Even if you stick to public leaks, the data can fill your drives quickly. Because of this, there are many hardware considerations.

First, your selection of computer is vital. This may sound contradictory to previous chapters, but I no longer recommend a virtual machine for this type of work. Downloading large files within a VM will be slightly slower than on a host machine. Decompressing large files within a VM will be much slower, and VM disk space will be depleted quickly. You could connect an external drive to the VM for storage, but the data transfer speed will be a huge bottleneck to the overall efficiency of data queries.

Unless you are simply dabbling in this area, I would stay away from VMs while trying to conduct any extensive breach work. However, I also never recommend using your primary personal or investigations host computer for this type of work. I would never download breach data onto the same machine I use for personal communications, banking, or other private tasks. I would also never consider processing breach data within the host operating system of my primary OSINT investigations computer. This is because the type of data we will be discussing is full of bad players, malicious files, viruses, and other shady concerns. I would never jeopardize the host operating systems of my personal or investigations machines to the risks associated with stolen data.

What do I do? I possess a machine solely for breaches, leaks, logs, and ransomware. It is a Linux machine with minimal applications and blazing fast hardware. When I am not actively downloading data, it is offline. I prefer a desktop without Wi-Fi connectivity so I can disconnect ethernet whenever I want to be sure it is offline. I insist on a minimum of a NVMe PCIe 4 drive which is inserted into a slot directly on the motherboard. I am currently using a System 76 Thelio desktop with an 8 TB NVMe PCIe 4 drive which can read data at over 7000 MB/s, which is faster than the current high-end Macbook Pro drive and three times faster than most SATA SSD drives. It has 32 GB of RAM, which helps with fast data processing. The desktop also possesses 20 TB of slower 2.5" 5 TB drives for long-term storage of data and only supplies integrated graphics.

This small desktop has 28 TB of storage; is faster than any laptop I have tested; and is quite affordable compared to Macs. I know that the data I download is toxic, but I don't care. If I encounter a virus included within some stealer log data, I know it is likely targeted toward the Windows operating system and would have no impact on my Linux machine. If I encountered a rare Linux threat, I know that there is no personal information on this machine which could expose me. If the machine becomes infected, it is no big deal to wipe it out and reinstall Linux. I backup all of my data to an external 20 TB drive.

Is this all overkill? Possibly. It depends on how much you rely on this type of data. Could you get by with just using more traditional hardware for this task? Absolutely. It will only take more time to download, decompress, process, clean, store, and query the data every day. If you begin to rely on this data often, you might find yourself looking at faster machines in order to make the process more tolerable. There is no harm using older equipment in order to understand the strategies presented within this section. The limit to your resources will become apparent as soon as you start dealing with large data sets.

As an example, I downloaded 250 GB of compressed stealer logs to a MacBook Pro laptop with a 2100 MB/s internal SSD. The decompression of the data took over an hour. Parsing the passwords from the text files took another four hours. I replicated the exact same process on my Linux desktop. The decompression took seven minutes, and the extraction of passwords took twelve minutes. I deal with enough data to justify the expense of a dedicated machine for this purpose. When we start talking about stealer logs, the vast number of files alone can be a bottleneck on most machines. It is common to extract over two million very small files within a single compressed archive.

I suspect many readers are not ready to commit to new hardware for these tasks. Therefore, I want to offer some considerations which cost less. If you do not have enough internal drive space to process and store this data, consider a proper external drive. I would never consider a mechanical or flash drive for this process. At a bare minimum, I would rely on portable SSD drives made by Sandisk.

I have had great success with the Sandisk Extreme Portable SSD (amzn.to/3yPcMJ2) and the SanDisk Extreme PRO Portable SSD (amzn.to/3MKbvZl). The "Pro" version boasts twice the speed of the regular version (2,000 MB/s vs. 1,000 MB/s), but this can be misleading. Many computers do not support "USB 3.2 Gen 2x2", and your computer may only take advantage of half the advertised speed anyway. Do your research first. Even the standard version should provide 1,000 MB/s of read access, which is eight-times faster than most mechanical drives. Also consider disk space. While a $100 1 TB drive might be enticing, how quickly will you fill it up? I believe the SanDisk 2 TB Extreme Portable SSD (amzn.to/3yPcMJ2) is a great starter drive. It can be purchased for well under $200 and should be compatible with any modern computer.

Once you have your desired computer, you should configure the software required for the lessons within this section. If you are using a Linux machine for these tasks, and I believe you should, you already have most of the utilities needed for our commands. Programs such as cut, sed, and awk are included within most Linux systems. However, Ripgrep is not. Make sure you have Ripgrep installed with the following commands.

- Linux: `sudo apt install -y ripgrep`
- macOS: `brew install ripgrep`
- Windows: `choco install ripgrep`

While I could agree with some scenarios of querying breach data within macOS machines, I would never consider a Windows OS host simply due to the common presence of Windows-targeted viruses within the data sets we will acquire.

It is now time to dive into some data. We will take things slow at first and then visit more questionable content. **Every command within this section is available on my site within your book portal page or directly at https://inteltechniques.com/osintbook10/data.txt.** Be sure to have your username (**osint**) and password (**book1443**) ready when you access your portal.

Please do not skip the following chapter! While you may have no interest in leaked data, the included tutorials explain usage of several important commands which will be abbreviated in the chapters about breaches, logs, and ransomware. At the minimum, please familiarize yourself with the overall techniques, and not necessarily the specific data collection.

CHAPTER THIRTY
DATA LEAKS

The chapter on data breaches and leaks within the previous edition of this book began with a demonstration involving voter data. I included links to websites which hosted the entire voter databases of nine states. The person who hosted those sites had filed FOIA requests for the public voter data, and published the results when received. In early 2022, all nine sites began forwarding to a genealogy site which did not possess any voter data. However, data never dies.

In early 2022, a new site called **BreachForums** (breached.to / breached.vc / bf.hn) surfaced to replace the recently seized RaidForums site. The new site is basically a clone and serves as a marketplace for breaches and leaks. **I do not recommend visiting this site until you have a full understanding of the consequences.** I also do not ever recommend purchasing credits which allow download of stolen data. There is no need, as we will access the data in other ways without paying criminals through a shady website. In June of 2022, BreachForums released an official database of complete voter registration details from twenty-eight states. This included the original leaked data I discussed in the previous edition, plus numerous additional states acquired from other leaks. All of this data is technically public, but it was never meant to be published in this way. The original source of this voter data was official government agencies, and the data was obtained legally. The publication in this format is likely against some, if not all, state laws.

If you believe possession of this type of invasive data is illegal, consider Ohio's stance. If you navigate to https://www6.ohiosos.gov/ords/f?p=VOTERFTP:HOME and click on the "Statewide Voter File" tab, you can download the entire database of registered voters directly from the official Ohio government domain. This is truly public data and avoids the use of third-party providers. The data set for Washington state is also available from the government if you promise not to further distribute or sell the content. You can download the entire database directly at https://www.sos.wa.gov/elections/vrdb/extract-requests.aspx. Other states' voter databases are also out there, and Google should get you to them. Today, you can pay a few bucks to download this data for 28 states from BreachForums, or you can rely on free cloned copies floating around the internet. I prefer the latter. The data set includes millions of people's names, dates of birth, and home addresses (some otherwise unpublished), along with numerous email addresses and cellular telephone numbers. I hope by now you could see how this might be beneficial to online investigations. Figure 30.01 displays the availability of the data set on BreachForums.

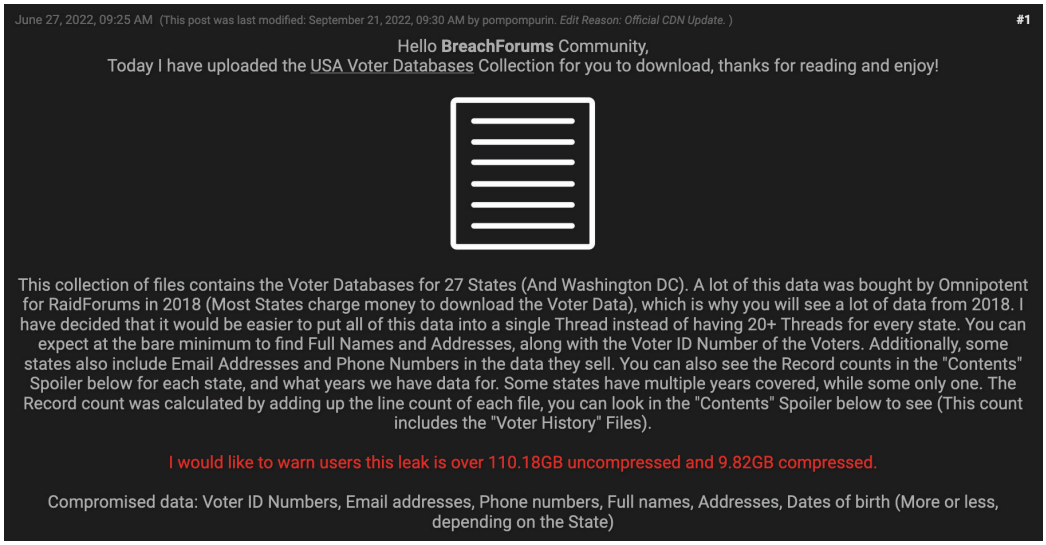


Figure 30.01: A BreachForums post offering full voter registration details.

Using the information within this post, I began conducting numerous queries across the open and dark web. My most successful query was "USA Voter Database" within the "Tor.link" search option included on the IntelTechniques Search tool. This led me to a Tor page which had to be accessed from within the Tor Browser, as previously explained. Figure 30.02 displays a current similar result.

Figure 30.02: A search result for a voter database.

It should be noted that the specific link I used to download this data is no longer available. However, you should be able to find a new link using the OSINT skills you have learned up to this point. For now, please focus on the strategies explained within this chapter before attempting to download this data to your collection. We have a lot of preparation to discuss which will be vital for the remaining chapters within this section.

The file downloaded through this site was titled USVoterData_BF.7z (this is another hint for your queries). It was almost 10 GB in size (compressed). Once I decompressed the file within my dedicated Linux host machine, it expanded to 118 GB. This would have likely crashed my virtual machine, and stresses the importance of a fast drive with plenty of space. We will clean this data into a more manageable file, but we will need space to do that.

The decompressed file presented a new folder titled USVoterData_BF, which contained several additional folders, one for each state. Inside each state's folder, I saw multiple years of voter data. Much of the content within these files is of no use to me, so I like to minimize the file size as much as possible for faster queries. Let's conduct a full demonstration using the 2018 Florida voter registration data within this set. The folder path and structure is as follows.

USVoterData_BF > data > Florida > 2018 > Voters

Inside this folder was 67 text files representing each of the counties within the state. Personally, I do not like having a separate text file for each county in Florida. I would rather combine all of them into one single file and title it appropriately. If you start ~~hoarding~~ collecting data, you may want to keep your collection tidy. Therefore, I would open Terminal and type the following commands. The first navigates you into the folder, and the second combines all of the files into one file titled Voter-FL-2018.txt.

```
cd ~/Downloads/USVoterData_BF/data/Florida/2018/Voters
cat * > Voter-FL-2018.txt
```

You could now move that file to another folder on your computer titled Databases, and delete the Voters folder within this 2018 data. This new large text file may take a long time to open, which is quite common with these datasets. Therefore, let's clean it up. I placed the 2.21 GB Voter-FL-2018 file directly in my Downloads folder for easy access.

Now that we have Ripgrep installed, we can use it to peek into very large files. First, let's take a look at the structure. The following commands within Terminal in Linux or Mac will navigate you to the Downloads folder and conduct a search for the name "Williamson" within the new file we just created. The search parameters will be explained in a moment.

```
cd ~/Downloads
rg -a -F -i -N Williamson Voter-FL-2018.txt
```

The output appears sporadic and similar to the text below. This is because there are unnecessary "tabs" in each line which causes the text to appear broken.

```
ALA     100397608      Williamson            Glenda  Dianne  N       217218  NW 48Th TER
        Gainesville            32606                          08/17/1951      08/14/1980      DEM
        22      0       22C3            ACT     3       21
```

The following optional command will replace each tab with a colon, which compresses the data and makes it easier to read. The same search from the previous example now produces the text visible below the command. This is much cleaner and will make future search results easier to digest. Note that the spaces in the command are created by pressing the control, v, and tab keys at the same time. This represents a "tab" to the command. When sed is executed, you will not see any output or indication it is working. However, you will be prompted when complete.

```
sed -i 's/    /:/g' Voter-FL-2018.txt
```

```
ALA:100397608:Williamson::Glenda:Dianne:N:217218  NW 48Th TER : :Gainesville:
:32606::::::::F:5:08/17/1951:08/14/1980:DEM:22:0:22C3::ACT:3:21:8:0:0::::
```

The presence of the unnecessary colons bothers me, so I will execute the following command twice to eliminate any two consecutive colons, and replace them with a single colon.

```
sed -i 's/::/:/g' Voter-FL.txt
```

Repeating the original search for Williamson now reveals the following output.

```
ALA:100397608:Williamson:Glenda:Dianne:N:217218  NW 48Th TER :Gainesville:
:32606:F:5:08/17/1951:08/14/1980:DEM:22:0:22C3:ACT:3:21:8:0:0:
```

If you have your Terminal window maximized, this all fits quite nicely on one line. I may have created more confusion than provided answers, so let's dissect each process. This will be beneficial in our future examples. In order to search the data collected, we will be using Ripgrep, as previously mentioned. The options we will be using are explained below.

rg	The command to launch Ripgrep
-a	The switch to search all data as text
-F	The switch to treat the pattern as a literal string
-i	The switch to ignore case
-N	The switch to exclude the line number

In my previous demonstration, I added a switch (-) for each option, which is overkill. We can add all of the options within one command and only one switch. In other words, our original Ripgrep query of our voter data file which was `rg -a -F -i -N Williamson Voter-FL-2018.txt` can be replicated much more simply as `rg -aFiN Williamson Voter-FL-2018.txt`. Ripgrep is a very powerful query tool with many options. Let's work through the most common query parameters which we will use over the next several chapters.

`rg -aFiN Test@Gmail.com`	Search EXACTLY test@gmail.com or TEST@GMAIL.com
`rg -aiN Test@Gmail.com`	Search ALL test and gmail and com
`rg -aFN Test@Gmail.com`	Search ONLY Test@Gmail.com and not test@gmail.com
`rg -aFi Test@Gmail.com`	Search EXACTLY test@gmail.com and show line #
`rg --help`	Show Ripgrep help menu

Each of these commands will search within all of the files contained in the current folder. If you are in Terminal within your Downloads folder and conduct these queries, they will search every file present within that folder, which may be undesired. As you begin to collect multiple large files, you will want to specify the file desired. The following would only search through our new voter file.

```
rg -aFiN Test@Gmail.com Voter-FL-2018.txt
```

If you want to search two specific pieces of data within a single file, the following would apply. It would display only lines within our file which contain BOTH "Michael" and Bazzell".

```
rg -aFiN "Michael" | rg -aFiN "Bazzell" Voter-FL-2018.txt
```

If you want to search two potential pieces of data within a single file, the following would apply. It would display only lines within our file which contain EITHER "Bazel" or Bazzell".

```
rg -aFiN "Bazel|Bazzell" Voter-FL-2018.txt
```

Overall, we will almost always use the combination of rg -aFiN "target". The commands that begin with "sed" tell the operating system to modify data. In the previous command of `sed -i 's/::/:/g' Voter-FL.txt`, the following should help explain.

`sed`	The command for "Sed"
`-i`	Modify the file and save it in real-time
`'s/::/:/g'`	Replace every occurrence of :: with : throughout the entire file
`Voter-FL-2018.txt`	The file to modify (* would modify all files in the folder)

If you feel overwhelmed, please don't let that convince you to abandon this chapter. I promise that the commands will start to make more sense as we progress. Let's jump back into the search results we received. They clearly identify the full name, home address, gender, date of birth, date of last registration change, and party affiliation. I find this extremely useful, but all of this could have been obtained from some previously mentioned websites. Therefore, let's conduct a search for any registered voters that live on the street of our previous example. The following would be the query, providing our search in single quotes because it contains a space. Instead of specifying the file, we can leave that off and it will simply search every file in the folder.

```
rg -aFiN 'NW 48Th TER'
```

The response contains the same type of details as seen previously, but displays hundreds of people. You cannot replicate these results with a traditional manual search on the sites themselves. Furthermore, we could conduct these searches across all of our databases, which we will do later. You could replicate all of this instruction throughout the various voter registration databases available for download. You might eventually have a single file for each state, with all of the unnecessary content removed from within the files. Some readers may choose not to modify the files while others may sanitize them to the point of ultimate conservation of file size. Each user's collection will be unique. You might try the following search options.

`rg -aFiN @gmail.com`	Display all Gmail accounts within the voter files
`rg -aFiN rita@gmail.com`	Search a specific email address within all the files

You could now replicate our merging of the 2018 Florida data to the 2013 and 2021 versions. Afterward, you could combine all three files into one and remove all of the duplicates. Let's do that together. The following assumes you have NOT taken the previous cleaning steps and would be conducted on the freshly downloaded data.

```
cd ~/Downloads/USVoterData_BF/data/Florida/2013
cat * > Voter-FL-2013.txt
mv Voter-FL-2013.txt ~/Downloads
cd ~/Downloads/USVoterData_BF/data/Florida/2018/Voters
cat * > Voter-FL-2018.txt
mv Voter-FL-2018.txt ~/Downloads
cd ~/Downloads/USVoterData_BF/data/Florida/2021/Voters
cat * > Voter-FL-2021.txt
mv Voter-FL-2021.txt ~/Downloads
cd ~/Downloads
cat Voter-FL-2* > Voter-FL-All.txt
```

Now, allow me to explain what each command accomplished.

- Change the active directory to the 2013 Florida data.
- Combine all of the 2013 files into one.
- Move that new file into our Downloads folder.
- Change the active directory to the 2018 Florida data.
- Combine all of the 2018 files into one.
- Move that new file into our Downloads folder.
- Change the active directory to the 2021 Florida data.
- Combine all of the 2021 files into one.
- Move that new file into our Downloads folder.
- Change the active directory to Downloads.
- Combine all files which begin with Voter-Fl-2 into one file called Voter-FL-All.txt.

Our new file of all Florida voter data is now 6.5 GB, which is quite large. It contains many duplicate lines because most voters have not changed their details since 2013. The following command removes any lines which are exact duplicates, sorts the data alphabetically, and saves the new file as "Voter-FL-All-Cleaned.txt".

```
sort -u -b -i -f Voter-FL-All.txt > Voter-FL-All-Cleaned.txt
```

I could then replicate the following commands to remove the unnecessary tabs and colons to save a bit more space if desired.

```
sed -i 's/    /:/g' Voter-FL-All-Cleaned.txt
sed -i 's/::/:/g' Voter-FL-All-Cleaned.txt
```

At this point, I could delete all of the Florida data with the exception of my "`Voter-FL-All-Cleaned.txt`" file. It contains everything I need from that state. We started with a folder called "Florida" which contained 12 GB of voter data. We ended with a single file called "`Voter-FL-All-Cleaned.txt`" which is 5 GB and contains no duplicates. I can now use Ripgrep to query that single file and access all of the Florida voter data for the past decade. The smaller single file will be much easier and faster to search and store.

It is important to note that each state stores their data in a unique format. The Florida example was just to begin the conversation about merging and cleaning of data. There is much more to discuss. Many of the voter files are already stored within comma-separated value files (CSV) and could simply be opened with any spreadsheet program. Ripgrep will easily search through those for you and you may not need to take any further action. Some are text files with a slightly different format. Please remember that this tutorial is not intended to provide a single solution to all data you might encounter. We are only dipping our toes into the world of leaked data.

Before leaving voter data, let's conduct one more demonstration in order to emphasize the power of the "cut" command. In that specific voter data, a folder of Colorado 2021 registration data contained nine text files, each about 200 MB in size. Opening each of the files displayed the following first line of the data.

"VOTER_ID","COUNTY_CODE","COUNTY","LAST_NAME","FIRST_NAME","MIDDLE_NAME","NAME_SUFFIX","VOTER_NAME","STATUS_CODE","PRECINCT_NAME","ADDRESS_LIBRARY_ID","HOUSE_NUM","HOUSE_SUFFIX","PRE_DIR","STREET_NAME","STREET_TYPE","POST_DIR","UNIT_TYPE","UNIT_NUM","RESIDENTIAL_ADDRESS","RESIDENTIAL_CITY","RESIDENTIAL_STATE","RESIDENTIAL_ZIP_CODE","RESIDENTIAL_ZIP_PLUS","EFFECTIVE_DATE","REGISTRATION_DATE","STATUS","STATUS_REASON","BIRTH_YEAR","GENDER","PRECINCT","SPLIT","VOTER_STATUS_ID","PARTY","PREFERENCE","PARTY_AFFILIATION_DATE","PHONE_NUM","MAIL_ADDR1","MAIL_ADDR2","MAIL_ADDR3","MAILING_CITY","MAILING_STATE","MAILING_ZIP_CODE","MAILING_ZIP_PLUS","MAILING_COUNTRY","SPL_ID","PERMANENT_MAIL_IN_VOTER","CONGRESSIONAL","STATE_SENATE","STATE_HOUSE","ID_REQUIRED"

This is because the files are actually comma-separated value (csv) files intended to be opened in a spreadsheet application, such as Excel. It is very common that leak data will be in this format, which is always preferred. As you can see in the text above, the Colorado data contains 51 columns, and each column possesses a specific type of data. I do not want to store every column of this data; I may only want to keep the following.

"LAST_NAME","FIRST_NAME","RESIDENTIAL_ADDRESS","RESIDENTIAL_ZIP_CODE","BIRTH_YEAR","PHONE_NUM"

I will need to identify the chronological number for each of these columns using the first full example. If I want only the data above, the following command would extract only those columns from each of the text files and save it to a single file called Voter-CO-2021.csv. I must first make sure I have already navigated to the proper directory within Terminal.

```
cd ~/Downloads/USVoterData_BF/data/Colorado/2021
cut -d, -f4,5,20,23,29,37 *.txt > Voter-CO-2021.csv
```

When I did this, I began receiving errors about "Illegal byte sequence". This happens often when the command encounters characters which it does not understand. Because of this, I typically begin my cut commands with "LC_ALL=C", which sets the character type and often bypasses these errors. My command is as follows.

```
LC_ALL=C cut -d, -f4,5,20,23,29,37 *.txt > Voter-CO-2021.csv
```

We should understand each part of the command, as follows.

`LC_ALL=C`	Sets the character type to avoid errors.
`cut`	The "cut" command.
`-d,`	Identifies the delimiter, which is a comma in our case (,).
`-f4,5,20,23,29,37`	Identifies the desired column by number.
`*.txt`	Identifies the input (all txt files).
`>`	Directs the result to output somewhere.
`Voter-CO-2021.csv`	Determines the new file name.

I have collected CSV files of leaked data which possessed over 200 columns of data. Many times, the majority of this content is not valuable and taking up precious space. The smaller your files are, the faster you can search them with Ripgrep. Note that I chose a csv output file extension in this example so that the command would not start cutting columns from itself as a text file. When the command finished, I renamed `Voter-CO-2021.csv` to `Voter-CO-2021.txt`. A partial redacted result of the `Voter-CO-2021.txt` is as follows.

"IVERSON","JOANN","1468 S xxx ST","80232","1931","303-xxx-67xx"
"WILLIAMS","WILLIAM","555 xxxx MOUNTAIN DR","80820","1941","303-xxx-xx85

I do not like all of those unnecessary quotation marks, so I ran the following command to eliminate them.

```
LC_ALL=C sed -i "s/[\"]//g" Voter-CO-2021.txt
```

My data now appears as follows (redacted).

IVERSON,JOANN,1468 S xxx ST,80232,1931,303-xxx-67xx
WILLIAMS,WILLIAM,555 xxxx MOUNTAIN DR,80820,1941,303-xxx-xx85

This is much better. My file contains many otherwise unavailable names, addresses, numbers, and the year of birth. We started with a folder called "2021" within "Colorado" which contained 1.7 GB of voter data. We ended with a single file called "`Voter-CO-2021.txt`" which is 200 MB and contains only the stuff we want. I can now use Ripgrep to query that single file and access all of the Colorado voter data for 2021. I could replicate this for the other Colorado sets if desired. I could also clean up all of the other states and make one huge file caller Voter.txt if I really want to minimize things. As you see, there are always many ways to do this.

Hopefully, you agree that possessing your own collection of this type of data could be useful. Let's add another collection of publicly available data. An individual previously purchased the entire Social Security Death Index, which is public data, and uploaded it for public consumption. This entire database can be downloaded from **https://archive.org/download/SocialSecurityDeathIndex/2013-05-31**/. After downloading the multiple files, you can combine them as stated previously into a single file and name it SSDI.txt. Assuming you placed the text file in the Downloads folder, the following commands would navigate to the folder and search for anyone in the file with my last name.

```
cd ~/Downloads
rg -aFiN bazzell SSDI.txt
```

The results will include multiple entries with the following format.

```
433220353 BAZZELL                 DOROTHY       S             0118201303191921
```

The first set of numbers is the Social Security Number of the deceased individual, followed by the last name, first name, and middle initial. The last set of numbers represents the date of death (01/18/2013) and the date of birth (03/19/1921). You could choose to replicate the "tab" removal process described earlier, or just leave this file as it was created. Let's recap where we are at right now. You might have single, very large, text files that each include all of the registered voter data for various states. These often include dates of birth, names, telephone numbers, addresses, and email accounts. You also have the entire SSDI. You can now search through all of that data with a simple search. If you wanted to identify any entries within all of the files, with a telephone number of 618-555-1212, the following would be the most appropriate searches. You could also now search by social security number, date of birth, or date of death.

```
rg -aFiN 6185551212
rg -aFiN 618-555-1212
```

Let's move on to something in a grey area. There are many companies that collect millions of email addresses and create spam lists. These then get sold to companies selling fake pharmaceuticals or designer handbags, and those unwanted messages end up in your inbox. We are all on a spam list somewhere. These are very expensive to buy, especially the good ones. One such database is the Kyle Data, or "Special K" spam list. We used this example in the browser chapter when DownThemAll easily acquired all of the data. This database contains 178,288,657 email addresses. There are no passwords, but it does contain other sensitive data. In November of

2015, a web page at updates4news.com temporarily possessed active links to this entire data set. This page is long gone by now. Fortunately for us, the Wayback Machine archived all of it. The following direct link displays hundreds of csv spreadsheets, each containing millions of email addresses.

https://web.archive.org/web/20151110195654/http://www.updates4news.com:80/kyledata/

If desired, you could use the Firefox extension mentioned previously to automatically download all of this data overnight into a folder on your Linux desktop titled "SpecialK". You could then execute a command through Terminal within that folder of `cat * > SpecialK.txt`. The result would be a single file, 19.73 gigabytes in size. That is very big, but also very powerful. Let's take a look at the content. Assume I was looking for a target who possessed an email address of robynsnest2006@yahoo.com. Assuming that I had made a single file titled SpecialK.txt within my Downloads folder on my Linux machine, my commands in Terminal would be as follows. The result of the search is directly after the search command.

```
cd ~/Downloads
rg -aFiN robynsnest2006@yahoo.com SpecialK.txt
```

```
MANHATTAN BEACH, robynsnest2006@yahoo.com,yahoo.com, Robyn Bazzell, 2015-04-07, 72.129.87.179,
paydayloaneveryday.com, CA, 90266
```

This tells me that my target lives in Manhattan Beach, CA 90266. Her name is Robyn Bazzell, and on 04/07/2015 her marketing data was collected from the website paydayloaneveryday.com. At the time, her computer IP address was 72.129.87.179. We basically just converted an email address into a full dossier on our target. This type of data is simply not available on public websites. If you doubt the validity of this data, please consider searching your own content. This is only one of dozens of interesting databases stored within the Wayback Machine. It is up to you to exercise your skills from previous chapters to locate them all.

Let's conduct one more archive.org demonstration before moving on to credentials. In January 2014, Snapchat had 4.6 million usernames and phone numbers exposed. The breach enabled individual usernames, which are often used across other services, to be resolved to phone numbers. This file is titled SnapChat.7z and can be found at archive.org/download/SnapChat.7z. This link contains the entire breach inside one text file. Conducting the following search would query a specific Snapchat username for any leaked data. Directly after this command is the result.

```
rg -a -F -i -N mikenap115 SnapChat.txt
```

```
('21220392XX', mikenap115, '', '')
```

This identifies the cellular telephone number of our target to include 212-203-92xx. While the last two numbers are redacted, you could try all 100 combinations within the previously discussed telephone search options. It is a great start. You could have also conducted this query on websites which possess this data. However, those sites require an absolute username. It cannot search partial names. If we only knew that our target had a Snapchat name that included the term "topher4", this database would provide the following users of interest.

```
('30351923XX', 'topher451', '', ''),
('41572499XX', 'topher413', '', ''),
('71974140XX', 'topher456', '', ''),
('75426428XX', 'topher481', '', ''),
```

We can also now search by telephone numbers. If you know your target has a cellular number of 303-519-2388, you could conduct the following search, with the partial result appearing after.

- `rg -a -F -i -N 30351923XX SnapChat.txt`

```
('30351923XX', 'topher451', '', ''),
('30351923XX', 'ben_davis', '', ''),
('30351923XX', 'rosemcdonald', '', ''),
('30351923XX', 'cuzzycook', '', ''),
('30351923XX', 'angelgallozzi', '', ''),
('30351923XX', 'kmo85', '', ''),
('30351923XX', 'rinisob', '', ''),
```

Your target may not appear in the results, but minimal investigation could result in a powerful new lead. You should note that the full numbers were never publicly leaked, and the group that conducted this attack redacted the results to exclude the last digits.

Real World Application: In 2017, I was provided a Snapchat username, similar to "yourdaddy3", of a suspect in a human trafficking investigation. A quick search revealed the first eight digits of his cellular telephone number which did not prove to be beneficial to various online searches. However, conducting a search for "yourdaddy" without the digit revealed a new partial number associated with "yourdaddy1" and "yourdaddy2", all within the same area code. Using the reverse caller ID APIs mentioned previously, I attempted to identify the owners of all 100 possible numbers. 58 of them were not registered, which left 42 valid numbers returning to individuals living in the area. Of those, only 19 of them were associated with males. Of those, only three fit a physical description of the suspect. A cooperating witness easily identified the suspect from these options. I truly believe in the power of this data.

We have only discussed a few databases stored within archive.org. I promise you there are many others containing sensitive details which would help greatly within your investigations. Conducting searches for any specific breached databases displayed on notification websites such as haveibeenpwned.com/PwnedWebsites should reveal interesting results. You have learned about numerous search methods throughout this book. What can you find?

Open Databases

The next time you hear about a "misconfigured server" which exposed millions of customer details, this is also likely a data leak. Our best tool to find this data is Shodan (shodan.io), and you will need to be logged in to a free or paid account to conduct these queries. I will focus only on open Elasticsearch databases for now, which are extremely easy to access. Our Shodan search is as follows.

product:elastic port:9200 [target data]

As a demonstration, I want to search for open Elasticsearch databases which contain an index titled "Customer". Think of an index as the title of a table or section of the database. The following search on Shodan produces over 200 results including an index titled "customer".

product:elastic port:9200 customer

The first result is an open database with 401 GB of data. I have redacted the IP address and modified each address throughout this example. I will use a fictitious address of 34.80.1.1 throughout the entire demonstration. Figure 30.03 displays this result. Clicking the red square with arrow next to the redacted IP address connects to the database within your browser in a new tab. Figure 30.04 displays this result. This confirms that the database is online and open.

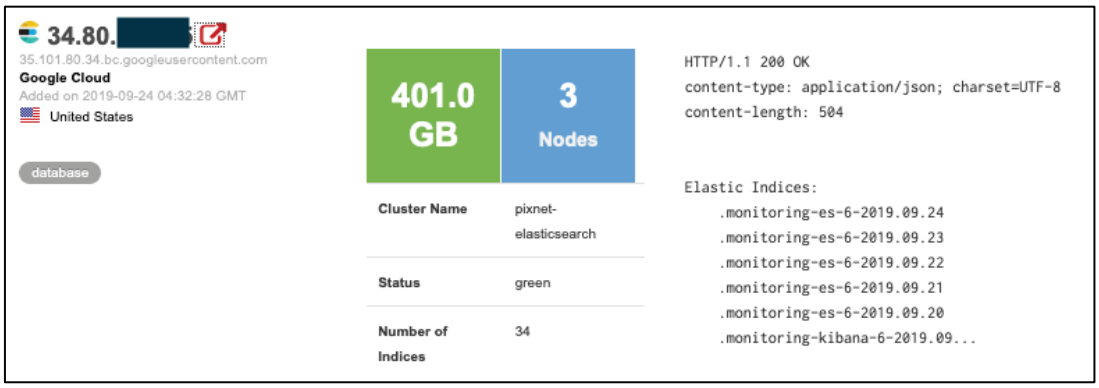


Figure 30.03: A Shodan result of an open Elasticsearch database.

Figure 30.04: An open Elasticsearch server.

Next, we want to obtain a list of all indexes within this database. These titles are often indicative of the content. The following URL queries this data based on a target IP address of 34.80.1.1. The result can be seen in Figure 30.05.

http://34.80.1.1:9200/_cat/indices?v

We now know there are several indexes which can be viewed. The first is titled "customer", but only contains a small amount of data (68 kb). Near the middle, we see an index titled "memberdisplayname20190903" which contains 124 MB of data including customer usernames. Some of these indexes contain email addresses and other sensitive data. Let's focus on the index titled "bank". We can view the content with the following URL.

http://34.80.1.1:9200/bank/_search?size=100

This combines the IP address (34.80.1.1), necessary port (9200), name of the index (bank), and instructions to display the first 100 records (/_search?size=100). If we wanted to view the maximum number of entries visible within the browser, we would query 10,000 records with the following URL.

http://34.80.1.1:9200/bank/_search?size=10000

Figure 30.06 displays actual redacted results from this query. It identifies the first name, last name, email address, gender, city, state, bank account number, and balance associated with a customer. The rest of the index contains the same information for additional customers.

health	status	index	uuid	pri	rep	docs.count	docs.deleted	store.size	pri.store.size
green	open	customer	xYPN7VCPQHuRdEfP5vEkvg	5	1	10	0	68.3kb	34.1kb
green	open	kkday	3kWWvvGSTOalAXMVOJa-eg	5	1	3514	115	22.3mb	11.1mb
green	open	minhash2	7aEhgxDDQ-2nu-Vpjpsibg	5	1	135617	2855	1.1gb	593.3mb
green	open	.monitoring-kibana-6-2019.09.18	6aE7CmBBQV2Iqk-8PeVV9A	1	1	74164	0	21.5mb	10.7mb
green	open	memberdisplayname20190903	OARMG9lLSP-rSOA49Pcc0g	1	1	220775	4567	124.6mb	62.3mb
green	open	.monitoring-kibana-6-2019.09.19	Yi7_DE_zTzaY7UCbIwVlWA	1	1	77753	0	22mb	10.9mb
green	open	streamtopic	KRzRxJ8eRd2Cpb5EG-tHXA	5	1	107379	771	211.4mb	105.7mb
green	open	articles20190114	5ZUblCdWQE6jo6JQK2J7sA	10	1	10531578	3885123	45.6gb	22.8gb
green	open	.monitoring-kibana-6-2019.09.20	taiMNlrvT3qMNTULnpFvEw	1	1	77579	0	21.5mb	10.8mb
green	open	articles20190130	4ZCi8yaxQvWN3hj2T0OECQ	10	1	0	0	5kb	2.5kb
green	open	member20181029	_8fDS93vSm-6hEECcEaLuA	5	1	7293911	20	94.1gb	47gb
green	open	.monitoring-es-6-2019.09.19	n0MGgBlbQrSE79xru9KlOw	1	1	346888	3384	461.6mb	231.9mb
green	open	articles20181224	7iO-SW40TPiORKtyj3qkJA	10	1	23422119	141719	114.1gb	57.3gb
green	open	minhash5	t6B40-XeRcq3Vx0NPEVAeA	5	1	223639	0	1.5gb	778.3mb
green	open	test	Q9fEJj9nRs6H4oXNvekTmg	10	1	674654	220802	29.9gb	15gb
green	open	.monitoring-es-6-2019.09.20	lj4OGJUDSJuG7EfSuQp6qA	1	1	354940	3666	478.9mb	239.6mb
green	open	.monitoring-es-6-2019.09.23	KgZ_kQUEThG5Olai27Yc9Q	1	1	346850	3102	487.5mb	245.2mb
green	open	minhash4	G9ablmkIQRCjyPx32-fVIA	5	1	223639	3	1.5gb	811.5mb
green	open	articles20181214	vbVscTkfTiCI64mlmHj44w	10	1	22582851	0	103.2gb	51.3gb
green	open	.kibana	6zZoT6nwTieIp3OlmTCUTw	1	1	4	0	44.8kb	22.4kb
green	open	.monitoring-es-6-2019.09.18	-oWrFwD-RBixBG-m4V2Osw	1	1	347378	3384	453.3mb	225.3mb
green	open	minhash3	yWXB2EJXS4itId5GhT-BLA	5	1	223639	2304	1.6gb	834.9mb
green	open	.monitoring-es-6-2019.09.22	Qg7IefQUSf65DUGlbOcNQA	1	1	347452	2820	474.2mb	236.8mb
green	open	rainytest	Dlgkerx_Q0iOl34yDeDePw	10	1	36	6	1.4mb	745.9kb
green	open	.monitoring-es-6-2019.09.24	Z9FjEfovRHe3sBCisb4Zog	1	1	224408	2820	347.4mb	185.2mb
green	open	.monitoring-es-6-2019.09.21	IqK2eVxwSHyX_mIzQLpkfg	1	1	346849	2365	472.5mb	235.6mb
green	open	.monitoring-kibana-6-2019.09.22	PzU8OoxKQJuMBurDPHmbmQ	1	1	77756	0	20.5mb	10.2mb
green	open	.monitoring-kibana-6-2019.09.24	o8yEB59SRZm7GbQjfCpDiw	1	1	50202	0	17.2mb	6.3mb
green	open	minhash	hFJWrKTiTP2RWRjTiRf1_w	5	1	135617	0	2.4gb	1.2gb
green	open	bank	DK8E3aiEQsCZPulCsHWzeA	5	1	1000	0	950.5kb	475.2kb
green	open	customer_service	mCAKa4duQKKSUJYtA4Jsdw	5	1	5677	0	16.2mb	8.1mb
green	open	articles20190917	crlh_EJITQGPq65SKHaaOg	10	1	1016016	0	2.3gb	1.1gb
green	open	.monitoring-kibana-6-2019.09.21	RIFVLfu_RUCh8oThv6sbJA	1	1	77753	0	20.3mb	10.1mb
green	open	.monitoring-kibana-6-2019.09.23	xoeXY0MuSo2yhptUCG29bw	1	1	77753	0	19.8mb	9.8mb

Figure 30.05: A list of indexes within an open Elasticsearch database.

Figure 30.06: An entry from an open Elasticsearch database.

Open Databases Issues

There are many complications with acquisition of open Elasticsearch data. The first is the record limitation. Displaying results within a URL limits the results to 10,000 records. Modifying the URL as described previously presents all data possible within the browser. Saving this page stores all of the content for future queries. However, many of the data sets I find contain millions of records. A Python script which parses through all results and stores every record is most appropriate, and is explained in a moment.

Next are the legal considerations. Technically, this data is publicly available, open to anyone in the world. However, some believe the act of manipulating URLs in order to access content stored within a database exceeds the definition of OSINT. I do not agree, but you might. I believe most of this data should have been secured, and we should not be able to easily collect it. The same could be said for FTP servers, paste sites, online documents, and cloud-hosted files. I believe accessing open databases becomes a legal grey area once you decide

to use the data. If you are collecting this content in order to sell it, extort the original owner, or publish it in any way, you are crossing the line and committing a crime. I remind you that the Computer Fraud and Abuse Act (CFAA) is extremely vague and can make most online activity illegal in the eyes of an aggressive prosecutor. Become familiar with the data access laws in your area and confirm that these techniques do not violate any laws or internal policies. My final reminder and warning is that I am not an attorney. I am not advising that you conduct any of these methods on behalf of your own investigations. I am simply presenting techniques which have proven to be extremely valuable to many investigators. If you believe that accessing an open (public) Elasticsearch database is legal in your area, and does not violate any internal policies, it is time to parse and download all content.

Elasticsearch Crawler (github.com/AmIJesse/Elasticsearch-Crawler)

In early 2019, I was made aware of an open Elasticsearch database which exposed sensitive data associated with 57 million people. In most cases, these records contained personal information such as first name, last name, email address, home address, state, postal code, phone number, and IP address. These types of records are extremely helpful to me as an investigator. Connecting personal email addresses with real people is often the best lead of all online research. I had found the database on Shodan using the methods discussed here. Specifically, I searched for an index titled "Leads" and sifted through any results of substantial size. Once I had located the data, I was desperate to download the entire archive. With a browser limit of 10,000, I knew I would need the help of a Python script.

I reached out to my friend and colleague Jesse and explained my scenario. Within a few moments, he sent me a small Python script. This file is now a staple in my data leaks arsenal of tools. He has agreed to share it publicly, please use it responsibly. If you installed all the software within the Linux chapters, you are ready for this tutorial. If not, you can enter the following commands within Terminal.

```
cd ~/Downloads/Programs
git clone https://github.com/AmIJesse/Elasticsearch-Crawler.git
cd Elasticsearch-Crawler
pip install nested-lookup
```

You are now ready to download an entire Elasticsearch open database and specify which fields should be acquired. Note that you must open Terminal and navigate to your script in order to launch this utility. I have included a desktop shortcut within your OSINT VM titled "Breaches/Leaks Tool" which automates this process, but let's understand the manual approach first. The following commands from within any Terminal session will launch the script.

```
cd ~/Downloads/Programs/Elasticsearch-Crawler
python crawl.py
```

This will present several user prompts to enter the target IP address, index name, port number, and fields to obtain. Let's walk through a real demonstration in order to understand the application. You must be logged in to a free or premium Shodan account in order to complete this tutorial. Using the IntelTechniques Breaches & Leaks Tool, seen in Figure 30.07 at the end of the chapter, I entered "leads" into the field titled "Elasticsearch". This conducted an exact search of "product:elastic port:9200 leads" within Shodan, which displayed a handful of results. One of these was an Elasticsearch server in India. This database appeared to contain test data, so I will not redact any of the results. The IP address was 111.93.162.238 and the database was approximately 1 GB in size. Clicking the red square within the result on Shodan opened a new tab to the following address.

http://111.93.162.238:9200/

The brief response confirmed that the server was online. The URL discloses the IP address (111.93.162.238) and port (9200). This is the default port and is almost always the same. Now that I had the IP address, I entered

it within the option titled "Index List" within the tool. Clicking that button launched a new tab at the following address.

http://111.93.162.238:9200/_cat/indices?v

This connects to the IP address (111.93.162.238) and port (9200), and then conducts a query to display all public indexes (/_cat/indices?v). The result included the following.

index	store.size
imobilestore	1.3mb
testcsv	190.7 kb
easyphix	1.8mb
index_test	7.2 kb
crazyparts	503.5 kb
valueparts	2.3mb
mobilemart	1.7mb
leads	280.8 kb

I usually look at both the names and the sizes. If I see an index titled "Customers", I know it usually contains people's information. If it is only 1 kb in size, I know it is too small to be of any use. When I see any index with multiple gigabytes of data, my curiosity kicks in and I want to see the contents. For this demonstration, let's focus on the index of our original search of "leads". Within our search tool, the next option is labeled "Index View". This requires the IP address of our target (111.93.162.238) and the name of the index you wish to view (leads). This opens a new tab at the following URL.

http://111.93.162.238:9200/leads/_search?size=100

This connects to the target IP address (111.93.162.238) and port (9200), loads the desired index (leads) and displays the first 100 results (/_search?size=100). This is usually sufficient to see enough target content, but this can be raised to 1000 or 10000 if desired. Below is a record.

"_index": "leads","_type": "leads",
"_id": "PXIhqmUBcHz5ZA2uOAe7",
"_source": {"id": "86",
"email": "test80@agencyinnovations.com",
"first_name": "test80","last_name": "test80",
"phone": "32569874",
"ip": "0.0.0.0",
"orgname": "Sales Arena",
"isDeleted": false,
"created_at": "2018-09-05 19:57:08",
"updated_at": "2018-09-05 19:57:08",

This is obviously test data, but assume it was a record containing a real person's name, email address, and phone number. Also assume there were over a million records within this index, which is quite common. We could save this page, but would be forced to save the undesired fields such as "tags" and "_source". Also, the data would be in a difficult format to search. This is where our new Python script is helpful. You have already launched the crawl.py script, and should have been presented with a prompt for the IP address of the target. The following displays each entry I submitted for this demonstration.

IP address: 111.93.162.238
Index name: leads
Port (Default is 9200): 9200

Field values to obtain (submit an empty line when finished):
Value: email
Value: first_name
Value: last_name
Value: phone
Value:

After being prompted for the IP address (111.93.162.238), it asked me for the target index name (leads) and port number (9200). It then prompted me to enter the first field I wanted to acquire (email). Since I entered a field, it then prompted for the next field (first_name). The tool will continue to ask for field names for as long as you provide them. Notice there is an empty line in the last "Value". This empty result tells the script you are finished, and it begins collecting the data. When finished, a text file will be saved in the same directory as your script. In this example, it was at ~/Downloads/Programs/Elasticsearch-Crawler/111.93.162.238-leads.txt. The title of the file was automatically created to reflect the IP address and name of the index. The following are the first three lines of this text file.

test65@agencyinnovations.com,test65,test65,987485746
test22@agencyinnovations.com,test22,test22,124958616
test69@agencyinnovations.com,test69,test69,2145968

If this were real data, you would see millions of people's email addresses, names, and telephone numbers. There are likely hundreds of legitimate databases on Shodan right now, just waiting to be found. The next time you see a news article about a security researcher who found an exposed database containing millions of sensitive records, it is very likely that Shodan and a similar script was used. If your downloaded file contains random text, you have likely encountered a patched version of Elasticsearch. At the time of this writing, Elasticsearch databases version 6.4.0 and newer were blocking the script. Anything older worked fine. There may be a new release of this crawler, and you may need to update your script as follows. Please note these commands are also included in the "linux.txt" file previously downloaded.

```
cd ~/Downloads/Programs/Elasticsearch-Crawler
git pull https://github.com/AmIJesse/Elasticsearch-Crawler.git
```

The final option within the search tool allows you to search a target IP address, index, and term. Let's conduct a new demonstration. Assume you have already searched "customer" within the first "Elasticsearch" option within the search tool. You have located an open database of interest and viewed the list of indexes with the "Index List" feature. You copied the name of the index and IP address to the "Index View" search. However, the file contains millions of records, and you can only see the first 10,000. You might want to search within the remaining records to see if your target is present. We can do this with the "Index Search" feature, as seen as the last option in Figure 30.07.

Enter your target Elasticsearch IP address in the first field and the target index name in the second. Enter any desired search term in the last. This could include the email address or name of your target. It is also less invasive than downloading all the content. In May of 2019, I located an extremely sensitive open Elasticsearch database. It contained Social Security Numbers and medical records. I did not want to download the data, but I did want to search my own name. I entered the IP address and index name in the first two fields and "Bazzell" in the last. The query returned dozens of patients' records associated with my last name, but nothing connected to me. This was all done within the web browser through their open server, and I did not archive any data onto my own machine. I identified the host and reported the leak anonymously. I never received a response, but the data was removed the next day.

I predict we will see fewer open databases in the future. While we still hear about sensitive leaks every week, word is spreading and companies are locking down their data. This is a good thing for all of us. Until then, I will continue to search.

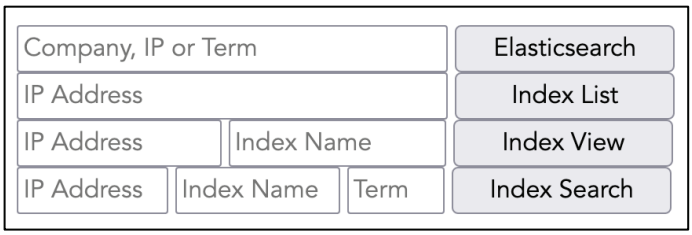


Figure 30.07: A portion of the IntelTechniques Breaches & Leaks Tool.

Public Data Sets

Many large data sets which are beneficial to investigations are not technically "leaks". This chapter has focused mostly on data which was never meant to become publicly available. However, there are numerous archives full of public information which should be considered for your data collection. Most archives cannot be searched through traditional resources such as Google. Instead, we must acquire the data, condense it, and conduct our own queries. As a demonstration, I will explain how I utilize Usenet archives as a vital part of my investigations. Usenet was my first introduction into newsgroups in the early nineties. My internet service provider allowed full access to thousands of topics through Outlook Express. I could subscribe to those of interest and communicate via email with people from all over the world. This sounds ridiculously common today, but it was fascinating at the time. I located a newsgroup about my favorite band, and I was quickly trading bootleg cassettes and exchanging gossip about the music industry. I was freely sending messages without any consideration of any abilities to permanently archive everything.

Today, the Internet Archive presents huge repositories of data containing over thirty years' worth of Usenet messages. It is presented in over 26,000 archives, each containing between 1 to 20,000 files. It is massive, and would take years to download manually. Fortunately, we can use a download script created by the Internet Archive to automatically obtain all desired data. The following tutorial will install the necessary software into our Linux virtual machine; download the entire Usenet archive; extract email addresses and names of each member; acquire newsgroup data to associate with each person; and condense the data into a usable format. This will take some time and may be overwhelming, but the final product is worth the effort. Let's begin.

First, we need to install the Internet Archive script within our OSINT Original VM. Within Terminal, enter the following commands.

```
mkdir ~/Downloads/Programs/internetarchive
cd ~/Downloads/Programs/internetarchive
python3 -m venv internetarchiveEnvironment
source internetarchiveEnvironment/bin/activate
sudo pip install -U internetarchive
deactivate
```

We now have the script installed and ready to use from any folder. Next, we need to identify the Internet Archive collections which we want to acquire. For this demonstration, I will focus on the following two data sets.

https://archive.org/details/giganews
https://archive.org/details/usenethistorical

These are two independent Usenet archives. Each contains unique records and some redundant data. Let's take a look at the second collection. The first folder is titled Usenet-Alt and contains over 15,000 files extracted from decades of conversations within the "Alt" communities. Opening the file titled alt.2600.fake-id.mbox reveals names, email addresses, dates, and entire messages dating back to 1997. The following is an excerpt.

From: "David N." <david.nine@juno.com>
Subject: Need Fake Texas DL
Date: 1998/07/11
Newsgroups: alt.2600.fake-id
I need a quality bogus TX DL for my 17 year old sister. Can you help me out?

Possessing a database of every email address and name from thirty years of Usenet posts can be very beneficial. Downloading every message can be overkill. This entire data set is over a terabyte in size. Instead of trying to download everything, I only want specific portions of the data. The Giganews collection includes two files for each archive. The first is the "mbox" file which includes the full messages along with user information. These are very large and could take months to download. The second is a "csv" file which only includes the date, message ID, name, email address, newsgroup, and subject of each post. This is a much more manageable amount of data which includes the main information desired (name and email address). We will only download the minimal information needed for our purposes.

First, we must create a text file which includes every archive within each collection. The following commands navigate to your Desktop and create text files for the Giganews and Usenet Historical archive.org data sets.

```
cd ~/Desktop
ia search 'collection:giganews' -i > giganews1.txt
ia search 'collection:usenethistorical -i > usenethistorical1.txt
```

Let's dissect the Internet Archive commands. "ia" is the application, "search" identifies the type of query, "collection:giganews" identifies the target data on archive.org, "-i" instructs the application to create an item list, and " > giganews1.txt" provides the desired output. These text files on your Desktop contain the names of all the archives within the collections. The following is an excerpt.

usenet-alt.2600
usenet-alt.2600a
usenet-alt.2600crackz

We can now instruct Internet Archive to begin downloading the necessary files. The following command downloads only the "csv" files from the Giganews collection. It can take several hours if you have a slow internet connection. If you do not have sufficient space within your VM, consider saving these to an external drive as previously instructed.

```
ia download --itemlist giganews1.txt --glob="*.csv.gz"
```

You should now have thousands of folders, each containing multiple compressed CSV files. This is not very useful or clean, so let's extract and combine all the valuable data. The following command will decompress all the files and leave only the actual CSV documents. It should be executed from whichever directory contains all of the downloaded folders. In my demonstration, it is in the Desktop.

```
gunzip -r .
```

"gunzip" is the command to extract the data from the compressed files, "-r" conducts the command recursively through any sub-folders, and " ." continues the action through every file. Below is a modified excerpt from one of the files. It identifies the date of the post (12/04/2003), name of the author (John Smith), email address associated with the account (John-smith@smh.com), specific newsgroup used (microsoft.windowsxp), and the subject of the post (Help me!).

#date	from	newsgroups	Subject
20031204	John Smith <John-smith@smh.com>	microsoft.windowsxp	Help Me!

We still have thousands of files, which is not ideal. The following command will combine every CSV file into a single text file titled Giganews2.txt. Furthermore, it will only extract the columns of data most valuable to us, as explained afterward.

```
find . -type f -name \*.csv -print0 | xargs -0 cut -f1,3,4,5 >
Giganews2.txt
```

Let's break down each portion of this command, as it can be quite useful toward other data sets.

find	This is the command to "find" data to manipulate.
.	This instructs the command to find all files.
-type f	This searches for a regular file type.
-name *.csv	This filters to only find a specific file extension (csv).
-print0	This directs output to a file instead of the screen.
|	This is a "pipe" character which separates the commands for a new instruction.
xargs -0	This builds our next command from the previous data as input.
cut -f1,3,4,5	This extracts only the data from columns 1, 3, 4, and 5 from each CSV.
>	This instructs the command to send the data to another file.
Giganews2.txt	This identifies the output file name.

The final result should be a very large file which contains all of the valuable content from within every downloaded file. In a moment, we will use this data to research our targets. The Usenet Historical collection is stored in a different format, so these instructions will need to be modified. The following steps will extract the beneficial data from that collection, and should appear similar to the previous actions. First, we must download the entire archive with the following command.

```
ia download --itemlist usenethistorical1.txt --glob="*.zip"
```

Next, we must extract the "mbox" files from their compressed containers with the following.

```
find . -name "*.zip" -exec unzip {} \;
```

Finally, we must extract the "From:" line from each file with the following command.

```
rg -aFiN "From: " > UsenetHistorical2.txt
```

This leaves us with a single large file titled UsenetHistorical2.txt. Below are a few lines.

```
alt.2600.mbox:From: "Bill Smith" <momop@mvc.biglobe.ne.jp>
alt.2600.mbox:From: "yosinaga jackson" <purizou@geocities.co.jp>
alt.2600.mbox:From: "yosinaga jackson" <purizou@geocities.co.jp>
```

We do not have as many details within this data set as we did with the Giganews option. However, possessing the names, email addresses, and newsgroups provides a lot of value. Both the Giganews2.txt and UsenetHistorical2.txt files possess many duplicate entries wasting valuable space. You might consider the following command which combines both files into one file while removing all duplicate lines.

```
sort -u -f Giganews2.txt UsenetHistorical2.txt > UsenetFinal.txt
```

We now possess a single file, quite large in size, which possesses the names, email addresses, and interests of most users of the Usenet system over a thirty-year period. Now, let's discus ways this can be used during investigations. Assume you are researching a target with my last name. When using Ripgrep, your command is as follows.

```
rg -aFiN bazzell UsenetFinal.txt
```

The result includes the following partial data.

```
microsoft.public.pocketpc,Steven Bazzell steveill@yahoo.com
sci.bio.microbiology,cbcWiW@UJOMHEH.EDU.TW Wayne A. Bazzell,M.P.S.E
sci.crypt,cbcWiW@UJOMHEH.EDU.TW Wayne A. Bazzell,M.P.S.E
sci.crypt,General Darcy J. Bazzell corporation@incidentally.mi.us
```

The first line identifies Steven Bazzell in the Usenet group of microsoft.public.pocketpc while using an email address of steveill@yahoo.com. You could search by names, email addresses, partial email addresses, domains, etc. I have successfully used my own Usenet archive in the following scenarios.

- When searching a unique target name, I have uncovered old email addresses which led me to otherwise unknown social network profiles.
- When searching an email address, I have identified various interests of the target, determined by the newgroups to which he or she has posted.
- When searching an email address, it serves as a great way to verify a valid account and establishes a minimum date of creation.
- When searching a domain, I often identify numerous email addresses used by the owner.
- When conducting background checks on targets over the age of 35, I often identify email addresses connected to questionable interests. While investigating a potential police officer, I located evidence he had previously posted images to a child pornography newsgroup. This was confirmed during the interview.
- When searching a username, I often locate confirmed email addresses from several years prior. These open new possibilities once provided to our search tools. In 2019, I located an otherwise unknown email address associated with my target username. It had been used to post to a hacking forum in 2002. The name, email, and username only identified "JohnDoe". This new email account was associated with a unique password within the Myspace breach data. Searching the password identified additional email accounts within other breach data sets. Searching those accounts within the Usenet data displayed the true name of my target. Everyone makes a mistake eventually.
- Any time I see a serial killer, active shooter, or otherwise deranged individual in the news, I seek any presence within old newsgroups. More often than not, if the subject was tech-savvy in the 90's, I find interesting evidence missed by most news media.

These collections possess over 43 million unique email addresses. You may be questioning the justification of the time it would take to create your own archive. While I encourage you to replicate the steps here, I also respect the difficulty involved. Therefore, I have made my own Usenet file available for download at the following address and shortened URL.

https://mega.nz/file/mSJGGDYI#oNIeyaG2oIHcHQFfGeFuq3zxUp_cCgARVf6bQNqp9ls
https://bit.ly/36937Pk

The file contains only the name, email address, and newsgroup fields from both collections in order to keep the size minimal. I have removed all duplicates and cleaned the file formatting to exclude any unessential data. It is 12 GB compressed to 3 GB. I hope you find it to be useful.

The JSON Dilemma

If you are lucky, you will always find a nicely organized CSV file with your new data, ready for "cutting" and storing. However, that is not always the case. Many times, you will encounter JSON data. At first, it may seem impossible to clean and minimize. The following is a partial example of JSON data extracted from a large file, which was shrunk to a small font in order to display the hierarchy.

```
"status": 200,
"likelihood": 6,
"data": {
  "id": "qEnOZ5Oh0poWnQ1luFBfVw_0000",
  "full_name": "sean thorne",
  "first_name": "sean",
  "middle_initial": "f",
  "middle_name": "fong",
  "last_initial": "t",
  "last_name": "thorne",
  "gender": "male",
  "birth_year": 1990,
  "birth_date": null,
  "linkedin_url": "linkedin.com/in/seanthorne",
  "linkedin_username": "seanthorne",
  "linkedin_id": "145991517",
  "facebook_url": "facebook.com/deseanthorne",
  "facebook_username": "deseanthorne",
  "facebook_id": "1089351304",
  "twitter_url": "twitter.com/seanthorne5",
  "twitter_username": "seanthorne5",
  "work_email": "sean@peopledatalabs.com",
  "personal_emails": [],
  "mobile_phone": "+14155688415",
```

The full portion for this single entry would have filled four pages in this book. Now image that the data set included 100 million entries. That would be 400 million pages if you wanted to print the data. If you were to query this selected data with Ripgrep, it would either only present the line where something was found (without any other data), or the entire entry (several pages of compacted text). If I searched "sean@peopledatalabs.com" via Ripgrep, the result might be "work_email": "sean@peopledatalabs.com" if the data was separated with line breaks. We would not receive all of the other associated content. You could use a lot of sed commands to try and move each piece of data into one line for queries, but that would be time consuming and unnecessary. When I encounter JSON data, I rely on **JQ** (stedolan.github.io/jq/).

First, we need to install it with the following commands:

- Ubuntu: `sudo apt install -y jq`
- macOS: `brew install jq`
- Windows: `chocolatey install jq`

We are now ready to launch JQ within Terminal (or Command Prompt). Assume we have a large file of JSON data titled people.json. The following commands would begin our extraction.

```
Cd ~/Downloads
jq --raw-output
'"\(.data.first_name),\(.data.last_name),\(.data.gender),\(.data.birth_year
),\(.data.birth_date),\(.data.linkedin_username),\(.data.facebook_id),\(.da
ta.twitter_username),\( .data.work_email),\(.data.mobile_phone)"'
people.json > people.txt
```

Let's dissect this:

`jq`	This command launches JQ.
`--raw-output`	This outputs pure text without formatting.
`'"`	This begins our string.
`\(.data.first_name),`	This extracts the first name field.
`\(.data.last_name),`	This extracts the last name field.
`\(.data.gender),`	This extracts the gender field.
`\(.data.birth_year),`	This extracts the birth year field.
`\(.data.birth_date),`	This extracts the birth date field.
`\(.data.linkedin_username),`	This extracts the linkedin username field.
`\(.data.facebook_id),`	This extracts the facebook field.
`\(.data.twitter_username),`	This extracts the twitter field.
`\( .data.work_email),`	This extracts the work email field.
`\(.data.mobile_phone)`	This extracts the mobile phone field (no trailing comma on final).
`"'`	This ends our string.
`people.json`	This identifies the input file.
`>`	This notifies of an output request.
`people.txt`	This identifies the new output file.

The result appears below.

```
sean,thorne,male,1990,null,seanthorne,1089351304,seanthorne5,sean@peopledatala
bs.com,+14155688415
```

If you had a JSON file with 100 million entries, you might have several billion lines of unsearchable data. You would now have a single file with the 100 million total entry lines, only containing your desired content. Notice in my command that I have "data" preceding each field request (`\(.data.gender)`). That is because those fields are under a section titled "data" within my JSON example on the previous page. If there was a section called "data" which contained a section called "people" which then contained a field called "gender", that field request would appear as follows.

```
\(.data.people.gender),
```

An entire chapter could be devoted to JQ and its usage. If you find this tool valuable, I encourage you to visit the Github page for the project. It includes complete usage and tutorials. I present a final warning about disk space. If you replicate all of these steps within a VM which is only allocated a small disk (less than 100 GB), expect to run out of space. If you plan to include data breaches and leaks into your daily routine, you should consider a dedicated host. Expect frustration, complications, and slow queries if using an external drive. The methods presented in this chapter can be conducted within any operating system as long as the proper utilities are installed. Please revisit my hardware requirement previously presented. Once you see the benefits of this data, you might be willing to make the jump.

LinkedIn 2021 Scrape

I want to offer one final demonstration in this chapter, and it is a big one which you might want to avoid. In 2021, an unknown group released a huge data set which contained over 400 million user profiles from LinkedIn. Profiles are public, so the headline did not seem interesting at first. However, this data included over 140 million email addresses and numerous cellular numbers associated with the profiles. That caught my attention. It is suspected that the data was acquired by querying email addresses and cellular numbers from previous leaks through the official LinkedIn API in order to identify the associated profiles. Additional public data was then used to connect Facebook and Twitter accounts. The result was 1.4 TB of JSON files which contained the personal email addresses and cellular numbers of many LinkedIn users. I wanted it, but in a better format.

Currently, there is an active Torrent containing the entire data, which consists of 700 compressed files, each approximately 300 MB in size and titled similar to "part-00055.gz", for a total of approximately 200 GB. Once extracted, the data is 1.4 TB, and the files are titled similar to "part-00055" with no file extension. Searching "400M LinkedIn Torrent" should begin your hunt, and might even lead you to Archive.org. There is also a copy available on "NitroFlare", but the free download tier can be quite slow. Searching "Linkedin Scrapped Data 2021" should find it. I downloaded the Torrent in an afternoon. I then decompressed all the files.

Each file contains JSON entries which possess a lot of redundant data I do not want in my final copy, similar to the previous example. However, I do want the fields titled first_name, middle_name, last_name, gender, birth_date, linkedin_username, facebook_username, twitter_username, work_email, mobile_phone, location_name, phone_numbers, and emails. I placed all of the "part" files in my Downloads folder and executed the following from that path.

```
jq --raw-output '"\(.first_name),\(.middle_name),\(.last_name),
\(.gender),\(.birth_date),\(.linkedin_username),\(.facebook_username),\(.tw
itter_username),\(.work_email),\(.mobile_phone),\(.location_name),\(.phone_
numbers),\(.emails)"' part* > LinkedIn.txt
```

This created a single file titled LinkedIn.txt, which only contained my desired data. It was 49.68 GB in size and possessed 400,101,052 entries. It contained a lot of entries containing "null" which offered no value, so I removed them with the following.

```
sed -i "s/null\,//g" LinkedIn.txt
```

The final result was a single 37.46 GB file. I can query this data within a few seconds and only see valuable information. That is a big difference from querying the original 1.4 TB data, which took up to an hour per query and displayed 90% useless data. The following is a slightly modified and redacted (xxx) entry from the new file. It displays my target's first name, middle initial, last name, gender, date of birth, LinkedIn username, Facebook username, Twitter username, location, telephone number, and email address.

```
chelsea, a, jansen, female, 1982-0x-xx, chelsea-jansen-685axxx, chelsea.
jansen.xxx, chelseaxxx, hartland, connecticut, +18608191xxx, cjansen@xxx.org
```

I can now associate the name, email address, cellular number, Twitter account, Facebook account, or date of birth with any of the other details. I can query a name and often identify the email address behind it. I can query a Facebook username and possibly identify the true person or date of birth. When I find a target on LinkedIn, I can query the unique portion of the URL (`chelsea-jansen-685axxx`) and see all information from this leak, including a valid email address. While this content does not possess all data within every field, it has been quite beneficial to my investigations. Consider the following examples.

- I searched the LinkedIn profile "seitzjustin" through the data and received the user's full name, Twitter handle, location, and a new email address from a custom domain which I did not previously possess. Historical records of this domain identified a physical address and telephone number.
- While stranded after a canceled flight, I queried the LinkedIn username of the airline's CEO and identified a personal Gmail account and cellular number. I made direct contact requesting a solution.
- A client was receiving harassing messages via Twitter. The Twitter handle of the suspect was associated with a LinkedIn profile and a 33Mail email address. The subdomain of the email address was only assigned to this subject, and a query through additional breach data revealed online profiles associated with the true name.

I hope you can see the power of this type of data. Cleaning the set from 1.4 TB to 37 GB took some time and effort, but I now have a small file for my queries, which completes in less than 30 seconds from an external SSD. I placed this file in the "People" folder of my script, which is explained at the end of Chapter Thirty-Three.

Vital Lessons

Before moving on, let's revisit some key lessons which will be valuable for the next few chapters.

- **Cat** is a command which allows us to combine multiple files into one large file.
- **Ripgrep** is a utility which allows us to search through large data files as fast as possible. It eliminates the need to possess a database with an index in order to conduct keyword searches.
- **Sed** is a command which allows us to substitute or remove portions of data within a file.
- **Cut** is a command which allows us to extract only the desired columns from a CSV file.
- **Sort** is a command which will remove duplicate lines and alphabetize our data within a file.
- **LC_ALL=C** can be used when unusual characters prevent a command from completing.
- **JQ** can be used to extract JSON data with ease, without the need for the previous utilities.

I encourage you to have a strong understanding of all these commands if you plan to include large data sets within your own investigations. There is plenty of information online about the advanced usage of these commands, and we will continue to visit new options as we proceed through these chapters.

We have only begun the journey into leaked data. I have personally obtained, cleaned, stored, and queried tens of thousands of valuable data leaks as part of my online investigations. Aside from the data presented here, there are countless copies of other data sets floating around. While I do not present specific tutorials within this book, the following may spark interest into beginning your own hunt for data.

People Data Labs: I mentioned this phenomenal resource previously in this book. In 2019, a security researcher claimed to have scraped an open database of over 1.2 billion PDL records including names, locations, and social network content. This data set is still floating around, and it is valuable.

Pipl: Similar to the previous example, a researcher claimed to extract 50 million profiles in 2019 from the people search site Pipl, containing full names, email addresses, vehicles owned, phone numbers and home addresses. Every week, I see someone offering a full copy for download.

IntelligenceX: I discussed this paste-scraping service in previous chapters. In 2021, a disgruntled customer "scraped their scrapes". Today, the 80,000 copied documents have been merged into one file possessing sensitive data associated with 46 million email addresses.

LiveRamp: In 2021, data allegedly sourced from LiveRamp (formerly Acxiom) was leaked. The data contains extensive information on most people living in the U.S., including home addresses, cellular telephone numbers, email addresses, names and more. It is commonly traded on Telegram, but the original source is unknown.

White Pages: Someone breached the White Pages site and copied 11 million profiles with user's names, email addresses, and passwords. It has been repackaged for years and offered as "new".

Verifications.io: This "enterprise email validation" service exposed their MongoDB database containing 763 million records including email addresses, names, genders, IP addresses, phone numbers and other personal information. It is widely available as a 150 GB download.

How do you find all of these valuable data sets? I dive into that within the next two chapters.

CHAPTER THIRTY-ONE
DATA BREACHES

Now that you understand the basics of searching and modifying data, we should discuss the true value of this technique. Until now, we have only experimented with publicly-available data. The real power lies within the stolen databases that are now publicly available which I am about to discuss. There are usually two types of people that download these databases. The first are amateur criminals that use the data to illegally obtain account access to victims' accounts by supplying the usernames and passwords from one service into another service. As an example, if a criminal learns your username and password that you used on LinkedIn, he or she may try that same combination on Twitter. We will never do this, as it is illegal and unethical.

The second group of individuals that download this data are security researchers. Numerous private organizations authorize employees to collect this stolen information and use the data to make their own systems more secure. I personally know of many researchers who download this data, protect it from further distribution, and only use it in an ethical manner. This is the only acceptable use of this data. For those that still believe that we should not access stolen databases released to the public, consider one last argument. Tens of thousands of criminal hackers use this content to "dox" people every day. Doxing is the act of exposing personal information about a victim online. This is a very common tactic used against law enforcement officers when an event such as an officer-involved shooting happens. The government employees' passwords often get leaked on Pastebin or other similar sites. This stolen data is already out there, and it can never be secured. We should embrace the same techniques used against us when researching criminals. Enough warnings. Use your best judgment.

By now you may just want to know where to obtain these databases. The source websites that allow download of this stolen data get shut down often. However, the magic of internet archives can help us regain the data. The next exercise also involves archive.org. Specifically, data from LinkedIn. This version surfaced in 2017 and does not possess any passwords. It contains only LinkedIn user ID numbers and the associated email addresses. While this type of data was included within the original 20 GB LinkedIn breach, it was excluded from the popular password dumps that can be found online today. Since it does not contain passwords, it can be found on the Wayback Machine. At the time of this writing, the direct link to download the entire user ID database could be found at https://archive.org/details/LIUsers.7z.

Decompress this file by right-clicking within Linux and choosing "Extract here". The text file contains the email addresses and user ID numbers extracted from the full LinkedIn breach. It is nothing more than each email address used to create a specific profile ID number. The following search within Terminal would display the results of a target email address, and the response is directly below the command.

```
rg -aFiN janewilson@microsoft.com linkedin_users.txt
```

```
1332567, janewilson@microsoft.com
```

You now know that your target's user ID number is 1332567. This data alone is not very helpful. Let's consider another perspective. Assume you find your target's LinkedIn profile page, and you want to know the email address used to create the account. Right-clicking on the profile allows the option to display the "Source Code" of the profile. Searching the term "member:" within this code presents numerous occurrences of that term. The third or fourth from last occurrence should appear similar to "member:1288635". This tells you that the member ID for your target is 1288635. The following displays the email address associated with that number, with the results immediately below.

```
rg -aFiN 1288635 linkedin_users.txt
```

```
1288635, John.h.007@gmail.com
```

Real World Application: I have used this technique numerous times over the past several years. During one investigation, I located the LinkedIn profile of my target. I had previously found no email addresses connected to him. By looking at the source code of the LinkedIn profile, I could see his user ID. Searching that user ID within this LinkedIn data provided the personal email address used to create the account. That email address led to old passwords, which led to additional email addresses, as explained in a moment. Obtaining a confirmed personal email address can lead to numerous new investigation techniques.

Credentials

Now that we are dipping our toes into the waters of "stolen" data, we should consider the big breaches. You have likely heard about LinkedIn, Dropbox, Myspace and Adobe being hacked. All of those breaches have been publicly released by various criminal organizations. At the time of this writing, most remaining sites which possessed this huge collection of stolen data had been shut down. However, the data lives on. Numerous "hacking" groups have collected tens of thousands of breached databases, both large and small, and combined them into credential lists. These lists contain only the email addresses and passwords of billions of accounts. They do not identify which breach each credential originated, but the data is extremely valuable for investigators. Before we analyze the content, we must obtain a full copy of the data. This is where things get tricky (again).

My attorney says I cannot HOST links to any of this content, which I understand. She also says that I cannot display any direct links to an identifiable breach, such as LinkedIn or Adobe. This would make me a target for a lawsuit, and seems like great advice. My only option is to "identify public resources which link to data without attribution of a specific originating source". In other words, I can tell you where you may find this "Combo List" content, but it is up to you to take action to obtain it. Let's connect our VPN; understand any employer policies which might prevent the following actions; and tiptoe into the world of stolen credentials.

In 2021, an unknown group of credential thieves created a huge collection of 3.2 billion credentials consisting only of email address and password combinations. This data set was titled "Compilation Of Many Breaches", otherwise known as "COMB". There have been numerous other combo lists released in previous years, such as Anti-Public, Exploit.in, and others. However, this set included a better search structure and ability to conduct queries within seconds. Some people may find that searching for "CompilationOfManyBreaches.7z" within Google might lead you to a copy. Others report that the following query will display the torrent file for this data.

site:anonfiles.com "CompilationOfManyBreaches.7z"

At the time of this writing, the single result from this search possessed a torrent file which could be opened with a torrent software application such as Transmission. This program will download the large file to your computer. This is a lot of data. It may be more content than your internal hard drive can store, and is definitely more than your VM is configured to handle. You may see better results by downloading it through your host, but every situation is unique. If appropriate, you may consider downloading the entire torrent to an external hard drive. During testing of this tutorial, I connected a 1 TB external USB drive to my VM and chose it as the destination for the downloaded files through Transmission. This kept my VM clean and all data was available on the external drive. This allows use within multiple VMs or host computers. Depending on your internet connection, this entire download can take hours or weeks to finish. If possessing user credentials violates your security clearance or organizational polices, do not proceed with this download.

Let's assume you now have a hard drive with the entire contents from COMB. The result is a 20 GB compressed file which contains all 3.2 billion credentials. The file itself is titled "CompilationOfManyBreaches.7z". If you chose to download this file, it must be decompressed. I prefer a utility such as 7-zip for Linux and Windows, or Keka for Mac. Upon decompression, you might be prompted for a password. A Twitter post located at https://twitter.com/BubbaMustafa/status/1370376039583657985 claims the required archive password is "+w/P3PRqQQoJ6g". Once decompressed, you should see a new folder titled "CompilationOfManyBreaches" which is 106 GB in size.

Assume you possess a folder called COMB within your external hard drive connected to your Linux VM which contains the data downloaded during this exercise. Opening Terminal and navigating to that folder may not be easy. You would need to know the exact path. Instead, open the Files application and you should see your external hard drive in the left menu. Click on it and select the COMB folder. Open Terminal, type cd, then space, then drag the COMB folder in the external storage from the Files application to Terminal. Strike enter and it should populate the path of the drive. Mine appeared as `cd '/media/osint/1TBExternal/COMB'`. You could also find the external drive within the Files application in Ubuntu, right-click, and choose "Open in Terminal". You should now be within your new data collection folder inside Terminal. We can conduct searches to start evaluating the benefit of this data. Since COMB includes a fast search option, we will start with it. Execute the following commands within Terminal from within the "CompilationOfManyBreaches" folder.

```
./query michaelbazzell@gmail.com
```

This should result in no hits within a few seconds. Now try the following.

```
./query michael.bazzell@gmail.com
```

The result should appear as follows.

```
michael.bazzell@gmail.com:password
```

This identifies a password "password". Notice that the first query failed because we did not include the "." within the search parameter. This tool can be fairly unforgiving and requires exact data. The following search provides any email address which begins with "michael.bazzell" with results which follow.

```
./query michael.bazzell
```

```
michael.bazzell@gmail.com:password
michael.bazzell@us.army.mil:redacted10
michael.bazzelle1970@yahoo.com:redacted201
```

Note that Mac users can submit queries as `bash /query michael.bazzell` and obtain the same results. This search tool is extremely fast. If you know the email address of your target, or at least the first portion of the address, searching through the native COMB query option is best. However, this presents limitations. You cannot use this tool to search a specific domain or password. For that we will once again rely on Ripgrep. The following queries will all assume that you have already opened Terminal and have navigated to the folder where your data is stored. The next search would attempt to identify a specific email address within all the files. Note that results were modified to protect the privacy of the individual. As a reminder, this command applies our parameters and identifies the target data to be searched. The result follows the command.

```
rg -aFiN mikewilson@microsoft.com
```

```
mikewilson@microsoft.com:bigbucks55
```

We now know that at some point a password of bigbucks55 was used in conjunction with an online account associated with our target email address. Would this password still work with any online accounts? Possibly, but we will never know. Attempting to log in to an account with this information is a crime. Instead, think about other ways that we could use this data offline. We know our target's work email account, but we may want his personal account. Since many people recycle the same password across multiple accounts, let's conduct a search for the password.

```
rg -aFiN bigbucks55
```

The results include accounts associated with our target, and some that are not. The more unique your target's password, the better quality of associated content you will receive. The following data was presented during my search. Notice the ways that data was displayed based on the search, and how any portion of the results were included. Since we specified our search parameters, we received results regardless of case.

```
bigbucks551@yahoo.com:towboat@1
bigbucks55@hotmail.co.uk:towboat@1
bigbucks55@hotmail.com:towboat@1
prizeunit@yahoo.com:bigbucks55
mike.wilson5@gmail.com:BigBucks55
mikewilson@microsoft.com:bigbucks55
```

This tells me that the last two results are very likely my target, since the names are similar and the passwords are almost identical. I can now assume that my target's personal email account is mike.wilson5@gmail.com. The first three accounts could be my target, but are probably not. This is likely just a coincidence. However, we can assume that the owner of one of those accounts is the owner of all three since the passwords are identical. The fourth response is a toss-up, but worth further investigation.

We can also use this to query all credentials from a specific domain.

```
rg -aFiN @altonpolice.com
```

One of the results is quite embarrassing, as follows. I promise I have not used that password since the late 90's.

```
bazzell@altonpolice.com:mb01mb01mb
```

Real World Application: The tactic of searching leaked passwords and recovering associated accounts is by far the most successful database leaks strategy that I have applied to my investigations. In 2017, I was assisting a federal agency with a child pornography investigation where a suspect email address had been identified. This address was confirmed as being connected to the original person that had manufactured illegal videos of children being molested, but an alias name was used during creation. A search of this address through a breach compilation revealed a unique password associated with the account. Searching this password revealed only one additional email account, which was the personal Gmail account of the suspect (in his real name). The suspect used the same password for both accounts. The primary investigator made an arrest the next day after confirming the connection. While some will say that we should never download leaked databases that contain personal login credentials, I disagree. There is great potential value in these data dumps that could solve cases on a grand scale, and make a huge impact on the prosecution of serious criminals.

I cannot overstate that this instruction is the very tip of the iceberg. There are tens of thousands of compromised databases online, and more being published every day. If you find "COMB" to be valuable, you may consider researching others. If you invest some time into seeking the sources of this data, you will quickly become overwhelmed at the mass amounts of content to properly obtain, clean, and store. I can only discuss the basics here. It is your job to proceed if you choose.

As a start, you may consider focusing on public credential leaks on Pastebin (pastebin.com). When you search an email address on Pastebin via Google, the results often include paste dumps. Clicking on these links will take you to the original paste document, which likely has many additional compromised credentials. A query of test@test.com on this site will likely present hundreds of data sets ready for download. Do you remember the "Index Of" technique mentioned previously? This is very useful in searching for leaks. Any time I see on HIBP that a new public leak has surfaced, I search for that specific data with a custom Google search. If I saw that Myspace has been exposed, I would use the following.

"myspace" ext:rar OR ext:zip OR ext:7z OR ext:txt OR ext:sql

There are many online database resources that will sell you the data. Please avoid these. First, you will likely get ripped off a few times before you find an "honest" seller. Second, you are giving money to criminals, and I don't like encouraging that behavior. Many researchers that I know possess over 10,000 databases which contain over four terabytes of total information, all found on public sites. A single query of this data can take a few minutes, even on a fast machine.

Hashes

Most websites store passwords in "hashed" form. This guards against the possibility that someone who gains unauthorized access to the database can retrieve the plain-text passwords of every user in the system. Hashing performs a one-way transformation on a password, turning the password into another string, called the hashed password. "One-way" means that it was practically impossible to go the other way and turn the hashed password back into the original password. This was true many years ago, but not so much today. There are several mathematically complex hashing algorithms that fulfill these needs. Some are very insecure and others are nearly impossible to crack. Let's look at a few examples.

Many older databases which have been breached possessed simple MD5 password hashes. The password of "password1234" is as follows as an MD5 hash. The Sha-1 hash of the same password, "password1234", is also as follows. Notice this is substantially longer, and a bit more secure. However, it will be quite easy for us to crack these passwords in just a moment. Below this example is the same password in Sha-256 and Sha-512 format. As you can see, these become increasingly complicated.

MD5: BDC87B9C894DA5168059E00EBFFB9077

SHA1: E6B6AFBD6D76BB5D2041542D7D2E3FAC5BB05593

SHA256: B9C950640E1B3740E98ACB93E669C65766F6670DD1609BA91FF41052BA48C6F3

SHA512: 8C7C9D16278AC60A19776F204F3109B1C2FC782FF8B671F42426A85CF72B
1021887DD9E4FEBE420DCD215BA499FF12E230DAF67AFFFDE8BF84BEFE867A8822C4

In regard to the various "older" breached databases which you are likely to find on the internet, you will most commonly see MD5 and SHA1 hashed passwords. Some of them possess a "salt". This is a small amount of data added to the hashing which makes cracking the password more difficult. If a breach does not possess the salt, the passwords are nearly impossible to crack. If the salt is present, it takes considerable additional resources in order to display the text password. The methods of "cracking" passwords exceed the scope of this book. Fortunately, we do not need the knowledge and computer horsepower to convert these hashes into valuable passwords. We simply need a third-party resource.

Hashes (hashes.org) attempted to reveal the plain text of your submitted password hash. This was usually done in an effort to assist security professionals to evaluate the security provided by the relevant hash submitted. For us, it provided a new lead to follow. The database contained billions of cracked hashes available via web search and API. If I queried " BDC87B9C894DA5168059E00EBFFB9077" via the search page on the site, I was immediately presented with "password1234" as the password. Unfortunately, the service disappeared in late 2020. However, online archives exist. I consider the following technique to be advanced, and only suitable for those who have a need to reveal hashed passwords as part of their daily operations. The entire archive of hashes and passwords which previously existed on hashes.org is available as a torrent file. The following websites contain a "magnet" torrent link within them. Copying and pasting this link within your browser should launch your default torrent software and begin the 90 GB compressed download. Make sure you have plenty of space.

https://pastebin.com/pS5AQNV0
https://old.reddit.com/r/DataHoarder/comments/ohlcye/hashesorg_archives_of_all_cracked_hash_lists_up

Once complete, you should have a folder titled "HashesOrg Archive" with folders of "Hashlists" and "Leaks" inside of it. These two folders contain thousands of compressed zip files. While you could issue commands via Terminal to decompress all files, I find it easier to simply select all files within a folder; right-click on them; and select to open with your desired decompression tool. Once you see the ".txt" versions of each file present, you might want to delete the original ".zip" files to free some space. Let's take a look inside the "Leaks" folder and open the file titled "20_casio-com_found_hash_algorithm_plain.txt". A partial excerpt follows.

MD5 00747af6279313863a0319070bdbfb80:168130
MD5 0075f8d9f099093bdba10e8b7e88b47c:8208201982

The data tells us that this specific breach stored passwords in MD5 format. In this excerpt, we know that the MD5 hash of "00747af6279313863a0319070bdbfb80" reveals a password of "168130". We also know that a password of " zhangker46" was used within the Casio website at the time of the breach. Both of these pieces of data will be valuable to us. Let's conduct an investigation.

Your target has an email address of "badguy42@gmail.com". A query of this email address within the COMB data reveals a password of "verybad1234". However, this does not tell you which breach is associated with this address. Within Terminal, after navigating to the "HashesOrg Archive" folder, the following command is issued.

```
rg -aFiN verybad1234
```

The result appears similar to the following.

```
Leaks/713_forums-nodoubt-com_found_hash_algorithm_plain.txt.zip
MYSQL5 887b0eb63dbc543991567864efc0b05aad5a8ab2:verybad1234
```

We now have circumstantial evidence that a user on a web forum for the band No Doubt was using a password of "verybad1234" and the MYSQL5 hash of that password is "887b0eb63dbc543991567864efc0b05aad5a8ab2". Does this prove that "badguy42@gmail.com" was using this forum? No, but it is a solid lead. It could also be someone else using the same password. Since Hashes.org does not share the username or email address, we must continue the investigation with the OSINT methods previously explained. Is that email address associated with any conversations about the band? That would give me more confidence with the result. We should look at the typical way that one would use the Hashes.org data set. Assume that you have identified the following data within a breach, leak, or online website.

badguy42@gmail.com:14FDF540E39F0F154C8D0B3BD82ACE100B779DFA

Searching "14FDF540E39F0F154C8D0B3BD82ACE100B779DFA" through the hash identification website **TunnelsUp** (tunnelsup.com/hash-analyzer) reveals the following.

14FDF540E39F0F154C8D0B3BD82ACE100B779DFA - Hash Type: SHA1

We know this is a SHA1 hash which represents a password, and can execute the following.

```
rg -aFiN 14FDF540E39F0F154C8D0B3BD82ACE100B779DFA
```

The results appear below.

```
Leaks/706_forums-utorrent-com_found_hash_algorithm_plain.txt.zip
SHA1 14FDF540E39F0F154C8D0B3BD82ACE100B779DFA:stillverybad1234

Leaks/1182_prowrestlingfans-com_found_hash_algorithm_plain.txt.zip
SHA1 14FDF540E39F0F154C8D0B3BD82ACE100B779DFA:stillverybad1234
```

We now know that "badguy42@gmail.com" likely used a password of "stillverybad1234" at some point in time. We also know that someone using the password of "stillverybad1234" used that password on a uTorrent forum and a wrestling website, both of which suffered a breach. Are these all the same person? We cannot definitely conclude that. However, these are great leads. My next search would be the following.

```
rg -aFiN stillverybad1234
```

This may identify more data to be analyzed. However, this is all circumstantial. If a password is unique and complex, I have more confidence in the relationship to my suspect. If the password is "password1234" and appears on hundreds of sites, this has no value.

The **HashMob** (hashmob.net) community now offers a paid search service and free downloads of lists similar to those previously found at Hashes.org. Registration is not required to download "Found" lists from the "Official lists" menu. These files are very similar to the Hashes.org data, but updated often. I downloaded all 2,326 official (non-combo) hash lists with the following command.

```
wget --content-disposition -i https://inteltechniques.com/data/hashlists.txt
```

WARNING: The result was 90 GB of data. A typical file was titled "330.3200.found", which identified the ID number of the breach (330) and the type of hash (3200-bcrypt). Each file contained numerous entries similar to 000392f0f900824707b3508abc78aa6bb89f1e61bca81b0bcb01bb2efc6a1057:yanks2008. Everything to the left of the colon is the hash, and everything right of it is the password. Searching "330" (the ID of the breach) at **https://hashmob.net/api/v2/hashlist/official** reveals this is the hash list from a breach at Zildjian.ru.

Just like the tutorials within the Hashes.org section, this data allows us to search for either a hash or a password to identify if either is present within known breaches. It also allows us to query a hash in order to identify its plain-text password. After witnessing Hashes.org disappear, I prefer to possess my own archive of this free data. However, I also maintain a paid account which allows real-time queries of hashes within their website, or via their API, which is much easier and immediately identifies passwords. If you need to query hashes often without downloading all the data, a paid account may be justified, and the results are updated daily. Next, let's take a closer look at some popular types of hashes.

MD5

Some older data breaches possess passwords hashed with MD5, an extremely insecure method. These hashes can be cracked very quickly. Below are two entries retrieved from various files.

174@gmail.com:482C811DA5D5B4BC6D497FFA98491E38
185@gmail.com:22F4182AAE2784FB3D1A432D44F07F46

Everything before the colon is the username or email address, and everything after is the MD5 hash. Searching these hashes in your Hashes.org data produces the following results. We now know that the first password is "password123" and the second is "reader12".

482c811da5d5b4bc6d497ffa98491e38:password123
22f4182aae2784fb3d1a432d44f07f46:reader12

MD5 + Salt (Vbulletin)

Some hashes will contain a "salt". This is usually a small piece of data after the hash, and both pieces are required to translate into plain-text. One of the most popular examples of this are the hundreds of Vbulletin online forums which have been infiltrated by various hacking groups. In the following examples, the salt is the final three characters after the last colon.

Aanas,menball@aol.com:9d9e3c372d054c0769bd93181240be36:tye
Traor,tranr@optusnet.com.au:9274583d060b3efb464115e65a8c1ead:vv#

Searching these hashes and salts through your Hashes.org data provides the following results.

9d9e3c372d054c0769bd93181240be36:tye:eliza!%
9274583d060b3efb464115e65a8c1ead:vv#:runner

In the first example, you see the target password is "eliza!%". This proves that semi-complex passwords containing special characters are often dehashed and ready for conversion. This is where Hashes sticks out from the crowd. There are many online reverse-hashing tools, but most of them focus only on one format, such as MD5, and use minimal dictionaries to crack the hashes. Let's try one more demonstration.

SHA1

This format is slightly more secure than MD5, but still extremely easy to crack. The passwords obtained from the LinkedIn breach were all hashed in SHA1. Below are two examples.

174@gmail.com:403E35A2B0243D40400AF6BB358B5C546CDDD981
185@gmail.com:B1C4BBC4D7546529895CFABF8C1139CA7E486E18

The results from Hashes follow. These two passwords are identical, but with different case. The first is all lowercase while the second contains uppercase. Notice that these create a completely different hash because of this. Keep this in mind when investigating passwords of your target.

403E35A2B0243D40400AF6BB358B5C546CDDD981:letmein!
B1C4BBC4D7546529895CFABF8C1139CA7E486E18:LetMeIn!

The golden days of data breaches from 2017 through 2019 are over. Sure, those databases are still floating around and very valuable, but recent breaches do not typically possess MD5 or SHA1 password hashes. We often see much more secure hashing algorithms such as BCRYPT or SHA512. Again, cracking these passwords with dedicated rigs exceeds the scope of this book. Searching "older" hashes can be time consuming, but the results can be extremely beneficial. It is quite a commitment to download hundreds of gigabytes of data for this purpose. Take time to think about your own needs.

Online Hash Search Resources

The following websites will convert MD5 and SHA1 hashes into passwords in some scenarios. The password must be within their limited system and should not be very complex. These are very limited, but may offer immediate data.

https://osint.sh/md5/
https://www.md5online.org/md5-decrypt.html
https://md5decrypt.net/en/
https://md5decrypt.net/en/Sha1/
https://www.dcode.fr/sha1-hash
https://md5hashing.net/hash/sha1

If you do not want to build, store, and maintain your own hash data set, I recommend **Name That Hash** and **Search That Hash** (github.com/HashPals/) over the online options. If you followed the steps in Chapters Four and Five, your OSINT VM is already configured for this tutorial. If not, you can install it within your OSINT virtual machine with the following steps.

```
mkdir ~/Downloads/Programs/search-that-hash
cd ~/Downloads/Programs/search-that-hash
python3 -m venv search-that-hashEnvironment
source search-that-hashEnvironment/bin/activate
```

```
sudo pip install -U search-that-hash
deactivate
mkdir ~/Downloads/Programs/name-that-hash
cd ~/Downloads/Programs/name-that-hash
python3 -m venv name-that-hashEnvironment
source name-that-hashEnvironment/bin/activate
sudo pip install -U name-that-hash 2>/dev/null
deactivate
```

Within Terminal, you can now execute the following to search a hash within multiple online converters.

```
sth --text "5f4dcc3b5aa765d61d8327deb882cf99"
```

The result appears as follows.

```
5f4dcc3b5aa765d61d8327deb882cf99
Text : password     Type : MD5
```

You can also search for only the type with `nth --text "5f4dcc3b5aa765d61d8327deb882cf99"`. The result appears as follows.

```
Most Likely MD5, HC: 0 JtR: raw-md5 Summary: Used for Linux Shadow files.
```

We now know that the hash value is a MD5 representation of the password "password". This process is included within the script titled "Breaches/Leaks Tool" included in the OSINT VM. I launch this application daily. When it cannot identify the password, I rely on my Hashes.org data set. If all of this is simply too complicated, we can always rely on more online services, as explained soon.

SQL Files

Some old-fashioned Google queries might find more data breaches and leaks than one can manage. Let's conduct a few examples with SQL files. SQL, often pronounced "sequel", is an acronym for Structured Query Language. It is a computer language used in programming and designed for managing data held in a relational database management system. In other words, most SQL files are databases of some sort. Many of the most popular database breaches were originally released as SQL files. WordPress backups, web forum exports, and other online maintenance files are also stored in this format. Searching for public SQL files can reveal surprising results. Let's search Google for SQL files with the following commands using the tutorials previously discussed.

`ext:sql`: This returns millions of results. While some are files with the extension of ".sql", most of the results are web pages ending with ".sql", and are not helpful. Let's modify the search as follows.

`ext:sql "create table"`: This returns 26,300 results which include the exact terms "create table". This is a standard statement within SQL files which specifies the names of tables within the database. This filters most of the website names from our search and displays only valid SQL files. Next, let's add to our search with the following modification.

`ext:sql "create table" "@gmail.com"`: This returns 5,160 results. Each is an SQL database with at least one entry of "@gmail.com" inside. This indicates that active email addresses are within the files, which is indicative of a true breach or leak. The following search should reveal data worth analyzing.

`ext:sql "create table" "@gmail.com" "'password'"`: This returns 3,060 results. Each is an SQL database with at least one entry of "gmail.com" inside and a table titled "password". Note the single quotes around password within double quotes. This tells Google to search specifically for 'password' and not the word

alone. 3,060 results are overwhelming, and include a lot of test files stored on Github. Since many people use Gmail addresses as the administrator of test databases, we should add another email domain as follows.

```
ext:sql "create table" "gmail.com" "'password'" "@yahoo.com" -site:github.com:
```

This reveals 67 results. Each are SQL files which contain at least one Gmail address, one Yahoo address, and a table titled password. Furthermore, all results from Github are removed. Most of these results will load as text files within your browser. However, some will be quite large and may crash your browser while loading. The following is a modified excerpt from an actual result, which I found by searching "@gmail.com" within the browser text.

```
(62,'RichardWilson','admin','richard@redactedemail.com','4d5e02c3f251286d83750
40ea2b54e22','Administrator',0,1,25,'2008-05-28 07:07:08','2009-04-02
13:08:07')
```

This is a very typical structure within SQL files. The following explains each piece.

`62,` (User ID)
`'RichardWilson',` (Name provided)
`'admin',` (Username)
`'richard@redactedemail.com',` (Email address)
`'4d5e02c3f251286d8375040ea2b54e22',` (Hashed password)
`'Administrator',` (Usertype)
`0,1,25,` (Variable internal codes)
`'2008-05-28 07:07:08',` (Registration date)
`'2009-04-02 13:08:07',` (Last visit date)

You could save the entire page as a .txt file (right-click > Save page as…) for your personal data archive. An alternative query for text files is as follows.

ext:txt "create table" "gmail.com" "'password'" "yahoo.com" -site:github.com

There may be a trove of sensitive information within these files, so use caution and always be responsible. Exercise good defense when browsing any of these sites or downloading any data. Trackers, viruses, and overall malicious software are always present in this environment. Using a Linux virtual machine and a reputable VPN will provide serious protection from these threats. I search and store leaked and breached databases as a part of every investigation which I conduct. I can say without hesitation that these strategies are more beneficial than any other online investigation technique of which I know. Some investigators within my circles possess several terabytes of this data from tens of thousands of breaches and leaks. Querying your own offline archive during your next investigation, and identifying unique data associated with your target, can be extremely rewarding. Again, I ask you to be responsible. Never use any credentials to access an account and never allow any data obtained to be further distributed. Use this public data, stolen by criminals, to investigate other criminals.

Hopefully, you now have a basic understanding of breaches, passwords, hashes, database files, and how all of it can work together to be a part of your online investigations. You might have a copy of the COMB data set which includes over a billion email addresses and associated passwords, but that was created for convenience. You may have encountered some hashes online and can reveal passwords hidden beneath, but those will be based on specific target queries. While the data discussed so far is easily assessable, new breaches might not be so easy to find and digest. Let's start over with two breach examples from data recently stolen.

As stated previously, I am not allowed to provide direct links to any breach data. However, I am allowed to discuss ways in which some people use OSINT to find this data. Let's take a look at BTDig (btdig.com). This torrent search engine is known for "digging" deeper into dark places in order to find torrent files containing

stolen content. If you were to go to this site and search for "public_", you would likely be presented a Torrent file which contains 241 GB of data titled "MyCloud". Expanding the data should reveal 1,315 files, each of which are a breached database obtained between 2016 and 2021. This is far from everything breached during that time, but it may serve as an introduction into this type of content. Warning: the **compressed** content is 241 GB, so expect the decompressed to reach a terabyte. I plucked two small random examples from this data set in order to explain the data within.

DogForum.sk: The file titled "Dump_public_dogforum.sk_2021.01.13.rar" is a 2.3 MB compressed file which decompresses to a 12.5 MB SQL database dump. This site's database was allegedly stolen on January 13, 2021. Opening the SQL file within a text editor reveals a table titled "phpbb3_users". The following displays only the ten most interesting fields for my investigations within this table.

```
CREATE TABLE `phpbb3_users` (
  `user_id` mediumint(8) unsigned NOT NULL AUTO_INCREMENT,
  `user_ip` varchar(40) COLLATE utf8_bin NOT NULL DEFAULT '',
  `user_regdate` int(11) unsigned NOT NULL DEFAULT '0',
  `username` varchar(255) COLLATE utf8_bin NOT NULL DEFAULT '',
  `user_password` varchar(40) COLLATE utf8_bin NOT NULL DEFAULT '',
  `user_passchg` int(11) unsigned NOT NULL DEFAULT '0',
  `user_email` varchar(100) COLLATE utf8_bin NOT NULL DEFAULT '',
  `user_birthday` varchar(10) COLLATE utf8_bin NOT NULL DEFAULT '',
  `user_lastvisit` int(11) unsigned NOT NULL DEFAULT '0',
  `user_timezone` decimal(5,2) NOT NULL DEFAULT '0.00',
```

A redacted example of one user entry is below (there are over 17,000 users in this database).

```
(8419,'178.40.120.170',1275503507,'Ode','$H$9ftKyo59RKktw7Idj2TGaVweMH2ITG.',1
350132160,'xxxxblackjack@gmail.com',' 0- 0-   0',1371625109,+3,)
```

We can now assume the following about this user based on this data.

`8419`: The user number for this user on this forum.
`178.40.120.170`: The IP address of the user during registration.
`1275503507`: The date and time the user registered (in Unix time-June 2, 2010 at 18:31:47 GMT).
`Ode`: The username for this user.
`$H$9ftKyo59RKktw7Idj2TGaVweMH2ITG.` : The hashed password for this account.
`1350132160`: The date and time the user last changed the password (October 13, 2012 at 12:42:40 GMT).
`xxxxblackjack@gmail.com`: The email address of the user.
`0- 0-   0`: The date of birth of the user (none was provided).
`1371625109`: The date and time of the user's last visit before breach (June 19, 2013 at 06:58:29 GMT).
`+3`: The user's time zone (+3 GMT - Middle East).

Please take a moment to digest this example. This entry is only one of 17,000 within this single stolen breach. The Torrent mentioned possesses over 1,000 breaches, and there are tens of thousands of known breaches out there. There is the potential to collect this type of information from billions of accounts. I hope you see the possibilities for your own investigations. Every day, we digest dozens of breaches into our own system, which I will discuss later, and we see this type of result with almost every query we conduct. Sometimes, we receive this type of data from within several breaches associated with our single target email address. It is quite powerful. Let's conduct another example.

EscortReviews.com: The file titled "Dump_public_escortreviews.com_2021.01.27.rar" is a 51.7 MB compressed file which decompresses to a 223.3 MB SQL database dump. This site's database was allegedly stolen on January 27, 2021, and was publicly reported a month later. Opening the SQL file within a text editor reveals a table titled "user". The following displays only the most interesting fields for my investigations within this table.

```
CREATE TABLE `user` (
  `userid` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `username` varchar(100) NOT NULL DEFAULT '',
  `password` char(32) NOT NULL DEFAULT '',
  `passworddate` date NOT NULL DEFAULT '0000-00-00',
  `email` char(100) NOT NULL DEFAULT '',
  `homepage` char(100) NOT NULL DEFAULT '',
  `joindate` int(10) unsigned NOT NULL DEFAULT '0',
  `lastvisit` int(10) unsigned NOT NULL DEFAULT '0',
  `birthday` char(10) NOT NULL DEFAULT '',
  `ipaddress` char(15) NOT NULL DEFAULT '',
```

A fictitious example of one user entry is below (there are over 470,000 users in this database).

```
(201,bobforfun,65184286F793B5E70ACA525E63DAE54F,2011-06-12,bob_love@ymail.com,
bobforfun.com,1238309520,1307841316,06-02-1983,76.187.13.130,2)
```

We can now assume the following about this user based on this data.

201: The user number for this user on this forum.
Bobforfun: The username for this user on this forum.
65184286F793B5E70ACA525E63DAE54F: The hashed password for this account.
2011-06-12: The date the user last changed the password.
bob_love@ymail.com: The email address associated with this account.
bobforfun.com: The personal website associated with this account.
1238309520: The Unix date and time the user joined (March 29, 2009 at 06:52:00 GMT).
1307841316: The Unix date and time the user last accessed (June 12, 2011 at 01:15:16 GMT).
06-02-1983: The provided date of birth for this user.
76.187.13.130: The IP address of the user during registration.

This site was used heavily by escorts, pimps, and johns in order to promote prostitution. Anyone who conducts human trafficking investigations might benefit from this data on 470,000 registered members. The hash for this fictitious example is MD5. If you were to place that hash into our Search-That-Hash tool previously discussed, you would receive the following result.

```
65184286F793B5E70ACA525E63DAE54F
Most Likely
MD5, HC: 0 JtR: raw-md5 Summary: Used for Linux Shadow files.
Text : escorts
```

You would now know that this user was using a weak password of "escorts" for this site in 2011. These example files are fairly small, but there is still a lot of content inside them which is unnecessary. You could clean these files using the previous techniques, but that may be overkill for your needs. Since I collect terabytes of this data, I try to minimize every file as much as possible. Reconsider the "DogForum.sk" file from earlier. The following commands would extract only the columns we desire and remove the unnecessary tabs.

```
cut -d, -f1,6,7,8,10,11,13,15,16,29 dogforum.sk.sql > DogForum.txt
sed -i ''s/(crtl-v-tab)//g"  DogForum-Cleaned.txt
```

This action shrank the 12.5 MB file to 2.4 MB and compressed each line to fit nicely within Terminal. That may not seem like much, but it can make quite a difference when dealing with thousands of files. I want to stress that this is only an introduction into this data. The amount of breaches publicly available is overwhelming. I will offer additional search techniques within the next chapter, but I leave you with one more consideration here for your attempts to locate breach data.

Archive.org Breach Data

Archive.org possesses a surprising number of data dumps. As one example, conduct a search on their site for "nulled.io" and click the "nulled.io_database_dump" page. It should present a download option for the entire 10 GB database from the Nulled hacking forum, as uploaded by "ntgrg". If you were to click on this profile, you would likely find other similar uploads. You could spend weeks on Archive.org identifying breach data.

Password Commonality

If you encounter a password for your target, you may want to know if it was unique. As an example, I would assume my previous example of the password "escorts" is fairly common. I can confirm this through the Have I Been Pwned password site at haveibeenpwned.com/Passwords. Figure 31.01 (top) displays the result identifying 847 breaches containing that password. However, what if your target's password was "escorts!"? Surpassingly, there is only one reported breach which had this password, as seen in Figure 31.01 (bottom).

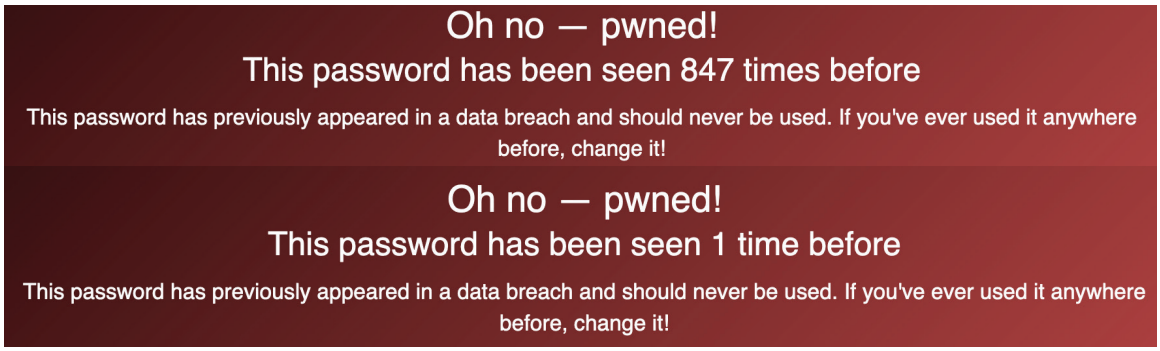


Figure 31.01: Password results from Have I Been Pwned.

Online Breach Search Resources

Throughout this book, I have mentioned various websites which allow query of target data against breached and leaked content. These do not always display passwords, but will confirm presence within a specific breach. I will not revisit each option, as they have all been previously explained, but I will provide a summary with direct URL submission options. This will be vital for our Data Tool mentioned at the end of the chapter.

Email Address (test@test.com)

https://haveibeenpwned.com/unifiedsearch/test@test.com
https://dehashed.com/search?query=test@test.com
https://psbdmp.ws/api/search/test@test.com
https://portal.spycloud.com/endpoint/enriched-stats/test@test.com
https://check.cybernews.com/chk/?lang=en_US&e=test@test.com
https://intelx.io/?s=test@test.com

Username (test)

https://haveibeenpwned.com/unifiedsearch/test
https://dehashed.com/search?query=test
https://psbdmp.ws/api/search/test

Domain (inteltechniques.com)

https://dehashed.com/search?query=inteltechniques.com
https://psbdmp.ws/api/search/inteltechniques.com
https://intelx.io/?s=inteltechniques.com

Telephone (6185551212)

https://dehashed.com/search?query=6185551212

IP Address (1.1.1.1)

https://dehashed.com/search?query=1.1.1.1
https://psbdmp.ws/api/search/1.1.1.1
https://intelx.io/?s=1.1.1.1

Name (Michael Bazzell)

https://dehashed.com/search?query=michael Bazzell
https://psbdmp.ws/api/search/Michael%20Bazzell

Password (password1234)

https://dehashed.com/search?query=password1234
https://psbdmp.ws/api/search/password1234
https://www.google.com/search?q=password1234

Hash (BDC87B9C894DA5168059E00EBFFB9077)

https://hash.ziggi.org/api/dehash.get?hash=BDC87B9C894DA5168059E00EBFFB9077&include_external_db
https://decrypt.tools/client-server/decrypt?type=md5&string=BDC87B9C894DA5168059E00EBFFB9077
https://md5.gromweb.com/?md5=BDC87B9C894DA5168059E00EBFFB9077
https://www.nitrxgen.net/md5db/BDC87B9C894DA5168059E00EBFFB9077
https://dehashed.com/search?query=BDC87B9C894DA5168059E00EBFFB9077
https://www.google.com/search?q=BDC87B9C894DA5168059E00EBFFB9077

Miscellaneous Sites

The following websites do not allow submission via URL, but a manual search may be beneficial. Please remember that all of these sites come and go quickly. By the time you read this, some of these services may have shut down. It is equally possible that new resources are waiting for your discovery.

LeakPeek (leakpeek.com)
Beach Directory (breachdirectory.org)

H8Mail (github.com/khast3x/h8mail)

H8Mail attempts to combine many of the breach services we have explored into one utility. It should never take the place of a full manual review, but the embedded automation can be beneficial to an investigation. If you followed the steps in Chapter Five, you already have this program and the automated script (Breaches/Leaks Tool) installed. If not, conduct the following within Terminal.

```
mkdir ~/Downloads/Programs/h8mail
cd ~/Downloads/Programs/h8mail
python3 -m venv h8mailEnvironment
source h8mailEnvironment/bin/activate
sudo pip install -U h8mail
deactivate
cd ~/Downloads
h8mail -g
sed -i 's/\;leak\-lookup\_pub/leak\-lookup\_pub/g' h8mail_config.ini
```

Launching the script with the following will produce minimal, if any, results.

```
h8mail -t bill@microsoft.com
```

Providing API keys from services such as Snusbase, WeLeakInfo, Leak-Lookup, HaveIBeenPwned, Emailrep, Dehashed, and Hunterio will provide MANY more results, but these services can be quite expensive. If you rely on breach data every day; can afford premium services; and do not want to collect your own breach data; there may be value for you within this option. After you have obtained API keys from your desired services, open the Files application, enter the Downloads folder, and double-click the file named "h8mail_config.ini". You should see text similar to the following partial example. Add your API keys within the appropriate lines, similar to the entry for "leak-lookup pub", and remove any semicolons within lines you want to be used. If a semicolon is at the beginning of a line, that option is ignored. At the minimum, make sure the semicolon is removed from the "leak-lookup pub" line and execute another query. You should see new results. If you conducted the "sed" command on the previous page, this should already be configured for you.

```
;;weleakinfo_priv =
;;weleakinfo_pub =
;hibp =
leak-lookup_pub = 1bf94ff907f68d511de9a610a6ff9263
;leak-lookup_priv =
;emailrep =
```

After this new configuration file modification, I executed a search for test@email.com. The result was a file which contained 172 results. The following partial view confirms that this address exists within breaches from TicketFly, TrueFire, and Tumblr. Eliminating this modification returned no results.

```
test@email.com,LEAKLOOKUP_PUB,ticketfly.com
test@email.com,LEAKLOOKUP_PUB,truefire.com
test@email.com,LEAKLOOKUP_PUB,tumblr.com
```

Sample Investigation

We have covered a lot so far within this chapter. Let's pause and conduct an investigation using this data. Assume that you possess the databases mentioned previously, especially "COMB". Your investigation identifies a suspect with an email address of johndoe1287@gmail.com. This appears to be a burner email account, as you find no evidence of it anywhere online. It was used to harass your client and likely created only for devious activity. Our first step is to query the address through all the databases you have acquired. Assume these are all stored on

your external SSD drive, which is plugged into your host computer. You have already connected this drive to a cloned OSINT virtual machine through the "Devices" menu in VirtualBox. Let's take each step slowly.

Open Terminal, type cd, then space, and drag and drop the external drive from your Files application into Terminal. This makes sure you are in the correct path. My command appears similar to the following.

```
cd '/media/osint/1TBExternal/'
```

I can now conduct a query within Terminal against all of my collected data. The following searches our target email address. Each query could take several minutes if you possess a lot of data and a slow drive.

```
rg -aFIN johndoe1287@gmail.com
```

This results in one entry as follows.

```
johndoe1287@gmail.com:H8teful0ne45
```

We now know that he used the password of H8teful0ne45 on a site. We should next conduct a search of that password to see if it is used anywhere else. The following query is appropriate.

```
rg -aFIN H8teful0ne45
```

This returned the following results.

```
johndoe1287@gmail.com:H8teful0ne45
johndoe@gmail.com:H8teful0ne45
johndoe1287@hotmail.com:H8teful0ne45
```

These addresses are likely controlled by our target since the passwords are the same and the addresses are similar. We now have new search options. However, this search only queries for this exact text password term. If you possess a database which has not been dehashed, your target password could be present within an MD5, SHA1, or other hash. Therefore, let's convert this password into the most commonly used hashes with the following websites, displaying the output below each.

https://passwordsgenerator.net/md5-hash-generator/
9EF0EC63E2E52320CB20E345DCBA8112

https://passwordsgenerator.net/sha1-hash-generator/
D15FB15C1BC88F4B7932FD29918D1E9E9BBE7CA5

https://passwordsgenerator.net/sha256-hash-generator/
37A790A268B9FE62B424BABFC3BCAB0646BFB24B93EC1619AAE7289E0D7086DB

We can now submit each of these throughout all of our data with the following three commands.

```
rg -aFIN 9EF0EC63E2E52320CB20E345DCBA8112
rg -aFIN D15FB15C1BC88F4B7932FD29918D1E9E9BBE7CA5
rg -aFIN 37A790A268B9FE62B424BABFC3BCAB0646BFB24B93EC1619AAE7289E0D7086DB
```

The first query returned the following result.

```
Leaks/1183_houstonastros-com_found_hash_algorithm_plain.txt.zip
SHA1 D15FB15C1BC88F4B7932FD29918D1E9E9BBE7CA5:H8teful0ne45
```

This tells us that a user with a password of "H8teful0ne45" was present on a breach about the Houston Astros. Is this the same person? It could be. It could also be a coincidence. The more unique a password is, the more confidence I have that it is the same individual. This definitely warrants further investigation. I might next try to locate the original breach data, which would likely include any email addresses associated with that password hash. All of these steps are designed to lead us to the next step. All of these results give me more confidence that these accounts are owned by the same person. The variant of the "hateful" password and presence of "johndoe" within the original email address and the new password convinces me we are on the right track. I would now target this new email address and replicate the searches mentioned within previous chapters. We should also check our Pastes search tool and the online breach resources previously explained.

As stated previously, your biggest frustration may be the speed of each query. If you are using an external spinning drive, expect your time to quadruple at best. If this technique proves to be valuable, you might consider a dedicated machine for this sole purpose. There is a lot to digest in this chapter. I want to state again that breach data has been the absolute biggest aid in my online investigations. I rely on it way more than Facebook, Twitter, and Instagram combined. At the end of Chapter Thirty-Three, I will offer additional storage and query techniques which should assist with immediate access to your data.

IntelTechniques Breaches & Leaks Tool

This final search tool combines most of the online search options mentioned throughout the chapter. The breach data resources are split into categories based on the target data (email, username, etc.). The last feature allows entry of any found password and immediately generates an MD5, SHA1, and SHA256 hash for further research. Figure 31.02 displays the online version at https://inteltechniques.com/tools/Breaches.html.

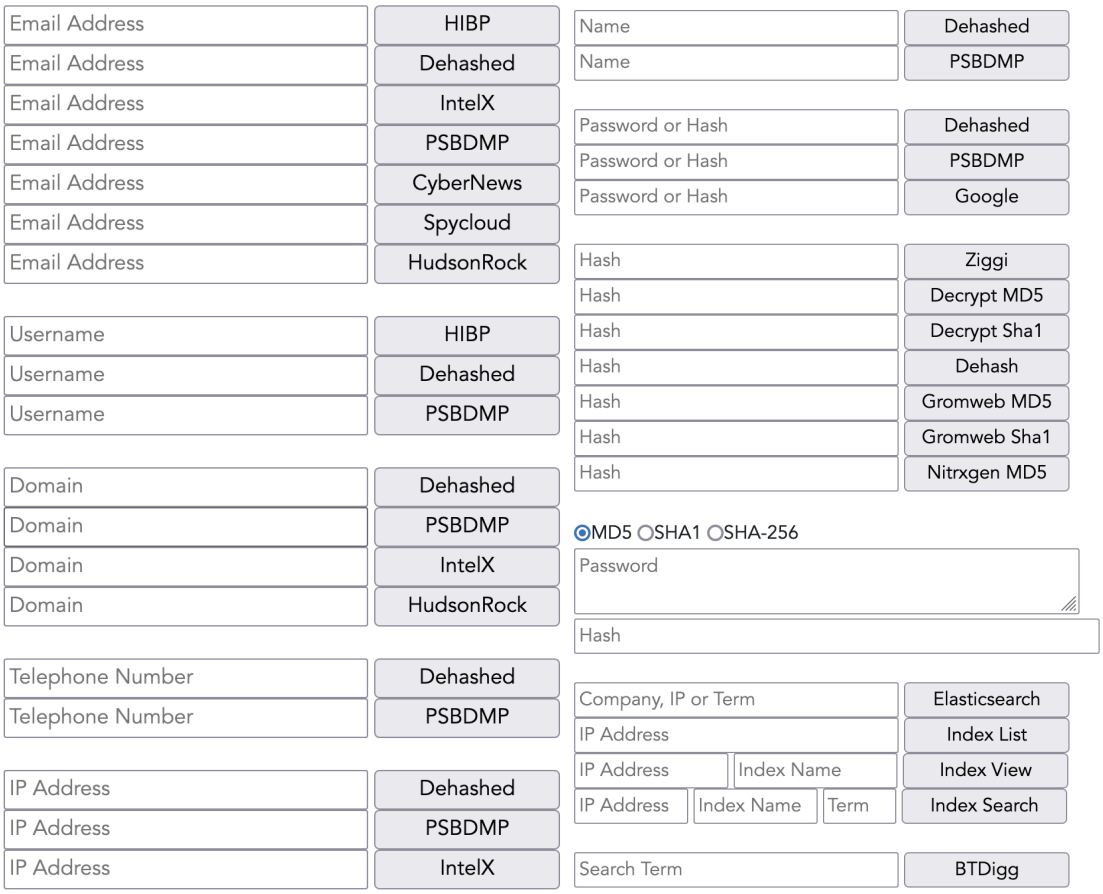


Figure 31.02: The IntelTechniques Breaches & Leaks Tool.

CHAPTER THIRTY-TWO
STEALER LOGS

I am absolutely fascinated by stealer log data. While I have always prioritized breach data as a vital part of our investigations, stealer log data presents a whole new world. First, let's revisit the basics.

Stealer logs are created once a malicious virus has been installed to a computer (typically Windows). The victim may be tricked into installing a program after visiting a malicious website, or the virus could be included within an unauthorized application, such as pirated software. The virus sniffs through the computer to identify, extract, and collect any valuable data. These "logs" are then transferred from the host machine and distributed within shady online networks.

The most common stealer logs we find are labeled as Raccoon Stealer, Redline Stealer, and Vidar Stealer. The groups which offer data stolen via these programs is limitless. A new group pops up every week. Criminal marketplaces trade this data as a commodity. They use the stolen data to unlawfully access online accounts, steal cryptocurrency, make unauthorized purchases, and wreak havoc on innocent people's digital lives. Our systems ingest over one million logs every day which are being shared online.

Let's take a look at some real data. In the following example, I extracted a random log which was generated by the Redline Stealer and uploaded to a criminal marketplace. I redacted much of the content. First, let's understand the file structure. The following is a tree of a single log file for one victim.

```
└── US[xx4909xC4Ex3C4x008x57AxD60BBCAF6] [2022-06-19T08_49_35]
├── -----000.txt
├── 00000000000.jpg
├── Autofills
│   ├── Google_[Chrome]_Default.txt
│   └── Microsoft_[Edge]_Default.txt
├── Cookies
│   ├── Google_[Chrome]_Default Network.txt
│   └──Microsoft_[Edge]_Default Network.txt
├── DomainDetects.txt
├── ImportantAutofills.txt
├── InstalledBrowsers.txt
├── InstalledSoftware.txt
├── Passwords.txt
├── Screenshot.jpg
└── UserInformation.txt
```

Now, let's walk through each folder and file.

```
└── US[xx4909xC4Ex3C4x008x57AxD60BBCAF6] [2022-06-19T08_49_35]
```

This top folder presents a two-letter abbreviation of the country of the victim (United States) followed by a unique Hardware Identifier (HWID) and the date and time of capture. The HWID is a security measure used by Microsoft upon the activation of Windows. This unique HWID is generated when the operating system is first installed. This will be vital in a moment. The date and time allow us to know the likely accuracy of the data.

```
├── -----000.txt
```

This is an information file about the thief. It often includes generic contact information, pricing for stolen data, and online communities associated with the product.

```
├── 00000000000.jpg
```

This is the logo of the Redline product.

```
├── Autofills
│   ├── Google_[Chrome]_Default.txt
│   └── Microsoft_[Edge]_Default.txt
```

This is where things get interesting. Your browser has likely asked you if you would like to store information which was entered into an online form. This could include your name, address, email, or other unique detail which gets entered into websites often. If you allow your browser to store this data, stealer logs easily collect it into their systems. Below is partial (redacted) data extracted from an actual victim. The original logs clearly displayed all data.

```
Name: email
Value: loxxx@gmail.com
===============
Name: username
Value: skxxxx22
===============
Name: lastname
Value: Loxxx
===============
Name: first-name-field
Value: Alixxx
===============
Name: address
Value: 20xx xxxx Lane
===============
Name: city
Value: Moxxx
===============
Name: phone
Value: 209-xxx-xxx
===============
Name: dob
Value: 10/xx/20xx
===============
Name: VIN
Value: kmhtcxxxxxxxxx
===============
Name: keyword
Value: 20xx Subaru xxxx
===============
Name: card-name
Value: Alixxx Loxxx
===============
Name: expiration-date
Value: 0x/2x
===============
```

I now have the name, DOB, home address, email, cell, vehicle, and partial credit card details of the victim.

```
├── Cookies
│   ├── Google_[Chrome]_Default Network.txt
│   └──Microsoft_[Edge]_Default Network.txt
```

Your browser stores temporary internet files about your credentialed sessions. I may not have your password to your email account, but possessing the cookies from your browser could allow me to steal your credentialed session and replicate access to your account. The following details could be beneficial to a criminal (obviously redacted and abbreviated).

```
.paypal.com TRUE / FALSE 196941xxx cookie_check yes
.paypal.com TRUE / FALSE 168533xxx cookie_prefs P%3D1%2CF%3D1%2Ctype%3Dimplicit
.paypal.com TRUE / FALSE 196941xxx d_id 9ebfcxxxe8545ae9a39xxx2d228xxx116537xxx
.paypal.com TRUE / FALSE 1969xxxG8 KN2xxx0aJZzhbL_R4HkiO_kb5_yMTkUPrF-Ml6xxx
.paypal.com TRUE / FALSE 1685334379 X-PP-ADxxxYsNaAuNxxxHBuQ9dI
.paypal.com TRUE / FALSE 1716870381 _ga GA1.2.137xxx.165379xxx
.paypal.com TRUE / FALSE 1716912230 login_email lopxxxgmail.com
.paypal.com TRUE / FALSE 1969417578 rmuc KhGxxxmVv_x1Oo9gQ7axxxUk
```

```
├── DomainDetects.txt
```

This file offers immediate access to the priority domains which exist in the overall record. This allows criminals to quickly identify logs of interest.

```
PDD: [Amazon] amazon.com (2), [Games] steamcommunity.com (2)
CDD: [PayPal] paypal.com (40), [Amazon] amazon.com (14), [Games] battle.net
(11), [Games] epicgames.com (1), [Games] steamcommunity.com (8)
```

```
├── ImportantAutofills.txt
```

This file parses data from the stored form fields which will be most beneficial to a criminal. The last two lines of my example suspect appear as follows. I also see full credit card details presented here often.

```
dob: 06/xx/xx
ssn: 248xx2xxx
```

```
├── InstalledBrowsers.txt
```

This file identifies all installed browsers and versions.

```
1) Name: Google Chrome, Path: C:\Program Files
(x86)\Google\Chrome\Application\chrome.exe, Version: 102.0.5005.115
2) Name: Internet Explorer, Path: C:\Program Files\Internet
Explorer\iexplore.exe, Version: 11.00.22000.1 (WinBuild.160101.0800)
3) Name: Microsoft Edge, Path: C:\Program Files
(x86)\Microsoft\Edge\Application\msedge.exe, Version: 102.0.1245.44
```

```
├── InstalledSoftware.txt
```

This file presents all applications installed within the machine. While this may not be extremely valuable to a criminal, it is gold to an investigator. If my target possesses a stealer program, I get to monitor a lot of details about the person's computer usage. In the following example, I would know which VPN my target uses, hardware details, and preferred games. That could lead to quite a social engineering attack.

```
1) Adobe Acrobat Reader DC [22.001.20117]
2) Adobe Creative Cloud [5.5.0.617]
3) Adobe Genuine Service [7.7.0.35]
```

```
4) Adobe Photoshop 2021 [22.1.1.138]
5) Adobe Refresh Manager [1.8.0]
6) Epic Games Launcher [1.1.279.0]
7) ExpressVPN [7.7.12.4]
8) ExpressVPN [7.7.12.4]
9) Google Chrome [102.0.5005.115]
10) HP Audio Switch [1.0.179.0]
11) HP Connection Optimizer [2.0.17.0]
12) HP PC Hardware Diagnostics UEFI [7.6.2.0]
13) Intel(R) Chipset Device Software [10.1.18295.8201]
14) Launcher Prerequisites (x64) [1.0.0.0]
15) LOOT version 0.16.0 [0.16.0]
16) McAfee LiveSafe [16.0 R27]
38) Minecraft Launcher [1.0.0.0]
39) NVIDIA Texture Tools Exporter for Adobe Photoshop [2020.1.3]
40) Razer Synapse [3.7.0531.052416]
41) Red Dead Redemption 2 [1.0.1436.31]
42) Rockstar Games Launcher [1.0.59.842]
43) Rockstar Games Social Club [2.1.3.7]
44) Steam [2.10.91.91]
45) UE4 Prerequisites (x64) [1.0.14.0]

├── Passwords.txt
```

This file presents all of the passwords stored within all browsers. This is why it is so important to only use reputable password managers, and never the native browser password storage option. Below is one of 56 examples for this victim, redacted. The original file displays all passwords in full.

```
URL: https://www.amazon.com/ap/signin
Username: xxxxxxx
Password: xxxxxxx

├── Screenshot.jpg
```

This is one of the most interesting pieces. It is a screen capture of the victim's machine at the time of infection. Figure 32.01 is an actual example (slightly redacted) which identifies the person's video interests, TikTok favorites, toolbar shortcuts, Google avatar (redacted), and an overall state of the computer at the time. This is quite invasive and can be a great lead in the investigation.

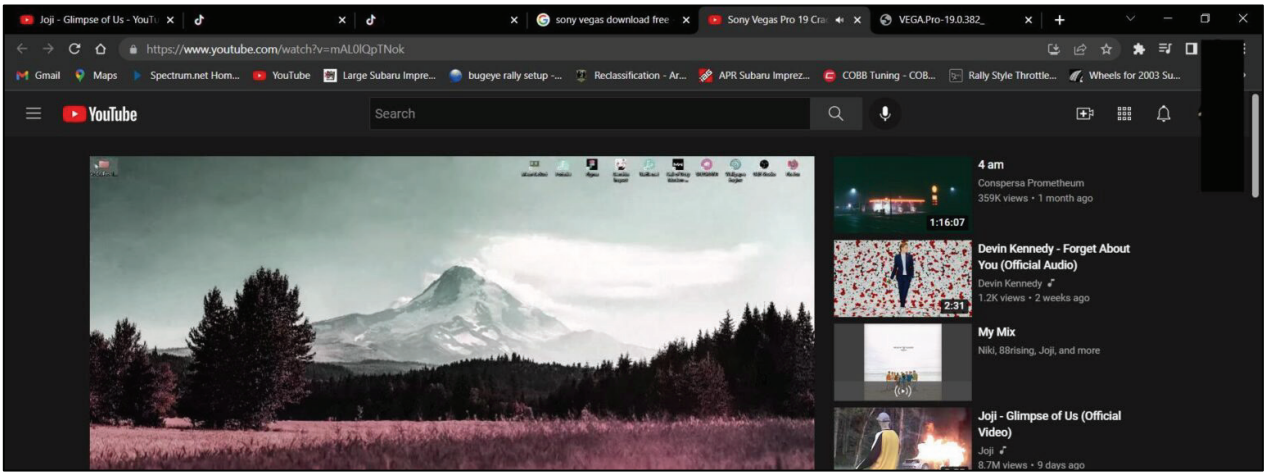


Figure 32.01: A Stealer Logs victim screen capture.

```
└── UserInformation.txt
```

The last file displays general details about the system, including the victim's IP address, hardware, location, and date. An example is below.

```
Build ID: REDLINEVIP
IP: 192.xx.xx.xx
FileLocation: C:\Windows\Microsoft.NET\Framework\v4.0.30319\InstallUtil.exe
UserName: xxx
Country: US
Zip Code: xxx
Location: xxx, Texas
HWID: 72xxxxxxxxxxxxxxCAF6
Current Language: English (United States)
ScreenSize: {Width=1920, Height=1080}
TimeZone: (UTC-07:00) Mountain Time (US & Canada)
Operation System: Windows 10 Home x64
UAC: AllowAll
Process Elevation: False
Log date: 19.06.2022 8:49:35
Name: Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz, 6 Cores
Name: NVIDIA GeForce RTX 2060, 4293918720 bytes
Name: Total of RAM, 12126.75 MB or 12715814912 bytes
```

You may be wondering why I get so excited about these logs when they expose sensitive details about innocent victims. I have two main reasons. First, it helps us defend our clients. We have had over a dozen clients who unknowingly installed a stealer virus. Our systems caught the infection within a few days and allowed us to contact the client to make notification. We have helped numerous organizations which had unknown infections within their network. Some of our staff calls this our "Pre-Crime Unit".

Next, it could be a priceless investigation tool. I have had numerous suspects within my own investigations who were victims of stealer logs. When this happens, I can see all of their email addresses, usernames, passwords, and computer information. This has revealed the true owner of many alias names and other deceitful tactics used by the suspect. If my target is the victim of stealer logs, my entire investigation is about to be wrapped up quickly. If I have the HWID, I can search through our troves of logger data to identify even more vital info about the suspect.

Because of this, we aggressively collect stealer log data every day. Some days we ingest over 250 GB of this data. The following Google queries might present interesting information, but most results lead to shady criminal marketplaces which require an account to see download links. Use extreme caution here, and I will present more options in a moment.

"stealer logs" "download"
"stealer logs" "Azorult"
"stealer logs" "Vidar"
"stealer logs" "Redline"
"stealer logs" "Raccoon"

It is now time to dive into some very shady criminal communities. Ensure that your organization's policies allow this behavior; never publish or disseminate anything you find; understand any local, state, or federal laws which could disallow these actions; and always focus on the good usage of this data. Never attempt to log in to anyone's account and make sure your computer is well protected from online threats. The following is presented as an educational demonstration, and I take no liability for your own actions taken.

Telegram (telegram.org)

Telegram is a cross-platform and cloud-based instant messaging service which provides group chat, optional end-to-end encrypted chats, VOIP calling, file sharing, and several other features. One must install the Desktop application in order to access all of the features. It has been scrutinized as a tool used by drug dealers, burglars, terrorists, child abusers, and many fringe groups. The software and services were developed, and are currently maintained, by a Russian company. I would never install this software on any personal or OSINT investigations machine. I only execute it on my dedicated leaks, breaches, logs, and ransomware desktop. Of any technique in this book, the following tutorial is the most likely to infect a machine with a virus. You have been warned.

However, I believe Telegram is quite possibly the best resource for Stealer Logs. Within seconds, you have access to terabytes of data without the need for links to third-party download providers which enforce slow download speeds for free accounts. We can download most of the data directly from Telegram servers at the full bandwidth of our internet connection. Let's start with a demonstration. First, you must install the Telegram desktop application. The official website offers direct downloads for Windows and macOS users. Linux users should execute the following command.

```
sudo apt install telegram-desktop
```

There are dozens of Telegram channels which offer free stealer logs in order to promote their paid services. I never recommend paying for any type of stolen data, but I will eagerly download and analyze any content they are sharing via their promotional channels. Ruban Cloud is one of these services. If you were to search "Ruban Cloud Free" in Google, you will likely only see a few BreachForums posts offering logs, which have since been removed. If you were to search "rubancloudfree", you might see additional posts which want payment in order to download the data. However, if you were to navigate directly to the "rubancloudfree" Telegram channel, you might find an overwhelming amount of free stealer logs. We next need to understand Telegram URLs. Telegram links might appear as the following fictious example.

https://t.me/inteltechniques

If you navigate to a URL like this, your browser will likely prompt you to either choose an application or cancel the request to open an external program. If you have Telegram installed, you can select it and allow your browser to forward any channel links directly to Telegram Desktop. If you do not have Telegram installed, many channels will allow you to see a preview of the content. The URL for this browser-based preview would appear as follows.

https://t.me/s/inteltechniques

In these two examples, I had my name (inteltechniques) in the URL. You would replace inteltechniques with whatever channel you are seeking. Remember, I cannot provide any direct hyperlinks to any stolen data, but you should be able to adapt my demonstration in order to access a room such as "rubancloudfree". If you were to find this, you would see a page within your browser similar to Figure 32.02. Notice this channel is promoting purchase of access to their logs collection, but also offering a compressed 967.2 MB sample for free. Figure 32.03 (left) displays how this would appear. In Figure 32.02, clicking the file icon next to this offering presents a request to open Telegram. Clicking the "down arrow" next to this file within Telegram begins the download.

After download, I possessed a 1 GB file titled " Ruban Private @rubanowner.rar". Extracting the compressed file resulted in a new folder titled "Ruban Private", which was 1.48 GB in size. Inside this folder was 1,101 folders, each titled similarly to "Ruban Private rubancloud (1).0025409]". Each of these folders possessed the folder structure as seen in Figure 32.03 (right). The "Autofills", "Cookies", "Passwords", and "Browsers" files contain the information as previously explained. The "FileGrabber" folder contains all documents stored within the default "Documents" and "Desktop" directories of the victim. These are typically full of Microsoft Word documents and PDFs. In this one sample file, we now have the passwords, documents, and other sensitive details of 1,101 victims. While this is scary and awful, it is also powerful as an investigative tool.

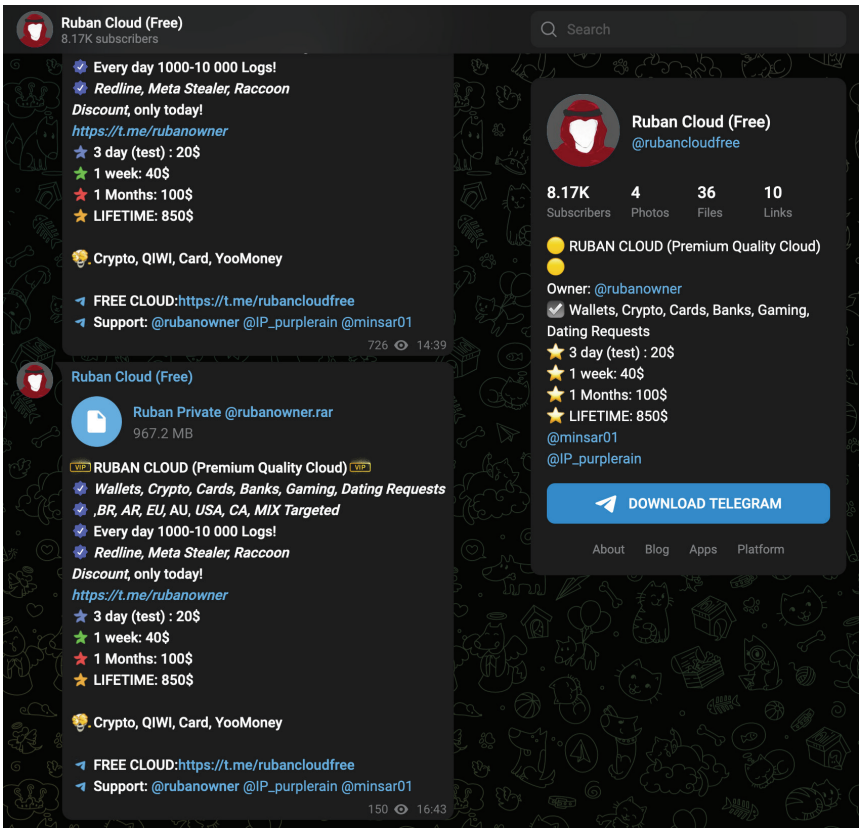


Figure 32.02: A Telegram channel preview as seen via a web browser.

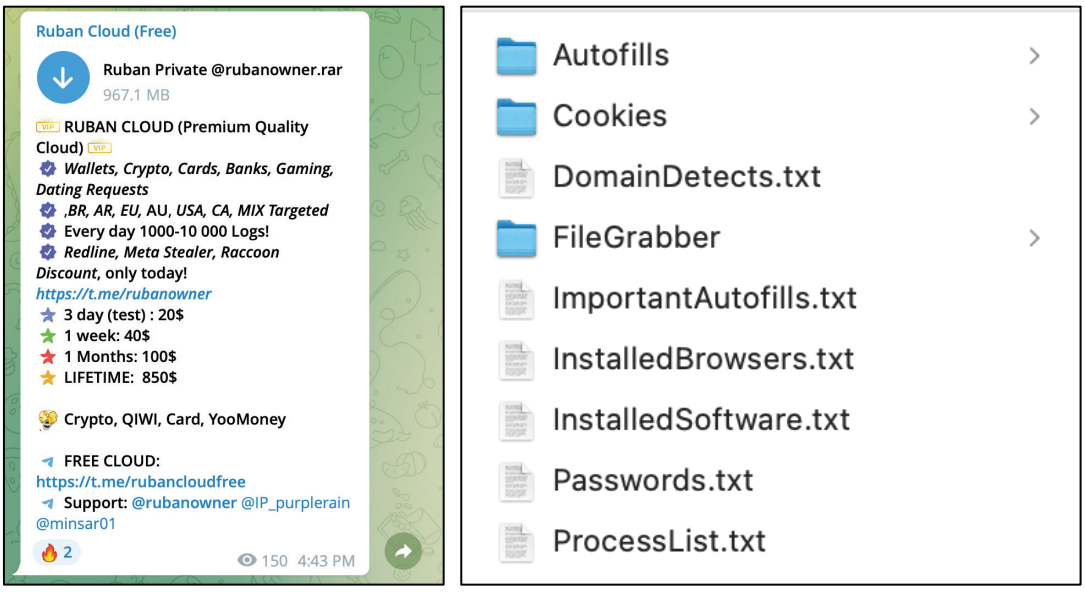


Figure 32.03: A Telegram post (left) and folder structure of entries (right).

Another Telegram channel is called "redlogscloud. Figure 32.04 (left) displays two recent files containing over 1,000 stealer logs. After download, I possessed a 353 MB compressed file titled "530 PCS 16 OCTOBER @REDLOGSCLOUD budget.rar" and a 485 MB compressed file called "530 PCS 16 OCTOBER @REDLOGSCLOUD.rar". Extracting these compressed files resulted in a new folder titled "530 PCS - 16 OCTOBER @REDLOGSCLOUD", which was 658 MB in size and new folder titled "530 PCS - 16

OCTOBER @REDLOGSCLOUD budget", which was 500 MB in size. Inside these folders was 1,063 folders, each titled similarly to " US[D51CAB8179F4CBFDEF8ED5FAF20432C5] [2022-10-15T21_57_48.5469259] ". The "US" is reference to the country of the victim.

Each of these folders possessed the folder structure as seen in Figure 32.04 (right). The files are very similar to the previous example. However, this specific stealer also grabs a screen capture of the victim's computer since it is relying on the "Redine" software. In this example file, we now have the passwords, documents, and other sensitive details of 1,063 victims. Combined with the previous example, we have the details of 2,164 people.

That may not seem like much. There are hundreds of millions of people in the U.S. alone, why would we care about such a small quantity of users? That takes us into the next consideration. We have only tapped into the most recent daily posts of two Telegram channels. As I write this, over a dozen new stealer log Telegram channels have posted new content. Today, I will see at least 50 GB of compressed log data be posted to the rooms I follow. Downloading all of it would probably expose another 50,000 victims. If you do this every day, you have the potential of downloading 1.5 million logs a month. Combined with other sources, you might obtain a few million per month. Do you see where this is going?

If you plan to only dabble in this type of data, these tutorials serve as an example to the threats and investigative benefits of stealer logs. The only way you could truly benefit from them within your own investigations is to download everything you find daily. That is quite a burden. I hired a full-time employee to do just that. He constantly acquires new leaks, breaches, logs, and ransomware, and then imports into our custom database for use by my staff. That comes with quite an expense, but we see results which justifies the activity.

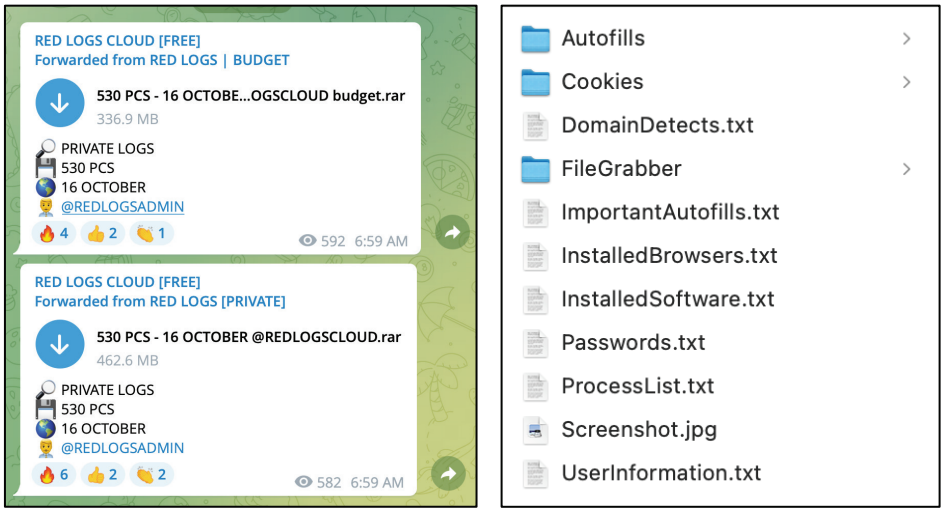


Figure 32.04: Telegram posts (left) and file structure (right).

Real World Applications: Once you have built a massive collection of these logs, you can use Ripgrep to query them as previously mentioned. In 2021, I was investigating an unknown person harassing one of my clients. He was using a throwaway email address which seemed impossible to trace. It was not present within any breach data. However, it appeared within my stealer logs, which included a device name similar to Desktop-u3ty6. Searching that device identifier presented dozens of email addresses and passwords in use on that machine. This quickly revealed my suspect, a 15-year-old kid. Further investigation confirmed his computer became infected after downloading a pirated version of anti-virus software. The irony. One week prior to writing this, I was investigating a financial account takeover attempt on behalf of a client. Someone switched her bank account to a "burner" email address. This criminal apparently possessed a stealer virus, because searching that email address immediately hit upon the entire logs for his computer. This included his "FileGrabber" folder which contained a recently modified resume. I was able to provide a full dossier on the suspect to my client, which was quite rewarding. The entire investigation took less than five minutes.

I suspect my attorney will scold me if I continue to provide more examples. Instead, I will leave you with a list of known Telegram stealer log rooms for you to conduct your own research. These include Logs Cloud, Keeper Cloud, Luffich Cloud, Crypton Logs, Bugatti Cloud, Ruban Cloud, Luxury Logs, Wild Logs, Eternal Logs, Logs Arthouse, HUBLOGS, Redlogs Cloud, OnelOgs, CBank, Observer Cloud, Expert Logs, BradMax, and Cloud Logs. This is only a sample, and there are many others which emerge and disappear daily. Start with Google.

I offer one final recommendation for Telegram: take advantage of the search field. As I wrote this, I noticed an announcement about a new breach at a cryptocurrency provider. Searching the name of the service within Telegram immediately presented numerous free download options. Follow as many breach-related channels in order to increase search results of interesting data.

You can also search sites such as BreachForums for logs, as a lot of members post direct download links which do not require purchased or earned credits. The following query would identify posts on BreachForums which include a link to Mega with the keywords "stealer log". The result is currently 1,260 results.

site:breached.vc "mega.nz" "stealer log"

The following query would identify posts on BreachForums which include a link to AnonFiles with the keywords "stealer log". The result is currently 1,410 results.

site:breached.vc "anonfiles.com" "stealer log"

You may also have success with the following queries which are not limited to BreachForums.

site:anonfiles.com "logs"
site:mediafire.com "logs"

I executed the first option and received the results visible in Figure 32.05. The second result intrigued me. It appeared to be 2.1 GB compressed file of logs, and only "part1". I modified my query to the following.

site:anonfiles.com "Stealer Logs.part2"

I received the result seen in Figure 32.06. I repeated this until I obtained all four parts of this 7 GB compressed file which contained thousands of stealer logs.

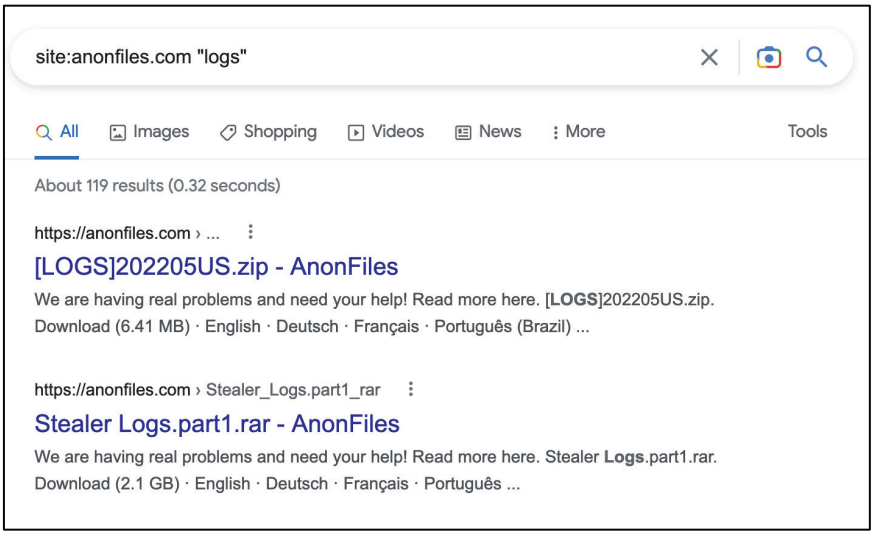


Figure 32.05: A Google result for stealer logs.

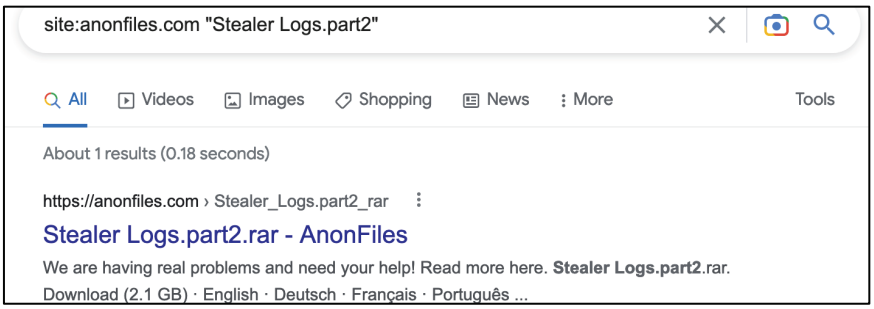


Figure 32.06: A modified Google result for stealer logs.

Telegram Leaks and Breaches

Telegram should not only be used as a tool to research stealer logs. It is also quite popular with groups who trade data leaks and breaches. As I was writing this chapter, I noticed a channel called "latest_leaks" posted the entire Verifications.io breach mentioned in Chapter Thirty.

Criticism

I know what some readers are probably thinking. You may believe that sharing these types of details is irresponsible and will lead to further abuse of the victims of these attacks. I respect your opinion and understand your concern. I also believe that thinking is naive. The criminals who are going to abuse this data do not need a tutorial from me. If anything, they would laugh at my basic understanding and explanation of their world. I am sure they could teach me better ways of doing all of this. I do not believe anyone is going to buy this book and begin further abusing this type of data.

Hardware stores sell hammers; websites sell lock picks; and local stores sell guns. All of these can (and are) abused. That does not mean that they should be restricted from those who are responsible with the items. I would rather take the chance that one person will misuse this information if it helps a thousand people use it for good. The more analysts and researchers who understand and collect this data, the better we will all be. If we had enough large companies monitoring and acting upon this data, we may see it go away because it is no longer useful to the criminals. Ignoring these techniques allows the criminals to continue their games and win. If I was the CEO of PayPal, I would dedicate employees to this task; have them identify any presence of PayPal credentials; and disable them before they are abused. You could replace PayPal with any other financial institution in those statements.

There are books about detailed computer hacking, malware creation, and various "black hat" lessons. They are used by security researchers, penetration testers, and other professionals who are tasked with responding to bad situations. My hope is that this book can help those who can use this data for good (defense and offense). I would have loved to have all of this when I was investigating cyber-crime for the FBI, but they probably would not have allowed me to use it. We can either pretend there is not a problem or we can take action to stop some of this mess. We can either allow criminals to have all of the power with this data or we can use it against them. I have countless examples of criminals accidentally infecting their own machines and exposing their own data.

Cleansing

Stealer logs are large. I estimate we have over 20 TB of stealer log data. That is not only a lot of data to store, it can take a long time for Ripgrep to query. At my company, we place everything within a custom database, as explained later, but that is not feasible for most people. While writing this chapter, I downloaded over 25 compressed files which contained over 30 GB of data including over 37,000 victim logs. That was just as an afternoon demonstration. Imagine how quickly this would pile up if you do it every day. Because of this, I offer an option for those without the need for all of the data included within a typical log.

Most people find the email addresses, usernames, and passwords to be the most valuable part of this data. While the screen captures, documents, and cookies are beneficial, they take up the most space. If desired, you could just keep all of the password files and delete the rest. Some fellow data enthusiasts just cringed at that, but I want to be realistic. Searching through millions of files in order to locate the email addresses and passwords of a suspect may be overkill for some. Instead of saving the password files, let's merge them all into one file. Imagine you have thousands or millions of stealer log files within your Downloads folder. The following Terminal command would navigate to each "passwords.txt" file; extract all of the text data; and compile it into one file titled Passwords.txt. Make sure you are in the right folder within Terminal.

```
find . 'passwords.txt' -exec cat {} \; > ~/Downloads/Passwords.txt
```

However, we have a problem. Some stealer software titles the passwords file "Passwords.txt" with an uppercase "P". Therefore, the following command is better. It ignores the case of the file and would extract data from both "passwords.txt" and "Passwords.txt". You could also use 'passwords*' to pick up all files beginning with "passwords".

```
find . -iname 'passwords.txt' -exec cat {} \; > ~/Downloads/Passwords.txt
```

If desired, you could replicate this for the " ImportantAutofills" files with the following.

```
find . -iname 'importantautofills.txt' -exec cat {} \; >
~/Downloads/ImportantAutofills.txt
```

I executed both commands against my test data, which produced a 200 MB Passwords.txt file and a 20 MB ImportantAutofills.txt file. These could be queried quite easily. If I had a terabyte of logs, my passwords file might be only 8 GB. While we keep all of the data we acquire, we also create an ever-growing Passwords.txt file which can be quickly queried for times we may not have immediate access to our entire database. If you were to execute the following, the date would be appended to your new file, such as Passwords.2022-11-28.txt.

```
timestamp=$(date +%Y-%m-%d) && find . -iname 'passwords.txt' -exec cat {}
\; > ~/Downloads/Passwords.$timestamp.txt
```

Finally, as a bonus to all macOS users, the following script would create a timestamp; switch to the Telegram download directory; use Keka (`brew install keka`) to decompress ONLY the files which include "passwords.txt" without case sensitivity (`-ssc-`); create new files instead of overwriting if there is a conflict (`-aou`); combine all of the data into one file; and add the date at the end of the final filename. This prevents us from decompressing data we might never use, and is much faster. Consider the tutorials from Chapters Four and Six to replicate your own script for this purpose, modifying the details as desired. I run mine daily, and these commands are all in the data.txt file within your online portal as previously explained. If a password is required for the archive, you could add `-p$pass`, replacing "pass" with the password, to the third command.

```
timestamp=$( date +%Y-%m-%d-%H-%M)
cd ~/Downloads/Telegram\ Desktop
for file in *; do /Applications/Keka.app/Contents/MacOS/Keka --ignore-file-
access -x --cli 7z x "$file" -aou -ssc- '*passwords*' -r; done
find . -iname '*passwords*' -exec cat {} \; >
~/Downloads/Passwords.$timestamp.txt
```

I suspect some readers are overwhelmed at the amount of data available for investigative use. I get it. I am overwhelmed every day. However, I believe that is part of the fun. We have one more chapter to go in this section, and it will possibly introduce more data than the past three chapters combined. At the end, I present some options for data management, including a script which performs all commands on your behalf.

CHAPTER THIRTY-THREE
RANSOMWARE

If you thought I was pushing the boundaries of ethical data acquisition, the following might make you uncomfortable. You have likely heard about ransomware infection. It is the illegal activity by criminals which steals data from a company, encrypts all of their files, and demands a ransom to gain access to the unusable data remaining on their own servers. When companies began creating better backups which eliminated the need to pay the ransom, the criminals took a new route. If the victims do not pay, all of their data is uploaded to the internet via Tor websites for anyone to download. This generates terabytes of private documents, email, databases, and every other imaginable digital file. Is this OSINT data? I don't think so. However, an investigator working on behalf of a victim company should know where to find this information.

Most sites which announce and distribute ransomware will require the Tor Browser previously explained. However, we will also retrieve recent data posted within clear-source options such as Telegram. Let's start with a demonstration using the Tor browser. The first step is to identify the current URLs for the various ransomware groups. I present the following three resources.

http://ransomwr3tsydeii4q43vazm7wofla5ujdajquitomtd47cxjtfgwyyd.onion/
https://github.com/fastfire/deepdarkCTI/blob/main/ransomware_gang.md
https://darkfeed.io/ransomgroups/

The first requires access to the Tor network, but the second and third are available openly within any browser. These sites attempt to provide a current status of various criminal groups; direct links to otherwise hidden sites offering data; and identify sites which have gone offline. Figure 33.01 displays the current view of the first option, visible through the Tor Browser.

Group Name	Onion V.	Link
Arvin Club	v3	Open
Babuk	v3	Open
Black Basta	v3	Open
AlphaVM/BlackCat	v3	Open
BlackByte	v3	Open
Bl4ckt0r	v3	Open
CL0P	v3	Open
CONTI	v3	Open

Figure 33.01: A list of Ransomware sites.

Next, Google and the search tutorials presented at the beginning of the book should help you find any other pages valuable to ransomware investigations. Queries for the following should assist. The ".onion" tells Google you want Tor-based URLs; the name of the group is followed in quotes; "ransomware" identifies the topic; and "url" helps Google focus on pages with URLs within the content.

".onion" "Ragnar" "ransomware" "url"
".onion" "REvil" "ransomware" "url"
".onion" "Conti" "ransomware" "url"
".onion" "Vice Society" "ransomware" "url"
".onion" "Clop" "ransomware" "url"
".onion" "Nefilim" "ransomware" "url"
".onion" "Everest" "ransomware" "url"

These should present information which will connect you through the best URLs. However, we have one other more direct option. The following Google search presents pages within the website Hack Notice which announce ransomware publications.

site:https://app.hacknotice.com "onion"

Opening these pages displays a notice of ransomware intrusion. Clicking the title of the article presents the Tor URL which may display the stolen data. Clicking "View Original Source" on this new page will attempt to open the Tor URL in your browser. I have discovered terabytes of data this way. Now that you have multiple methods to identify ransomware data dumps, let's focus on two demonstrations with actual data.

Within the first Tor-based resource previously mentioned, I clicked the Everest link to navigate to the official Everest ransomware distribution page. I then saw dozens of victims listed within their page. I will not identify any victims throughout this chapter, and will focus only on the data and type of organization. I clicked on the first victim, which was a non-profit health care organization which was hit with ransomware in October 2022. The page offered three links hosted at gofile.io, each approximately 30 GB compressed. Figure 33.02 displays one of the download options. Notice we can see the number of successful downloads. This could be vital if you are investigating this breach on behalf of the company. This confirms people are actively downloading the stolen data from this group.

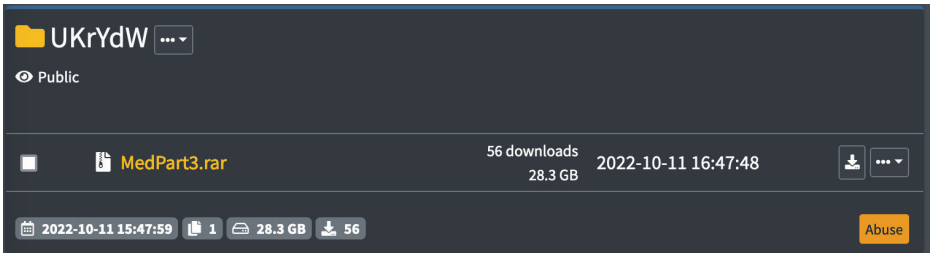


Figure 33.02: A ransomware download page.

If you were to download these files, you would have 120 GB of data which was stolen during the breach. Most of the data is of no use to an online investigator. We intentionally destroy anything which contains intellectual property or internal housekeeping affairs. The breach provided over 37,000 medical documents including patient records. This breach also included many copies of driver's licenses (front and back), insurance cards, and other forms of identification. This data is quite invasive and you may feel uncomfortable accessing any of it. I completely understand. We digest it into our system in order to make immediate notifications to any of our clients which are present within the breach. We only use this data for defensive practices, and mark all medical data to be ignored when staff conducts OSINT queries into the data.

Recently, we took in a large ransomware data set which included scans of the front and back of thousands of customers' driver's licenses. Our systems analyze all images, documents, and PDFs, and extract any text content via Optical Character Recognition (OCR). This allows us to query or monitor our clients' details, which can notify us when their identification cards or financial data is present online. This process has weaknesses, especially when scanned images are of poor quality. The text must be properly identified in order to receive any alerts about client data.

We also started scanning all documents for any type of barcode, including QR codes and license barcodes. This has paid off greatly, and this recent breach turned out to be quite beneficial. Our automated barcode scanning element immediately identified the text details of all customers, which allows us to query by name, address, DOB, DL, etc. Figure 33.03 (left) displays a redacted back of a scanned DL from this recent breach.

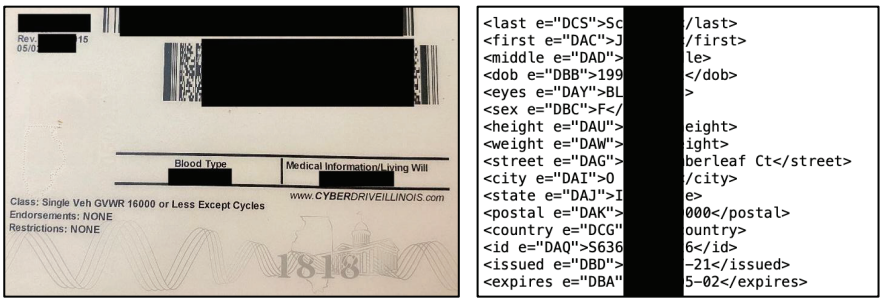


Figure 33.03: A barcode (left) and the text results (right).

Figure 33.03 (right) displays the (redacted) automated text conversion of that barcode, which allows us to ingest text data into our overall breach database for easy access. The first lesson here is that barcode analysis of entire data sets can reveal much more data about the victims within the breach. In this example, our systems ingested the images; performed OCR on all text; scanned the barcodes; populated all text data within our database; and alerted us that a client's driver's license was within the breach, all without any manual effort from us.

The second lesson is to never allow companies to scan your identification. It will never be stored securely and will be leaked during the next attack. The barcode on the back of a passport card only includes the passport number, and may be much safer whenever required to hand over ID. My driver's license possesses a vinyl sticker on the back which contains a new barcode. Scanning it reveals a stern text message about consumer rights. I have yet to allow anyone to scan it though.

The Everest site has terabytes of data published for anyone to see. Add that to the other dozens of groups and you have more data than you likely want to store. We typically eliminate large files such as PDFs and Office documents once our system has ingested the text from the data. Let's conduct another example on the clear web and look at some pure text data which would be much easier to analyze.

In October of 2022, I reviewed a Telegram channel called "baseleak". This channel was offering a 1 GB download of recently stolen medical data in text format. Figure 33.04 (left) displays the original post. However, there was no instant download option. Please note I have redacted these posts in order to protect the identity of the victim organization. This group adds a layer of complexity in order to retrieve and analyze the information. I had to copy the text on the last line, as visible in Figure 33.04 (right). I then had to click "MotokoBot" to load the view seen in Figure 33.05 (left), where I had to paste this text. The password to access the file was present in the comments of the post, as seen in Figure 33.05 (right). This finally unlocked the link, which is visible in Figure 33.06 (left), which connected me to the download site Mega, as seen in Figure 33.06 (right). The entire process took only a couple of minutes.

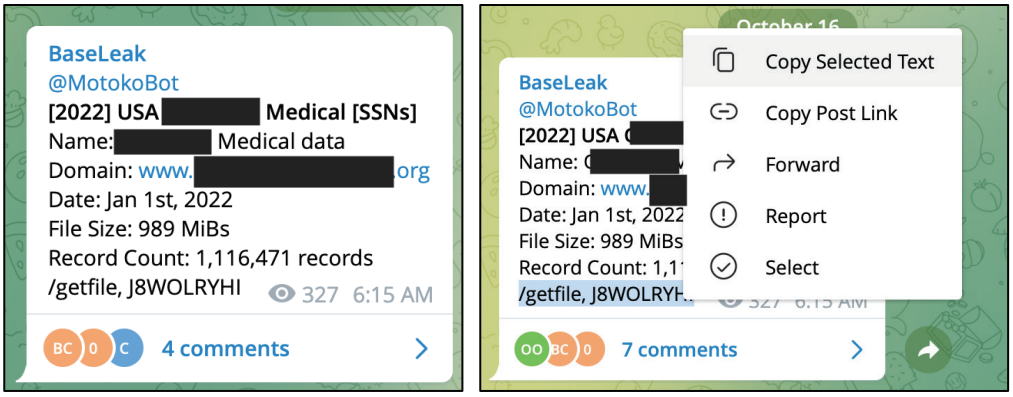


Figure 33.04: Telegram posts containing stolen ransomware data.

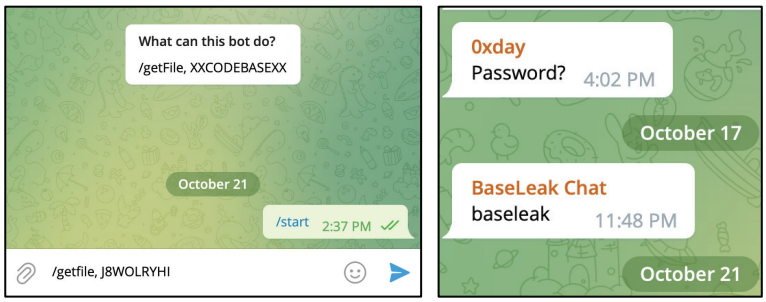


Figure 33.05: A Telegram file request and password notification.

Figure 33.06: Ransomware download links.

The 70 MB compressed file extracted to an almost 1 GB CSV file titled "data.csv". It contained the following 92 fields of data.

```
Master Patient ID, Name, Sex, Date of Birth, DOB Approximate, Age, AgeInDays,
AgeText, PrematureDays, GestationalAge, BSA, Ideal Body Weight, Height,
Weight, HeightMod, WeightMod, SSN, BMI, Brand Preference, Safety Caps, Race,
Language, Religion, Pregnant, Breast Feeding, Smoker, Address 1, Address 2,
City, State, ZIP, Last Room, Last Hospital#, Last Admit Date, Last Discharge
Date, Last Visit #, AllergyList, DiagnosisList, PhoneNumber,
AssigningAuthority, Visit #, Medical Record #, Hospital #, Room #,
CurrentOrPreviousRoom, Room, Prev Room #, Admit Date, Discharge Date,
PlannedDischargeDate, PCU, Facility Code, Service Type, Service Type
Description, CommentsTextID, Physician Code, Physician, Unverified Orders,
CrCl, AdminRecord, Diet Code, Diet Description, Clinical Service, PassStart,
PassStop, Encounter Number, AccountNumber, AllowLateCharges,
AccountNumberAssignAuth, MedicalRecordNumAssignAuth, HospitalNumberAssignAuth,
InternalCommentsTextID, InternalCommentsText, CommentsText, org_int_id,
loc_lvl_3_id, loc_lvl_4_id, loc_lvl_5_id, pcu_int_id, pat_grp_int_id,
psn_int_id, Patient Type Group, SerumCreatinine, PatientCategory, VisitActive,
out_prt_lc, pat_cat_cd, vst_sta_cd, visit_status_code, PatientInBed,
Complaint, FinancialClass
```

That is a lot of invasive and sensitive content. We avoid possessing explicit medical data, such as medical conditions (DiagnosisList, AllergyList) or patient data (PatientCategory, PatientInBed). However, we do collect and store names, DOBs, SSNs, addresses, and phone numbers in order to make proper notifications to our clients. Therefore, I executed the following command to extract only the fields important to me (and my clients), as previously explained.

```
cut -d, -f1,2,3,4,6,17,27,31 data.csv > data-cleaned.txt
```

This minimized my CSV to a 228 MB txt file. I noticed there were many duplicate entries, so I executed the following, also as previously explained.

```
LC_ALL=C gsort -u -b -i -f  data-cleaned.txt > Data-Final.txt
```

I then noticed a lot of extra spaces within the lines, so I executed the following to replace every two spaces with only a single space.

```
sed -i 's/  / /g' Data-Final.txt
```

This resulted in a 31 MB text file with 375,000 unique entries. A redacted example of the text follows.

```
Master Patient ID, Name, Sex, Date of Birth, Age, SSN, Address 1, ZIP
00xx3, PRESTON, JAxx, F, 1982-xx-xx, 459xxxxxx,47xx Rxxxxx RD, ROxx, TX, 77xxx
```

This file contains extremely sensitive stolen information about 375,000 people. If any of our clients are within this data, we will receive an alert and can make notification. I hope you now see why it is so important to protect any breach data which you acquire. While it is publicly available content, you must ensure that you do not leak it further. This is why we never store any breach data within the cloud. It is all stored on a physical server in our office, and access is tightly controlled. We also monitor all activity for abuse.

Putting It All Together

Breach data is nothing new to investigators. I have explained how we ingest typical text-based breach data since the 7th edition of Open Source Intelligence Techniques. The new world of daily ransomware dumps and stealer logs changes everything. We are collecting this data in masses never seen before. We often bring in over 2 terabytes of new stolen content weekly, which we parse down to an average of 25 GB of useful data for the week. Our current collection is over 25 TB. We focus mostly on text and private documents, and try to eliminate all public docs, company materials, and anything else which does not have an immediate impact on an individual.

Figure 33.07 displays a heavily redacted screen capture of my company's internal investigation portal, which was created specifically for our needs. In this example, I searched only the email address of my target. Since this address appears within multiple breaches, it connects me to her real name (from the MGM and LinkedIn breaches). From there, the system cross-references that name to her driver's license. This license was scanned by a mortgage company, which was later hit with a ransomware attack, which resulted in all of their data being published on an onion site. Our system scanned the characters within the scanned image but also the barcode, as explained in the previous posts. Now that our system has enough true data about our target, it can cross-reference everything within our collection of breach data, including ransomware dumps, Tor content, and stealer logs. From there it determines her home address, which is cross-referenced with people search data, historic Whois data, vehicle data, and numerous public APIs. The result is an immediate view of all available public, private, and stolen data. Our team relies heavily on this portal, and each query takes approximately 5-10 seconds. It cross-references any data we have until all options have been exhausted, as encouraged throughout this book.

This brings us to the elephant in the room. How does one new to breach data collection organize all of it? I have no easy answers and database creation easily exceeds the scope of this book. The obvious optimal solution is to create a database and a powerful index, then identify any fields which should be indexed when importing each breach. Maintaining our internal database is a full-time job for a member of our staff. The daily benefits of access to this data cannot be overstated and I believe the availability of this data justifies his salary. I do not pretend to understand the nuances of our entire database. I look forward to a book about importing this type of data, but it will not (and should not) be written by me. Where does that leave you? I have a simpler option.

While I was writing this section, I put my feet in the shoes of a reader new to this type of data collection. I pretended I did not have access to our investigations server and grabbed a couple of terabytes of data similar to that which has been previously discussed here. I then created a script which would manually query the data in the most efficient way possible, absent a structured database. First, I categorized all files in order to limit the data for each query. I then placed everything on a SanDisk 4 TB Extreme Portable SSD (amzn.to/3GO0cP2). The drive was titled "4TBData" and possessed the following folders at the root of the drive. After each folder name, I include a brief description of the content within parentheses.

Business	(Business and Government data leaks)
COMB	(The COMB user:pass database previously explained)
Databases	(35,000 breached databases containing SQL, txt, csv, and other formats of credentials)
Domains	(Various Whois, historical, and other data sets associated with domains)
Hashes	(The Hashes.org and Hashmob data sets)
International	(International data which may not apply to every daily investigation)
Logs-Full	(The full stealer logs acquired using the previously-explained techniques)
Logs-Pass	(Only the passwords.txt files extracted from stealer logs)
People	(Various people search database leaks and breaches)
Ransomware	(Any valuable ransomware content including txt, pdf, csv, and other data)
Voter	(All voter data acquired during the previous tutorials)

Now that my data was split into these folders, I modified my script to match. The entire text from this script is available within the data.txt file previously mentioned within this section. Below is partial text for COMB data.

```
"COMB Email Search")
cd /Volumes/4TBData/COMB
echo "Enter Email Address: "
read data
bash ./query.sh $data;;
"COMB Term Search")
cd /Volumes/4TBData/COMB
echo "Enter Search Term: "
read data
rg -aFiN $data;;
```

If you were to save the entire text from this script into a text file on your macOS or Linux Desktop as "data.sh", the following command would make the file executable. The second command would launch the script.

```
cd ~/Desktop && chmod +x data.sh
./data.sh
```

Upon launching, the script presents the following.

```
 1) Business Records Search              10) Logs Search (Full)
 2) COMB Email Search                    11) Logs Search (Pass)
 3) COMB Term Search                     12) Logs Extract
 4) Databases Search (Files)             13) People Search
 5) Databases Search (Content)           14) Ransomware Search
 6) Databases Search (File & Content)    15) Voter Search
 7) Domains Search                       16) All Search (Files)
 8) Hashes Search                        17) All Search (Content)
 9) International Search                 18) All Search (Files & Content)
```

The first (Business), seventh (Domains), eighth (Hashes), ninth (International), thirteenth (People), fourteenth (Ransomware), and fifteenth (Voter) options simply navigate the user into the appropriate folder; prompt the user for the search term; and launch a Ripgrep search of the content entered. This allows you to focus on a specific area and limit the search data for a faster response. The COMB options present both an immediate query when you know the email address and a Ripgrep search against all data, such as passwords. Options four through six allow you to query by file name, such as "link". When I did this, I was immediately presented dozens of files, including the following.

/Volumes/4TBData/Databases/Turbolink.it.txt
/Volumes/4TBData/Databases/Linkrehber.gen.tr.txt
/Volumes/4TBData/Databases/Linkedin.txt
/Volumes/4TBData/Databases/Link-kun.com.txt

You could then copy the desired filename (Linkedin.txt), and press enter to reveal the menu again. Choosing the sixth option prompts you for the filename and then any search data. You could supply an email address and immediately receive only the desired result. This would all occur within seconds since you are focusing on specific data.

The Logs search options allow you to either search through all logs or only the passwords (if you applied the previous technique for this). If you have not, you could place all of your logs within the Downloads folder of your macOS or Linux machine and execute option twelve. It will apply the technique and create a new file called Passwords[Current Date].txt at the root of the Downloads folder which only contains the credentials from your data. The date is added so that the file is not overwritten the next time you execute it. Possessing these smaller searchable files greatly reduces the need to query huge data sets.

Finally, the last three options allow you to either search for file names, file content, or content within a specific file throughout your entire collection of data, similar to the "Databases" option. Having all of these options reduces your query time, and gives you greater control over the data included.

I am simplifying the explanation of this script, but you have everything you need to conduct your own research into its capabilities. Study the paths within the `data.txt` file in your online portal and modify the content as needed based on the tutorials within Chapters Four, Five, and Six. Possessing an efficient way to query your data will increase the likelihood that you will take advantage of these techniques. During my testing, I could use this script to query the entire COMB data within a few seconds; display file names from thousands of breaches immediately; or focus my queries within generic data types or specific files in less than a minute, all from an external drive. Consider the following typical usage.

- **Internal Investigations**: This one is obvious. This type of data can be crucial to investigations. It can immediately uncover real and alias information. Every day, breach data reveals the true person behind a burner account due to sloppy OPSEC.
- **Client Vetting**: We take our clients' privacy very seriously. We would never use a third-party system to properly verify the identity of a potential client. Instead, we use our own in-house system to make sure we know who we are dealing with. At least twice, a potential client's record in our system revealed active warrants for arrest, identifying the reason they were asking about anonymous relocation services (we never assist in those situations).
- **Client Exposure**: We have many clients who keep us on a retainer. Part of that service is an initial analysis of exposed information, and a constant monitoring for any new details. Every week, we reach out to clients to let them know of a new breach, ransomware attack, or password log which may impact them or their company. Just a few months ago, I was able to alert a close friend that his daughter's full name, address, DOB, SSN, and banking details were being passed around a criminal marketplace focused on identity theft. We received the internal alert within two days of the initial exposure.

As I finished writing this section, I wanted to test my script with one last demonstration. I downloaded 50 GB of stealer logs from Telegram into my Downloads folder on my OSINT MacBook Pro. I then launched my script and selected the "Logs Extract" option. Within a few minutes, I possessed a 1.5 GB file titled "_Passwords.txt", which I moved into the "Logs-Pass" folder on my external drive. Back at my script, I pressed return to bring the menu back up and selected the "Logs Search-Pass" option. I was prompted to enter my search query, which was "inteltechniques". Within a few seconds, I was presented the login credentials of two of the members of the online video training. Apparently, they unknowingly possessed infected computers. I was able to reach out to them and offer guidance on removing the infections (and changing all of their passwords). This displays the true power of this data. Imagine the possibilities once you possess several terabytes of leaks, breaches, logs, and ransomware. **Please remember that these types of files are likely to contain malicious and undesired software!** Always practice good digital hygiene and isolate this data from anything valuable or important to you or your investigations.

Data Cleansing

It will not take long for your own data collection to fill your drive space. Every day, we encounter files over 5 GB in size which only possess a few megabytes of interesting information. This is where data cleansing can be vital. I have already shared a few ways to minimize unwanted data within the previous examples. The following is a cheat-sheet of the Linux commands we rely on every day. Each assumes you are modifying a file named data.txt and you have already navigated to the desired folder within Terminal. Use these overall strategies by replacing my data with your own.

Replace "OLD" with "NEW": `sed -i 's/OLD/NEW/g' data.txt`
Replace all commas with a hyphen: `sed -i 's/\,/\-/g' data.txt`
Replace all tabs with a comma: `sed -i 's/[strike ctrl-v-tab]/\,/g' data.txt`
Remove all data until the first comma: `sed -i 's/^\([^,]*,\)//g' data.txt`
Remove all data until the first colon: `sed -i 's/^[^:]*://g' data.txt`
Remove all single quotes: `sed -i ' s/\'//g' data.txt`
Remove all double quotes: `sed -i 's/\"//g' data.txt`
Remove "junk": `sed -i 's/junk//g' data.txt`
Remove all between "FIRST" & "THIRD": `sed -i 's/\(FIRST\).*\(THIRD\)/\1\2/' data.txt`
Remove all digits between commas: `sed -i 's/\,[0-9]*\,//g' data.txt`
Remove any line beginning with "TEST": `sed -i '/^TEST/d' data.txt`
Remove any line not containing "@": `awk '/@/' data.txt > newfile.txt`
Remove empty lines: `sed -i "' '/^$/d" data.txt`
Remove first 10 lines: `sed -i '1,10d' data.txt`
Remove first ten characters: `sed -i 's/^.\{10\}//' data.txt`
Remove everything after the last "_": `sed -i "s/_[^_]*$//" data.txt`
Remove duplicate rows: `awk '!seen[$0]++' data.txt > newfile.txt`
Remove duplicate rows and sort: `LC_ALL=C sort -u -b -i -f  data.txt > newfile.txt`
Remove data between "{" and "}": `sed -i 's/{[^}]*}//g' data.txt`
Extract email addresses: `grep -E -o "\b[a-zA-Z0-9.-]+@[a-zA-Z0-9.-]+\.[a-zA-Z0-9.-]+\b" < data.txt > newfile.txt`
Split large file into multiple files: `split -l 200000000 data.txt 1`
Display total lines in a file: `wc -l data.txt`
Cut columns 1, 2, and 6: `LANG=C cut -d, -f1,2,6 data.txt > newfile.txt`
Remove hyphens from phone numbers: `sed -i 's/\([0-9]\{3,\}\)-/\1/g' data.txt`
Cut columns from JSON: `jq --raw-output '"\(.email)\(.password),"' 1.json > new.txt`

I highly recommend the free version of **BBEdit** (barebones.com/products/bbedit) for all macOS users. It opens incredibly large files and allows you to easily view and modify the content. I know of no other text viewer which will open a 5 GB file in seconds and allow immediate find and replace functionality. It may eliminate the need for all of these commands.

Duplicate Files

If you collect enough breach data, you will likely find many duplicate files with different names. This could be due to poor file naming or deliberate trickery in order to distribute inaccurate data. Regardless, I find it beneficial to eliminate duplicate files often. The following commands install fdupes.

- Linux: `sudo apt-get install fdupes`
- macOS: `brew install fdupes`

The next command launches fdupes, scans recursive files within the Downloads directory, and prompts you to select which duplicate file to keep.

```
fdupes -r -d ~/Downloads
```

I placed two identical files with different names in the same directory, the following was the result. Entering "1" would keep the first file, while "2" would preserve the second.

```
Set 1 of 1:
  1 [ ] /Users/mbp/Downloads/1/Real Breach.txt
  2 [ ] /Users/mbp/Downloads/1/Fake Breach.txt
( Preserve files [1 - 2, all, help] ):
```

I want to close this section with some final warnings and reminders. Forgive me if this seems redundant, but we can all use another mention of the risks and concerns. Leaks, breaches, logs, and ransomware always consist of data which was either stolen or never meant to become publicly available. It is abused every day by professional and amateur criminals. Many readers will be disgusted by these chapters and upset that I share details on the acquisition of this content. When collected and used properly, I believe it can be beneficial to both investigators and those who monitor this activity from a defense perspective.

If your employer tasks you with the investigation of content stolen from your organization, I want you to know what to do. I want you to locate, obtain, and analyze the exact same data which criminals are abusing. We can no longer keep our heads in the sand and hope no one notices the leakage of sensitive data. We must think like the criminals and attack from their mindset. If you are an online investigator who can responsibly benefit from this type of data, I want you to know what to do with the content you find. I also want you to protect the data and make sure you cause no further harm. Every day, this type of data assists me with positive identification of the true identity of stalkers, harassers, scammers, and other unsavory people. We can either allow this data to be solely used by criminals to make our lives more difficult, or we can use it as a tool to expose those criminals which would otherwise remain hidden.

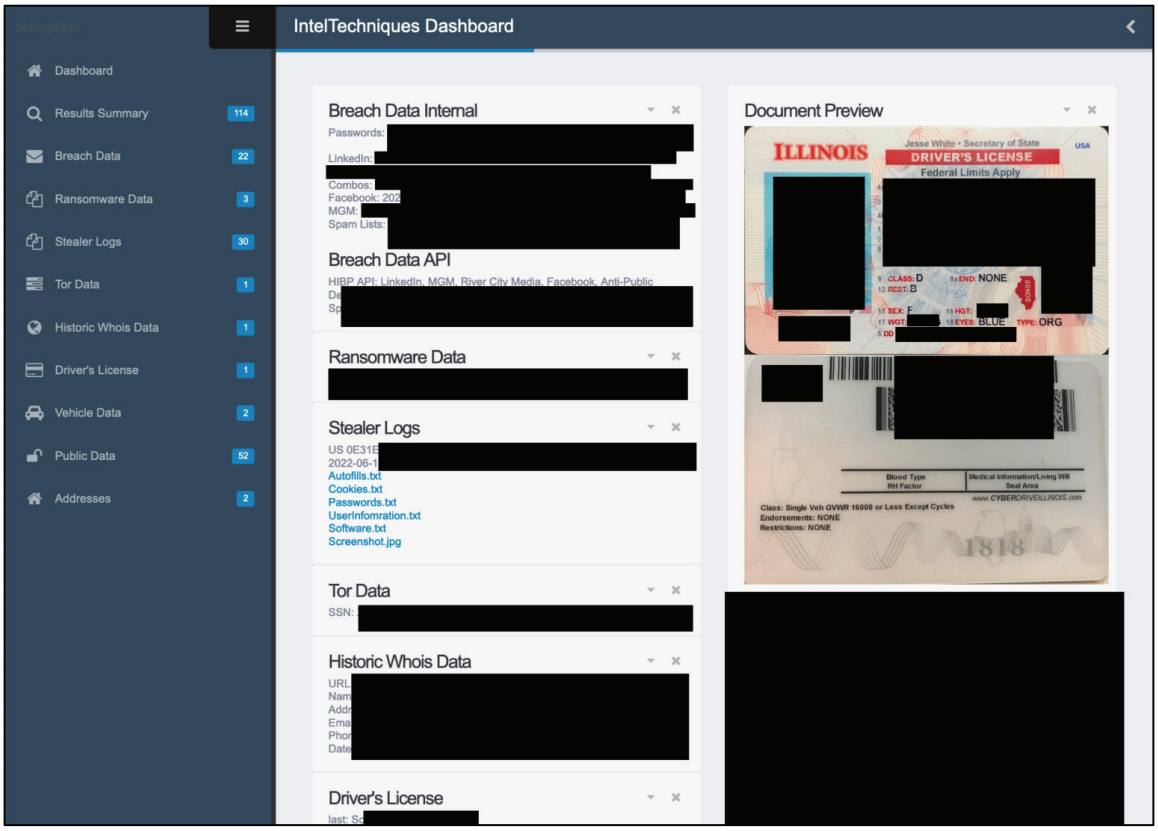


Figure 33.07: A private breach database result portal.

SECTION IV
OSINT METHODOLOGY

In the late 90's, I was tasked to investigate a computer-related crime involving inappropriate online contact from a registered sex offender to children in his neighborhood. The internet was new to most people; AOL dial-up connections were common; and there was very little monitoring or enforcement in place. I contacted the subject at his home and conducted an interview. He admitted to inappropriate behavior and showed me the evidence on his computer. I had no forensic imaging machine or acquisition methods. I didn't even have a digital camera. I had my notepad and pen. Months later, I testified about the illegal activity this suspect conducted with local children. I verbally explained what I observed on his computer without any digital evidence. It was a very different time, and would never be acceptable today. Current prosecution would require forensic acquisition, detailed logs, and pictorial proof of every step. This is a good thing, but presents a higher demand toward your own documentation and overall OSINT methodology. Without digital evidence, the computer crime or online incident you are investigating never happened. Without proper training and policies, your evidence may never be considered. Without confidence in your work, you may not be taken seriously.

This section enters territory I have always avoided in previous editions. We can no longer ignore discussions about workflow, documentation, and other formalities of our investigations. It is also time that we tackle the ethics surrounding online investigations. These are not easy conversations, and many people will have their own opinions. I do not claim to have all of the answers. I only have my own experiences and lessons learned from many mistakes.

I rely heavily on assistance from my friend and colleague Jason Edison throughout this entire section. Jason is a 20-year veteran of a major U.S. police department where he serves as the investigative lead for the agency's Digital Crimes Unit. He has trained thousands of students in the public and private sectors on various topics related to open source intelligence and cyber-crime investigations. In fact, he is an official IntelTechniques OSINT instructor who travels the world presenting my methods. He also maintains the IntelTechniques online video training courses at IntelTechniques.net. Most of the content in this section is directly from him. I maintain the first-person usage of "I" throughout the section. It is a collective "I" from both of us.

Jason and I do not agree on everything presented here. This is why you see alternative tools and methods which may contradict each other. This is a good thing. We need as many thoughts and opinions as possible in order to present ideas applicable to many situations. As an example, I try not to use Microsoft or Google products unless absolutely necessary. I have forced myself to use Linux whenever possible, and avoid closed-sourced tools which "call home". Jason prefers Microsoft OneNote, which is extremely robust. His need for an ideal note-taking solution outweighs my paranoia of metadata collection by Microsoft. He prefers Chrome while I insist on Firefox. Neither of us are right or wrong. We simply have strong preferences. We only hope to present numerous options which may help you choose the best methods for your own investigations. Only you can decide what is most appropriate for your daily workload.

It is now time to take a breath and get back to basics. You may have been overwhelmed with the techniques discussed throughout the previous sections. You may wonder how you will present your findings, create a report, and defend your right to access public information from the internet. This section tackles these issues. Throughout this section, we present numerous document templates and workflow diagrams. All of these are available to you via digital download at the following URL.

https://inteltechniques.com/osintbook10

Enter a username of "**osint**" and password of "**book1443**" (without quotes) if required.

CHAPTER THIRTY-FOUR
METHODOLOGY & WORKFLOW

An often overlooked component of open source intelligence gathering is the importance of establishing an efficient and repeatable workflow. You need to be thoughtful and deliberate in how you proceed on an investigation rather than wading in haphazardly. As an instructor, one of the most common stumbling blocks with which I see new practitioners struggle is putting the tools into action in a fashion that results in a professional looking work product. This section provides a step by step walkthrough of the entire investigative process, from receiving an OSINT assignment all the way to submitting a professional case report.

This chapter assumes you have already completed the steps laid out in the previous sections of this book. You will need familiarity with each of those tools and techniques if you wish to take full advantage of the recommended workflow. If you have an existing investigative process, there will likely be pieces shared here that can be folded into your current procedures. The examples used here were chosen purely for demonstration purposes and not due to any association with ongoing criminal investigations.

Receiving the OSINT Mission

The first step in most investigations is what we in law enforcement refer to as "intake". This is the process of receiving a mission assignment from a supervisor or fielding a request for investigative support from another internal unit or outside agency. For those in the private sector, this might be accepting a contract investigation from a client or conducting a security assessment as part of your normal duties. The following are examples of OSINT requests that we receive on a regular basis:

- **Threat Assessments (Individuals):** Online threats to carry out an act that we wish to prevent. Who is this person? Where are they? What is their capability and true intent?
- **Threat Assessments (Events):** Monitor intelligence prior to and during a significant event that impacts the organization or region of responsibility. Who is involved? What are their intentions? What is the scale of impact on available resources?
- **Target Profiles (Individuals):** Uncover the target's entire online presence, including email addresses, home addresses, friends, hobbies, etc., often referred to as "due diligence" investigations.
- **Target Profiles (Organizations):** Uncover an organization's online footprint and/or entire technological infrastructure. This can be a business, criminal enterprise, or group of individuals organized to pursue a shared goal.
- **Subscriber Identification/Account Attribution:** Identify the real person associated with a domain, IP address, or online account. Who runs a malicious website? Which child predator has web traffic through this IP address?

The following recommendations can be applied to any of these common investigative scenarios. More than anything else, the key to success is staying organized and having a repeatable process.

Triage

Triage is the practice of assessing a situation or mission to calculate an approach that is likely to result in the best possible outcome. A common mistake that is made when working OSINT investigations is to rush to action with no clear plan or direction. You should take time at the beginning of a mission to ensure you are following a productive path to relevant answers. Depending on the urgency of the situation, this step could be 30 seconds or 30 minutes. The important thing is to make a plan of attack and move forward with purpose rather than just bush-whacking your way through the internet. Here are some of the key considerations during the triage phase of your investigation.

Be certain of the mission objectives. If you ask a professional analyst to describe the first step they take in any assessment, they will tell you that it is to identify the question. This of course could be multiple questions, but the important thing is that you articulate the investigative goals. This can be a verbal or written confirmation, depending on your situation, but written is preferred should the other party later misremember the conversation.

The first benefit of articulating the questions is establishing a clear set of expectations with the person asking you to do the work. This could be a supervisor, contract client, colleague, or victim of a crime. Do not overthink it. An example could be: "To be clear, you want to know the real name and physical addresses associated with the person in control of the email account of ramit@iwillteachyoutoberich.com, and we have two hours to accomplish this. Is this correct?"

Include in your verification any specific identifiers (email addresses, names, phone numbers, IP addresses, etc.) that were originally provided by the requestor. It gives them a chance to catch any typos or miscommunications. They may have given you the email address of the victim rather than the suspect. Those types of mix-ups occur frequently and can waste a lot of valuable investigative time and resources if not caught and corrected early on. That quick clarification also defines the primary goals for our investigations, similar to the following.

- Find the legal name of the real person associated with ramit@iwillteachyoutoberich.com.
- Find any home and/or work addresses for ramit@iwillteachyoutoberich.com.

When it comes time to write your investigative report, these questions should be clearly addressed in the summary of key findings. Taking the time to articulate and clarify mission goals up front lays the groundwork for your final work product. You should also ask questions regarding the source of any initial leads or other intelligence on your target. Why do we believe that email address belongs to our suspect? How was that lead obtained and how certain are we that it is correct? **Information is not intelligence until it has context.** You need to ask questions up front to establish any available context for the target. Do we know anything about their location or culture? Are they into video games or an avid pro-baseball fan? Once you get to the research phase of the investigation, you will have a far easier time locating pages, accounts, and identifiers related to your target if you start learning about his or her day to day life. Never assume that the person tasking you with this work is giving you all available information. Ask questions and be persistent.

Legal Service & Preservation Letters

If you work in support of law enforcement, you should consider if there is likely to be a legal request made to any known social media platforms. For example, if a Gmail address was involved in a crime, you might want to issue a preservation letter to Google requesting that they retain any data related to the specified address. The preservation letter is issued in anticipation of a future subpoena or search warrant for account data, such as subscriber information. If you are unsure of whom to contact at a particular provider in order to submit this request, a good starting point is the ISP list at https://www.search.org/resources/isp-list/. Try to get a live person on the phone rather than just sending an email. Build rapport with the support person or legal contact and shepherd them into doing the right thing based on the urgency of the situation. No one wants to be responsible for a teen suicide or the next school shooting. Often, they will be much more cooperative and productive if they feel invested in the situation.

Deconfliction

Not all investigations involve infiltration into criminal organizations. However, when they do, you may want to check with colleagues in other agencies to make sure you are not stepping on any ongoing investigations. This could also save you time should you locate an investigator who has already laid groundwork into the online communities in question. We always want to be respectful of another professional's work and collaborate whenever operationally appropriate. In the past, I have concluded long-term investigations only to find out later that other teams were running operations in that community at the same time. While reviewing the case, we found that we had wasted time working to gain reputation with users who, unbeknownst to us, were other

undercover investigators. This is not only a waste of time and focus, but can complicate the individual cases where they overlap. Ask the person for whom you are doing the work if anyone else is working on the case. You would be surprised how often two branches of the same agency are unknowingly pursuing a common target.

Note-Taking

The triage stage is the appropriate time to begin your note-taking. I will discuss specific tools in the next chapter, but at its core you will need a paper scratch pad and digital notebook, such as Microsoft OneNote. A paper notepad allows for quickly noting key details without having to move out of your browser during searches. This is even more crucial if you are on a laptop or otherwise have limited screen real estate with which to work. Your digital note-taking application is for pasting content as you copy it from your browser. Keep in mind using Microsoft products allows them to collect user information, so make sure that this is within the operational security requirements of your organization. At the top of your legal pad, list out the details you are trying to find and any initial investigative steps. This does not need to include extreme details, but it establishes your plan.

Key Questions/Goals

- Find the real name associated with ramit@iwillteachyoutoberich.com.
- Find any home and/or work addresses for ramit@iwillteachyoutoberich.com.

Investigative Steps

- Using Chrome, Google search ramit@iwillteachyoutoberich.com.
- Query ramit@iwillteachyoutoberich.com using custom email tools.

In your digital notebook, create a new section and title it logically based on the date of request, incident type, or case number if there is one. For example, "OSINT Request Jan_10_2023" or "Robbery 23-486544". Any emails or other written correspondence received leading into the case should be copied into your digital notebook. Finally, before moving on, ask yourself if OSINT is the right tool for the job. I have made the mistake of investing hours into online searches, only to realize later that a two-minute phone call would have likely given me the same information. Do not make the mistake of overlooking traditional investigative resources. Would a phone call to a postal inspector identify the occupants of a residence quicker than an online search? The strongest investigators are ones who think outside the box while also using every tool in it.

Knoll Your Tools

Now that you have established a plan and a clear understanding of the mission goals, you need to prepare your workspace for the investigation. "Knolling" is the process of organizing your resources so they are ready to go and easily accessible once you start the actual work. Think of how a surgeon's instruments are sanitized and laid out in an organized fashion. The time spent preparing up front will result in a more efficient overall operation while also reducing the chances of unnecessary mishaps. If you followed the instructions and recommendations in previous chapters, you should already have a custom Linux VM. It should be patched and preloaded with your preferred OSINT applications. Additional recommended preparations prior to the search phase include the following.

- Check the status of your VPN on your host machine. If not already connected, join a node in a geographical area close to where you believe the target is located.
- Start VirtualBox and load your custom OSINT virtual machine.
- If you are using a Windows-based digital notebook, such as OneNote, you will need to switch back to your host environment (Windows) when adding content to that notebook. In the next chapter we will look at a Linux compatible notebook that has some of OneNote's desired functionality.
- Once in your OSINT VM, open your browser from the Dock.

- If you have not already done so, log in to your covert social network accounts used for OSINT research. We will likely need to conduct searches on these platforms and pre-authenticating in this browser session will save us time later.
- If you need to make new covert social media accounts for research, you should disconnect from your VPN prior to doing so. It should also be noted that running several queries on freshly made accounts is highly likely to result in security challenges and/or suspended accounts. Try to always have a supply of semi-mature accounts available for time-sensitive investigations.
- Open the custom OSINT tools which you built in previous chapters of this book.
- Create a new folder in the shared directory of your VM and rename it to match your digital notebook including date, target name, or case number. This is where we will store any manually captured digital evidence such as saved images or pdf captures. I keep a directory in my Original VM that is prepopulated with folders titled to reflect my custom tools. This gives me an organized destination for anything I recover and saves me from having to repeat this step every time I open a new case. Figure 34.01 displays an example of this, and the digital templates contain the same structure, which can be copied if desired.
- If you use a digital notebook that accepts the pasting of multimedia filetypes, you have the option of storing files within your digital notes.

Your knolling is complete. You have a virtual machine preloaded with the most useful OSINT tools, and you are on a secure and private connection. We are prepared to search quickly, collect pertinent content, store it logically, and track our progress within our notes.

Closed-Source & Premium Data

You should now be ready with all your tools and note-taking resources. Begin the research phase of the investigation by querying your target against any in-house, premium, or proprietary data sources. This includes any of the following.

- Commercial aggregators such as Accurint (LexisNexis), TLO, Clear, or others.
- Premium products such as BeenVerified, Intelius, Spokeo, Pipl, and WhitepagesPro.
- Government and LE databases such as Department of Licensing, Criminal Records, Department of Corrections, and Agency Records Management Systems.

Whereas using purely open-source tools typically requires visiting dozens of sites in order to find just a few leads, paid services often quickly provide a list of possible addresses, associates, and accounts. If you have services like LexisNexis or Clear available, use them early for easy additional leads on your target. These services obtain much of their data from credit histories and utilities. Therefore, they tend to be good sources for residential address history, land-line phone numbers, employers, roommates, and family members. They tend to work very poorly with email addresses, usernames, and social media accounts.

This is also when you should run any premium people-search services such as Pipl, Spokeo, or BeenVerified. These types of services range from $15-$300 a month depending on the subscription tier, but tend to offer a much richer, immediate return on queries than their free counterparts. Although Pipl Pro formerly offered some of the best premium results, they are also one of the most expensive. Additionally, they have moved to a complex per-record pricing model. Spokeo is one of the cheapest at $15-$20 a month depending on your plan, but they have a modest amount of data for a paid service and charge extra to make reports easily printable. BeenVerified allows you to run an unlimited number of fairly extensive reports for $53 quarterly and they are print friendly. However, they will push you to spend more money for "premium" data about your target. Many investigators kickstart the early stages of their open source investigations using one of these cheap premium aggregators, but keep in mind everything you get from paid people search sites is available for free elsewhere (although with considerably more time and effort).

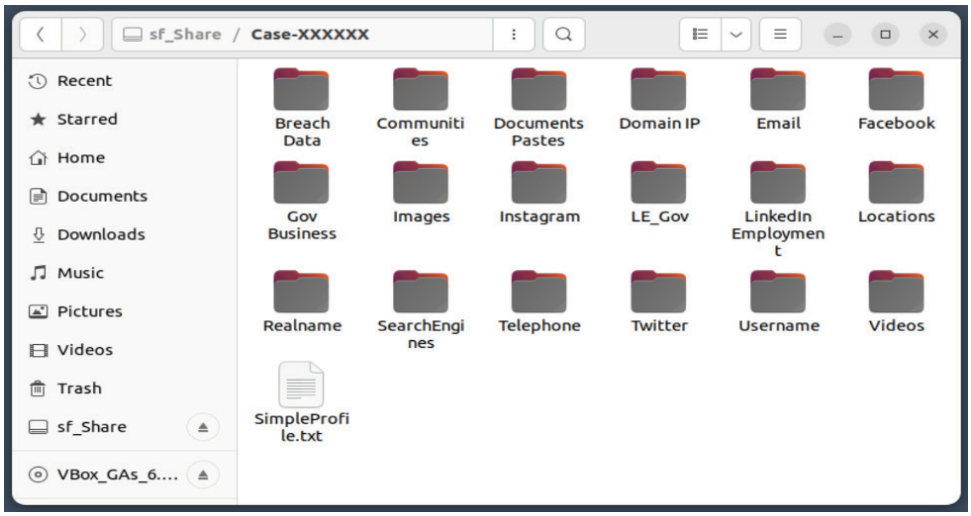


Figure 34.01: A logically structured case directory.

For those on the government or law enforcement side of the house, internal agency records systems, department of licensing requests, and criminal history queries can be very powerful additions to your early digging. An advantage that government investigators have is that many of these systems will provide access to a photo of the target, which can be used to verify or rule out possible social media accounts. These records also typically include recent associates, phone numbers, and residential addresses. Even if the subject did not use their own address during a previous contact with government agents, they likely used one where they can receive mail, such as a relative's house. Most people are not trained to think on their feet and will use familiar data when cornered with hard questions.

Any promising identifiers from your premium or government searches should be added to your notepad, and generated reports should be dropped into your digital notebook as pdfs. Photos can be copied and pasted into your digital notes or dropped into your designated directory within the shared folder on your desktop. This reflects our workflow going forward. Any valuable page, image, or identifier is noted, and a corresponding pdf or image capture is placed either in our digital notebook or investigative folder.

Open-Source Research & Collection

Once you have exhausted your in-house and paid data resources, it is time to dive into your OSINT tools and resources. This tends to be the most involved stage of research due to the large number of sites that you will check for your target's information. Tab management is critical in staying organized. If you have not already done so, add the **OneTab** (one-tab.com) extension to Chrome and Firefox within your VM.

- In your VM, conduct a quick Google search on your known identifiers.
- Open your custom OSINT tools and use the tool category that corresponds to your known identifiers, such as the email and search engine tools.
- Perform any additional queries on sites not included in your custom toolset. For example, a colleague may have very recently recommended a new email search site. If that resource provides good results, consider adding it to your custom toolset.

At this point, you should be in your VM looking at several open tabs in your browser. These tabs represent the results from the Google and custom tools queries which you have executed. The rule going forward is to deal with each tab completely, and intentionally keep or discard it before moving on to the next. A common misstep is to start clicking on leads that look interesting prior to completely reviewing the page on which you are currently visiting. Therefore, tab discipline should be in the forefront of your mind as you parse through your first batch of search results. Consider the following.

- Review the first tab of Google results, looking for anything that stands out as a likely valid lead on your target. For any results that look promising, right-click the link and choose "Open link in new tab".
- Continue to scroll through the first page of Google results and when you get to the image results, right-click on it and choose "Open link in new tab". If Google does not include an "Images for…" section in the first page of results, you may need to select "Images" from the tabs at the top of the page. The image results are always worth reviewing as you can quickly scan the page for potential profile images or other photos of your target.
- Once you are satisfied that you have fully reviewed the first page of Google results and have opened any promising leads in their own tabs, you can move on to the next tab.
- As you start to do more OSINT work, small efficiencies compound to save a lot of time in the overall investigation. Learning keyboard commands for frequently used browser actions will be very beneficial. In this case, you can press "Ctrl" + "tab" (Windows) or "command" + "tab" (Mac) to move to the next tab to the right. Holding down "Shift" with the previous key combinations will cycle through tabs in the opposite direction, from right to left.

This system of exhausting leads on the current page before moving on to other tabs is crucial in ensuring that you do not overlook potential intelligence or lose your way by moving too quickly from lead to lead. That is called "rabbit holing" and is the biggest pitfall with which new investigators inevitably struggle. You also need to be disciplined about closing any tabs that are false positives or otherwise present no fruitful results. This will help to control browser clutter and reduce the load on your workstation resources.

As you move through your tabbed results methodically, you may come upon a page of results which is a jackpot of links to potential target data. This is a good problem to have, but a problem nonetheless. The same rules apply, but with one additional recommendation, which is that any lead that warrants its own full set of queries should be opened in a new window rather than a new tab. Consider the following example.

- You have located a social media account containing several strong leads which require their own full set of queries using Google and your custom OSINT tools.
- The Twitter usernames need to be queried through the Twitter tools and the email addresses through the email tools. Think of each of these as a new path that needs to be followed independently.
- Any strong leads should be represented in your notes. Write down any account identifiers on your legal pad, and for each create a new page in your digital notebook. Figure 34.02 displays the documentation in OneNote.
- Much like a family tree forks into new branches which create even more branches, following online leads often presents new leads. Browser tabs and windows can help you categorize and isolate all leads, providing a sense of structure to your investigation.

This OneNote digital notebook is logically structured to organize intelligence leads as they are uncovered. The notebook title on the top left reflects the case number and name of the target organization. I should mention that this example was chosen arbitrarily, and the group depicted is not likely criminal in nature. I have tabbed sections for the target individual and the organization. I also have a tab which contains fresh copies of my OSINT templates should I need them. The visible section represents a "real-name" investigation into my primary target. On the right, I have added pages that reflect each strong lead, which was created for each account identifier. This ensures that every new investigative path I open has a place to paste relevant data, while also making it easier to visualize my leads as a whole. The following explains some of the options present in Figure 34.02.

- The Scratch Page is for quickly pasting links and reminders for items which I want to have quick access or revisit later.
- The Request Details page is where I paste the details of the investigative request along with any other important details gleaned during triage.
- The various Premium/Government Data resource pages contain pasted reports or snippets from closed-source, in-house, and paid services.

Strong leads are given a new browser window and their own page in my digital notebook. The list of pages can also be used as a checklist or "to-do" list. Once you have fully exhausted that lead, you can add a "+" symbol or any other desired character to the title to indicate that it is complete.

Tab Management

When you reach the last open tab in your current search, look back and make certain that any open tabs are pages that have useful data. Prior to moving on to a new window and path of inquiry, you should preserve your list of tabs. This is where your tab manager can be beneficial. OneTab's primary purpose is its ability to quickly collapse all tabs into an exportable list of URLs. These bookmarks can then be pasted into your notes or shared with a colleague who can import them into their own OneNote instance. Once you are finished working with any set of tabs, conduct the following.

- Right-click anywhere in the window, select the blue OneTab icon, and click "Send all tabs to OneTab".
- You will now be looking at the OneTab manager, which was explained in Chapter Three. The list of bookmarks for the tabs you just collapsed will be at the top of the list.
- Click in front of the number of tabs to add a title, such as "Google Target Name". Logical naming is the cornerstone of staying organized and making your work searchable. Eventually this set of bookmarks will get pushed farther down the page as you add newer tab sets. To find it again, press "Ctrl" + "F" (Windows) or "command" + "F" (Mac) to conduct a keyword search for the custom title you added.

The list of tabs is saved locally in your VM within your browser extension data, but you will want it in your notes for easy access. Click on "Export/Import URLs" on the top right of the page. Figure 34.03 displays an example. The export page is missing titles, but each set is separated by a space and the URLs are in the same order as the lists on the OneTab management page. Consider the following steps.

- Left-click and drag your mouse to highlight your set of URLs. Press "Ctrl" + "C" (Windows) or "command" + "C" (Mac) to copy the list.
- Move to your digital notebook, select the appropriate page, and press "Ctrl" + "V" (Windows) or "command" + "V" (Mac) to paste the list into your notes. Figure 34.04 displays this content within OneNote.

Although OneTab is the tab manager I recommend for most people, if you require online sync or advanced features, some other tab extensions are Toby, Tabs-Outliner, Workona, and Graphitabs. As discussed earlier in this book, extensions always come with a security cost, so use them sparingly and only when the added functionality is mission critical.

Context Menu Queries

Speed and efficiency are key if you are conducting OSINT at the professional level. Context search extensions, such as **ContextSearch** (github.com/ssborbis/ContextSearch-web-ext), allow you to query a keyword or link just by right-clicking on it. These extensions come with a predefined set of search options, such as Google reverse image, and allow you to also add your own custom queries.

Queries are customized via the options menu by adding structured URLs to the existing list of popular search engines. To add your own custom search strings, conduct the following.

- Left-click on the "ContextSearch" extension in your toolbar and then click the gear icon to go to settings.
- Select the "Search Engines" tab and click the "Add" button.
- Type in a name for your new search and click "OK".

- In the template field, paste the URL query for the search you want to execute. These can be some of the same queries that you have added to your custom OSINT toolset. At the end of the URL add {SEARCHTERMS}.

If I highlight a phrase on a page, such as "tom jones" and right-click that phrase, I now have a Dehashed option in my ContextSearch menu. Clicking on that Dehashed search will provide immediate search results. Context-based search capabilities complement, but do not replace your custom toolset. They offer speed and convenience, but lack the full level of control and customizability that you have with your own tools.

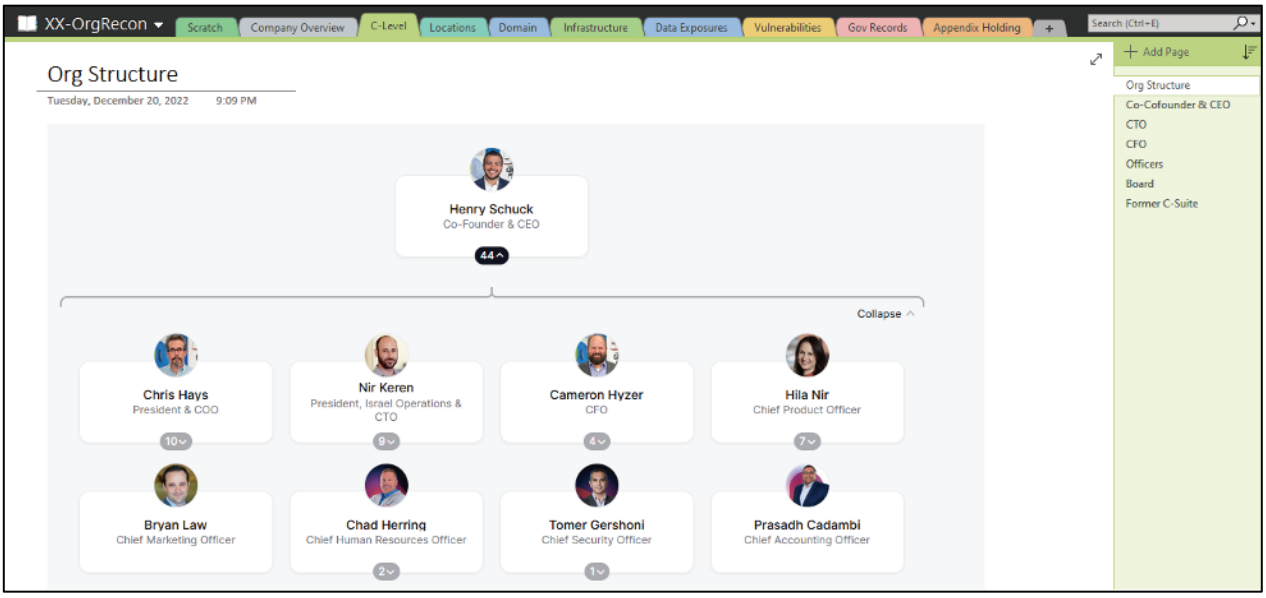


Figure 34.02: Structuring digital notes in OneNote.

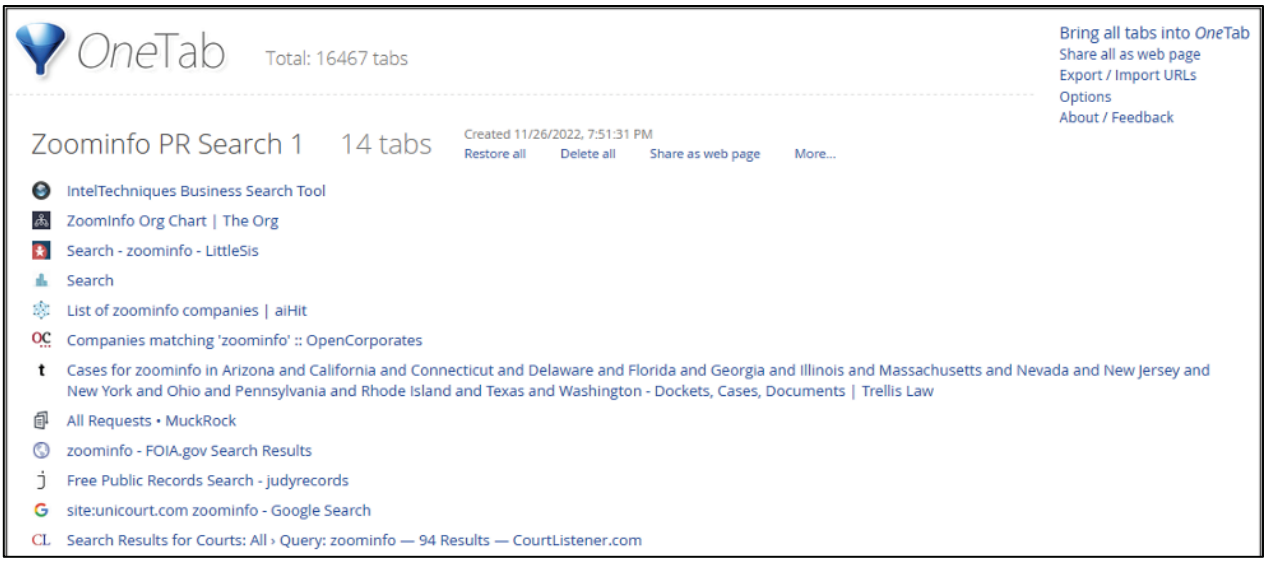


Figure 34.03: The OneTab management page.

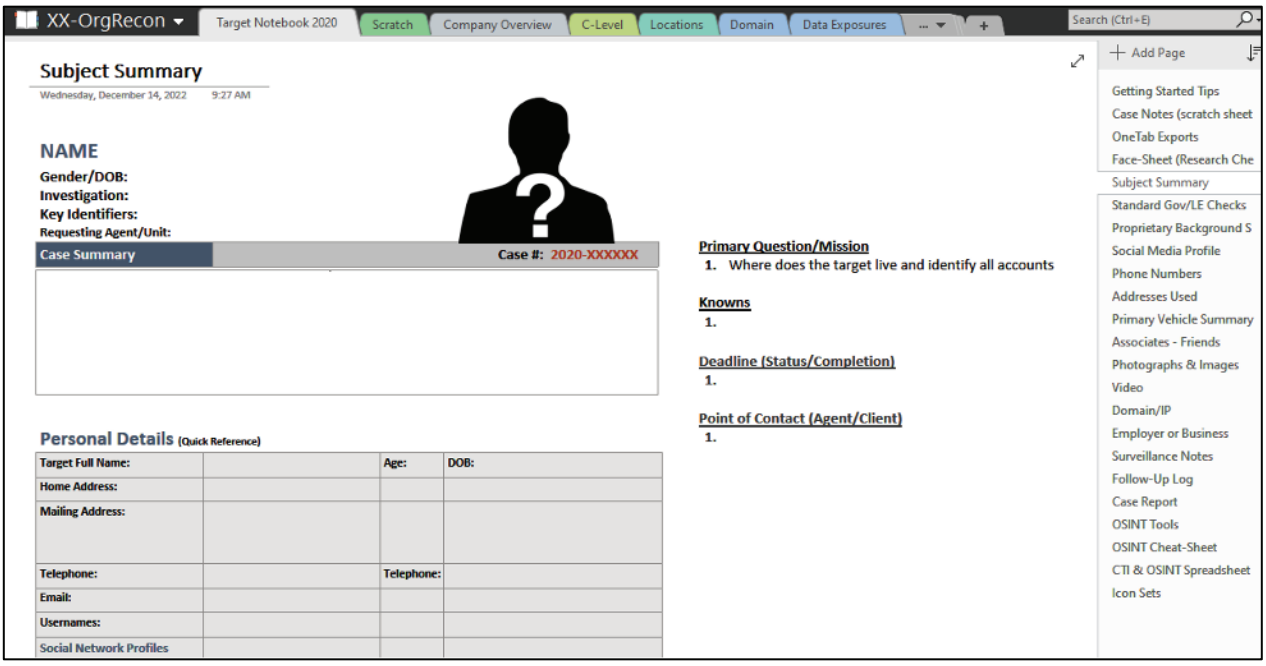


Figure 34.04: An OSINT case template in OneNote.

Capture and Collection

You should now have a reliable, repeatable process for working through your OSINT queries and capturing results in your paper and digital notes. The final piece of the research phase is capture and collection. This is made up of the steps involved in preserving content for evidentiary or reporting purposes. There are three approaches to collection depending on the tools you have available.

Manual Capture

Manual capture includes any technique that is user-initiated and results in capturing a single file or multiple files within a single page. These are often browser-based tools such as JavaScript or extensions. Here are the steps to integrating manual capture into your workflow.

- Create a case folder in your VM shared directory named logically for your case.
- Open that case folder and create a new folder matching the corresponding category from your toolset. If I am working on an email address, this will be titled "Email Addresses".
- Open that directory and create a folder titled appropriately for the identifier you are querying. If an email address is the target, the folder may be titled similar to "ramit@iwillteachyoutoberich.com".
- Repeat this for any other strong leads, such as Twitter usernames, names, domains, etc.
- Now you have a logically structured set of folders to store any saved digital content.
- As you work through your set of tabs specifically for that lead, capture any pages that support your findings and save those to this folder.
- Any time you save an image or video related to that lead, also save it to this directory using any of the tools referenced earlier in the book.
- Any time you save a specific image or video, you should also save a screen capture of the page from where you obtained it. This shows visually how that image or video was laid out on the page. The capture of the entire page is saved to the same folder where you placed the image or video it references.

If you followed the prior steps for tab management, these subfolders should match up with any strong leads that you have opened in their own windows. At the conclusion of your investigation, the digital evidence is nicely organized by its corresponding identifier and tools.

Passive Capture

The best example of a passive capture tool is Hunchly. It records pages loaded in Chrome at the source code level, as well as any images on those pages. It is providing a wide safety net, but you should be more intentional in taking advantage of its capture capabilities. The following steps assume you have Hunchly and Chrome installed in your custom OSINT VM. If you are not a Hunchly user, you may skip this section and move on to scripted capture.

- Create a new Hunchly case named the same as your investigative notebook and your digital evidence directory.
- Click the Hunchly extension icon on the top right of your Chrome browser and make sure it is set to capture and that it is set to the correct case name.
- Proceed with your research in Chrome as described in the previous sections. Any time you find an image that is key to your case, right-click it, select the Hunchly entry on the context menu, and choose "Tag Image". Provide a logical caption and click "Save".
- Hunchly can later generate a forensically sound report containing all tagged images.

Scripted Capture

Scripted capture is made up of the manually activated programs that collect or "mine" digital content on our behalf. A good example of this is using Instaloader to rip all of the photos from a specified Instagram account. These types of tools were covered earlier in the book, and there are only a few things to keep in mind on how they fit in our workflow, as explained below.

- For scripts that prompt you for a save location, you should use the same system that was described for manual capture: a series of logical nested folders.
- Some scripts will store collected data in a default directory, such as Downloads or Documents. In these cases, complete the collection and then manually move the files over to your case directory. When reasonable, move rather than copy files to limit clutter and abandoned case data.
- Add a line to your notes indicating which script was used, what it was directed to collect, and the date and time. Unlike Hunchly, most of these tools do not annotate or generate a log of their actions.
- If you are collecting digital evidence for use in court, you should consider also conducting a manual capture of any crucial items. The problem with scripts is that you may not be able to explain how they work in court. A manual save is easy to explain confidently to a jury when the time comes to testify.

Multimedia and Link Analysis

Whether you are working your case independently or have the support of a dedicated team, the research phase will include some level of multimedia analysis. This is the process of examining the visual and metadata characteristics of recovered images and video. Visual examination is exactly what it sounds like. View each image or clip at the highest resolution possible and methodically examine media for any intelligence that was unintentionally included in the frame. You are looking for things like business signage in the background of your target's profile photo. Identify anything that narrows down who or where they might be, and include this in your case notes. This process can be very time consuming, but it remains one of the best methods of locating an elusive target who has otherwise covered their online tracks.

Link analysis is the process of using charts and diagrams to illustrate how information and entities are connected. These can be people, locations, websites, phone numbers, or any other identifiers you see associated with online accounts. Figure 34.05 displays a link analysis showing connections between specific people and domains. You

will see specific examples of link analysis tools in the following chapter. Not all cases require a link chart, but you should consider its value when faced with complex organizations or anytime your case might benefit from a visualization of how entities or accounts are connected.

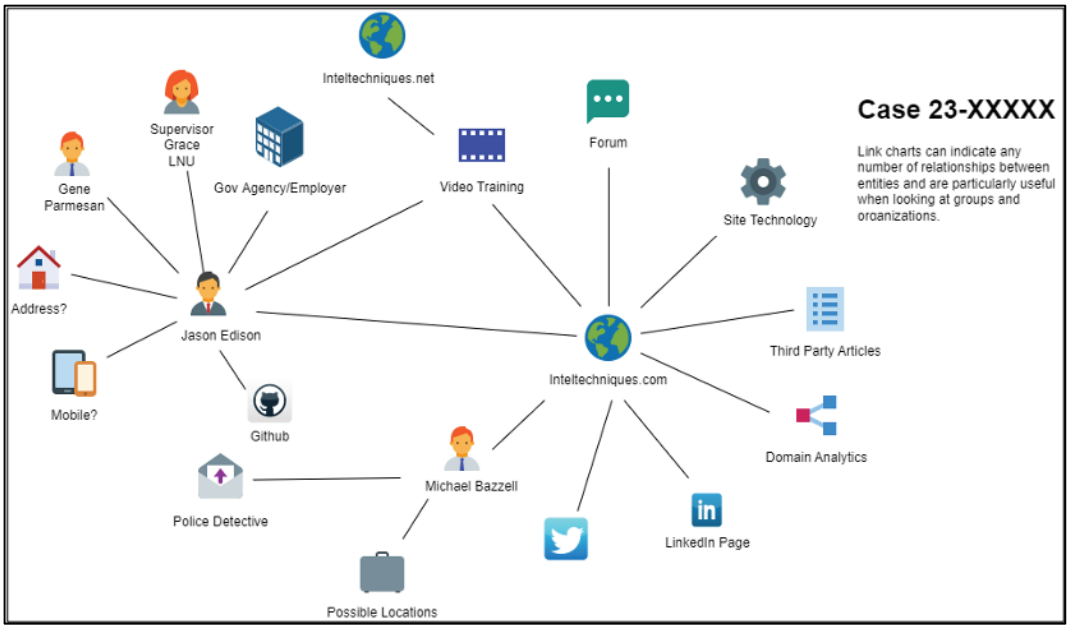


Figure 34.05: A link analysis example with Draw.io.

Once your research is complete, you will need to prepare your report. Several of the steps in this workflow were in preparation for the reporting phase. If you have followed along diligently, your efforts will be rewarded by painless report creation. Chapter Thirty-Five is dedicated to taking all of the intelligence that you have collected and using it to build a professional case report.

Submission and Cleanup

It is not unusual to move from one investigation to another very quickly. Just as we took time to properly set up our research, we also need to appropriately close out our work before moving on to the next task. The following may be beneficial.

- Transfer any handwritten notes to either your digital notes or final report. If you prefer, you can scan the notes as a pdf using your scan enabled printer or scanning straight to OneNote. Any paper notes or printouts are then either filed in a secure location and in compliance with your agency's policies or they are shredded.
- Do not leave case assets scattered about or they will get mixed in with future case work. Your investigative directories should be archived in accordance with your agency's evidence submission and retention policies. Some teams retain copies of these "working files" on network attached storage or optical discs for 3-6 months. If the subject should resurface, as criminals tend to do, having historical notes from previous incidents can be a huge time saver.
- If appropriate, export a copy of the VM that you used for the case. Then return to a clean snapshot or clone as described in earlier sections of this book. Consider preparing fresh accounts for the next investigation and find replacements for broken tools.

Order 0138471223

20 Minutes vs. 20 Days

Some major case investigations take place over a series of days or even months, while critical incidents may require you to give a threat assessment in 20-30 minutes. Your workflow will remain the same for each situation, but the time spent on each step will obviously be reduced. When you reduce the "time-to-solve" drastically, there will be compromises made to the quality of work and security. A common scenario where I use a streamlined workflow is a threat assessment, such as a person threatening suicide in an online chatroom. Consider the following threat assessment steps and workflow charts on the following pages.

- **Triage:** Verbally clarify the known identifiers and expected intelligence, such as: "User D1ckTraC on 4chan is threatening to kill himself in a post-we want to know who he really is, where he lives, and whether he is likely to carry out the threat".
- **Knoll Your Tools:** Grab your legal pad and a pen. Ideally you will have a fresh instance of your OSINT VM ready to use. If you do not have this prepared ahead of time, use "utility" social media accounts. Utility social media accounts are those on hand for non-criminal assessments where speed is essential and cross-contamination is a reasonable concession. Fresh accounts are always preferable.
- **Collaboration:** If you are collaborating with a team on a platform such as OneNote, create a page for each user to paste key findings so that you do not confuse one another. Keep in mind that OneNote in a browser syncs almost instantly, whereas several users on OneNote desktop will have syncing issues. Assign one person to keep track of everyone's progress and build out the big picture.
- **Premium/Government Resources:** Run your target through any commercial aggregators and government databases. These checks should run very quickly and return basic data with minimal effort.
- **OSINT:** Begin with Google searches such as: site:4chan.org "username". Then query your target's known identifiers through your custom OSINT tools.
- Only open very promising links in new tabs and visually scan each page quickly for words or images that jump out at you. The images results can be especially useful on time sensitive assessments because your brain can process them exponentially faster than text.
- For anything useful, make a note on your legal pad and leave the corresponding tab open.
- Be prepared to give a briefing at the deadline, even if it is just a situational report similar to "we've located and preserved the original posting, there's a history of similar suicidal threats from that user, but we do not know who or where he/she is yet".
- Take care of any additional collection, analysis, and reporting once the crisis has passed. At that point you will fall back into the normal workflow and documentation steps.

Target Flowcharts

I have conducted numerous OSINT training programs over the past few years. Regardless of the audience, I receive the question "Is there a standard process or workflow for each type of OSINT target?" at every event. My short answer was always "no". I had always looked at each investigation as unique. The type of investigation dictated the avenues and routes that would lead me to valuable intelligence. There was no cheat-sheet that could be used for every scenario. While I still believe there is no complete template-based solution for this type of work, I now admit that some standards can be developed. This section will display my attempt at creating workflows that can quickly assist with direction and guidance when you possess a specific piece of information. These documents are presented in six views based on the information being searched. Each example should be considered when you are researching the chosen topic. The categories are Email, Username, Real Name, Telephone Number, Domain Name, and Location.

Each example will try to show the standard path that I would take when provided the chosen type of data. The goal with my investigations is to get to the next topic. For example, if I am given an email address, my goal is to find any usernames and real names. When I have a username, my goal is to find any social networks and verify an email address. When I have a real name, the goal is to find email addresses, usernames, and a telephone number. When I have a telephone number, my goal is to verify the name and identify a physical address and

relatives. When I have a domain name, my goal is to locate a real name and physical address. The cycle continues after each new piece of information is discovered.

Each example will identify only the services used. It will not display the actual address to navigate to the website. However, every method listed within these charts is explained throughout this book. These documents do not contain every avenue that may provide good information. They only display the most beneficial resources at the time of this writing. Think of them as the obvious steps to take when you receive a target to search. The results of your queries can lead you to more places than can be displayed on a single page in this book. These are just the first priorities.

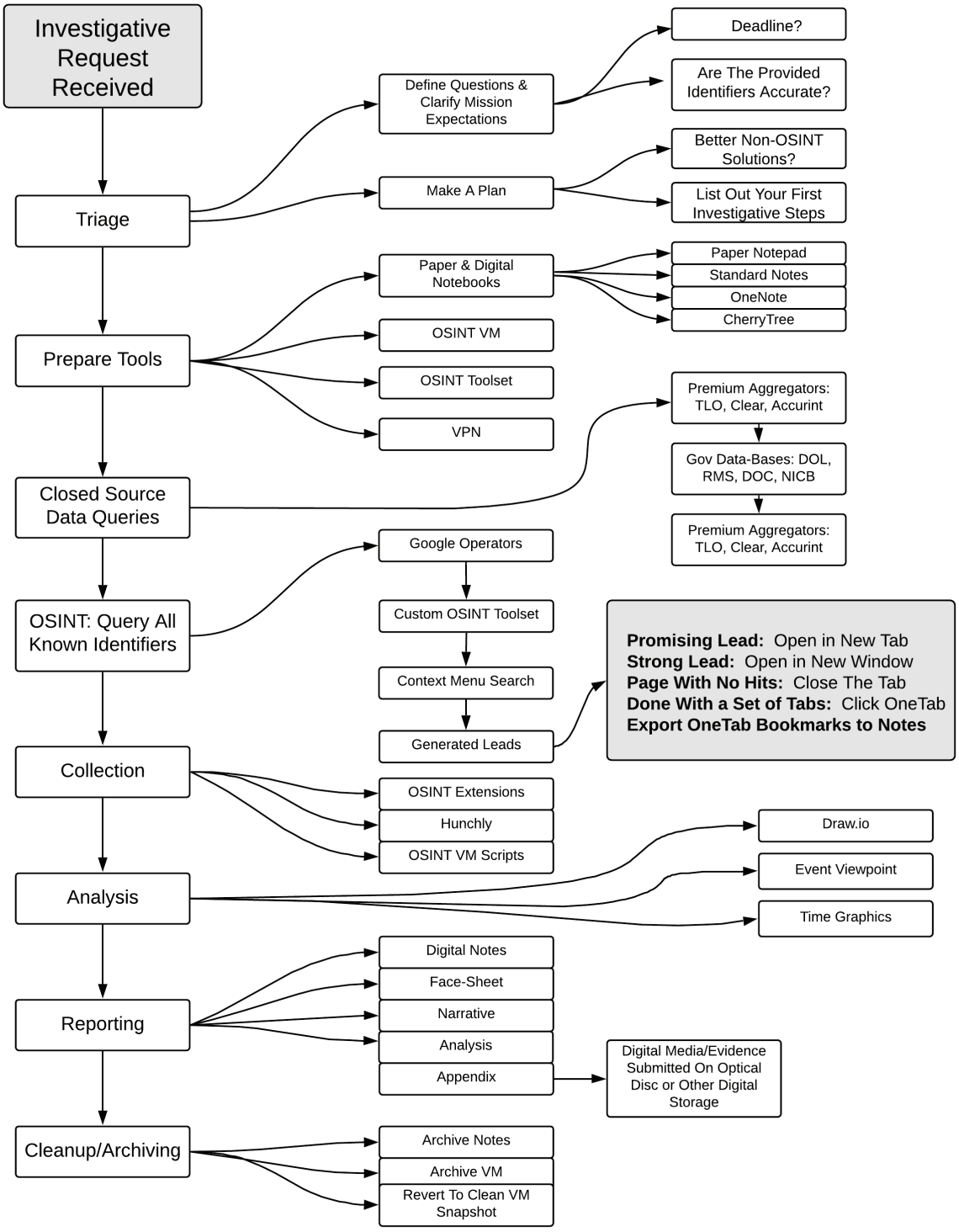
Consider the Email flowchart included with the digital materials accompanying this book (compressed view presented in two pages from now). The written translation of this would be to take the email address and search it within the EmailRep email validation tool. Next, conduct searches of the address within quotation marks on the main search engines. After that, check the compromised databases and all options on the IntelTechniques Email Addresses Tool. These options are likely to lead to a presence on social networks. Following these to the bottom of the chart encourages you to conduct the email assumptions previously mentioned, which you can verify and start over with the newly acquired information. You would then continue through the remainder of the chart. Download all flowcharts at **https://inteltechniques.com/data/osintbook10/flowcharts.zip**. This file includes full-size image exports of each flowchart and the original Draw.io file which can be modified.

I believe that all of these will always be a work in progress. As everything else in OSINT changes, these will too. I will try to keep them updated on the website. If you have suggestions, I am honored to receive and apply them. If you would like to create better formulas, I encourage you to get creative. I used the free open-source application Draw.io (draw.io) to create each of these. I installed the desktop application, but you could also use their free browser-based solution to open and modify the documents to fit your own investigations.

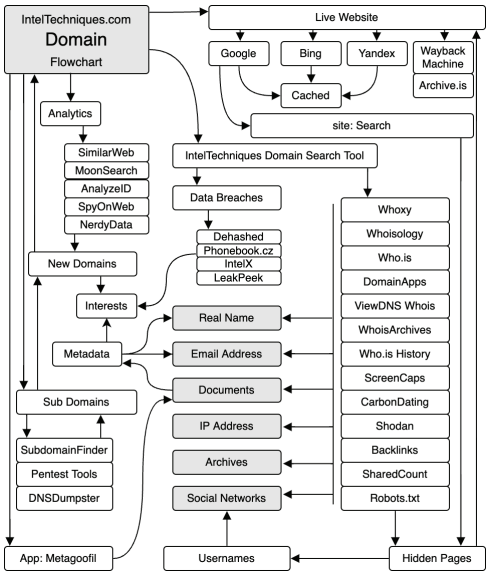
Many readers have requested practical exercises in order to test their OSINT skill. I agree that this would be helpful, but maintaining active and accurate online demonstrations with live data for public use can be overwhelming. Instead, I encourage you to test your skills with real data, unbeknownst to the target. Consider the following scenarios, and use the flowcharts here as a guide.

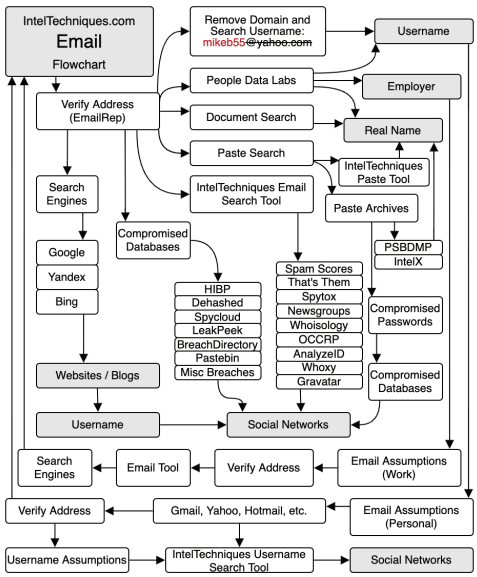
Zillow: Pick a random home and find all info about the previous owners.
Wrong Number (incoming): Reverse-search it, text them their details.
Spam Calls: Reverse-search incoming spam calls and texts to locate others receiving similar calls.
Wanted Criminals: Many Law Enforcement agencies have online databases listing known fugitives.
Friends & Family: Locate any significant others' online profiles with photos.
Waiter/Waitress: Research your server from dinner last night and identify their vehicle.
AirBnB: Locate all details about a host (owner) and email them directly.
Radio: Pick a morning "Happy Birthday" target, obtain full DOB and relatives' comments online.
Reviews: Find 5 people that have patronized a local business and locate their home addresses.
Game Show Contestant: Identify full address, phone number, photos, and relatives.
Newspaper: Choose a person quoted in today's newspaper and identify their social networks.
News: When a local Facebook comment is cited, explore the hidden data about the person.
Library: Locate an employee's Amazon wish list and buy them the book he or she wants (creepy).
Yearbook: Dust off an old yearbook and locate people from your graduating class.
Companies: Locate a profile on theorg.com and create an OSINT report on the CEO and board of directors.

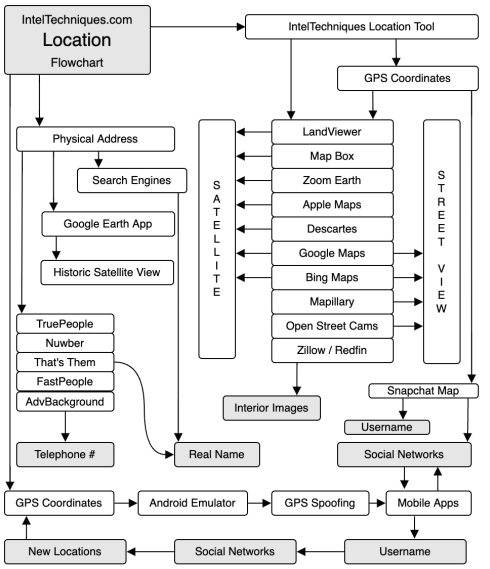
This list could grow for many pages. Overall, there are endless targets available that provide the best practice possible for exercising these techniques. This practice will increase the confidence in your research during an actual investigation. Members of the IntelTechniques online video training have access to a private server which continuously updates live real-time challenges for practice. The hard part is not disclosing what you find to them. While you may think they will be impressed with your new skills, they won't. Please trust me. Also, never share or publish any details about people.

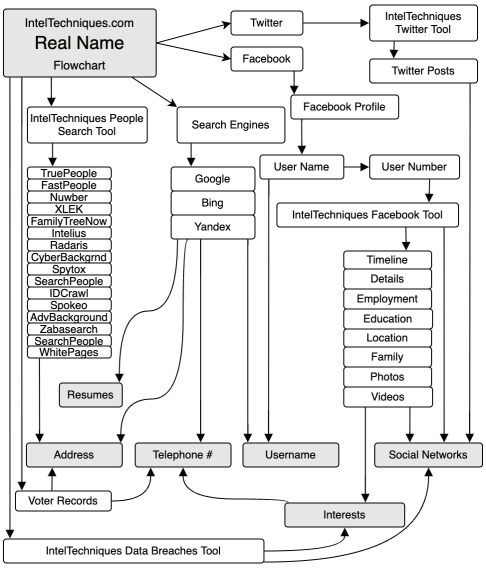


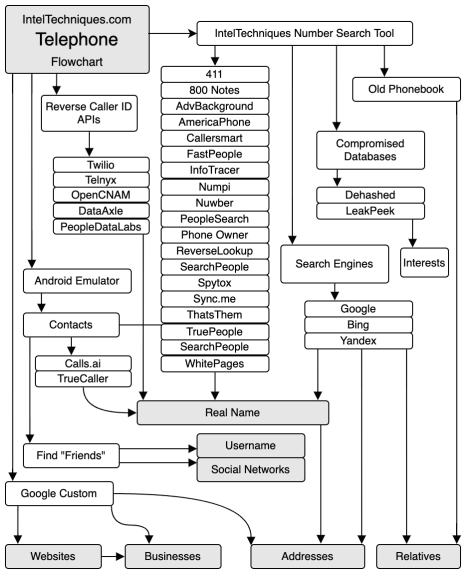
IntelTechniques.com OSNT Workflow Chart: Case Workflow

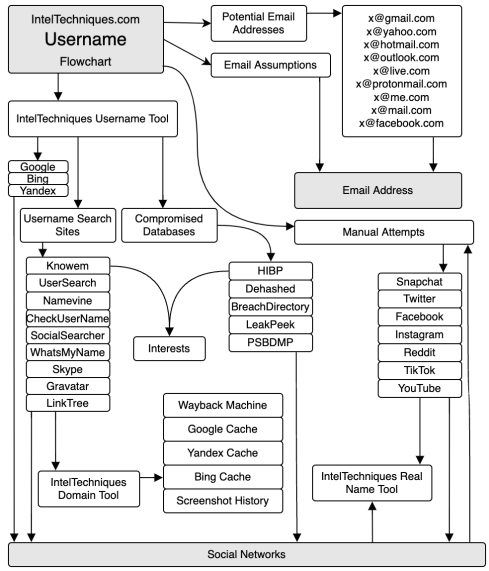












IntelTechniques flowcharts available at **https://inteltechniques.com/data/osintbook10/flowcharts.zip**

Order 0138471223

CHAPTER THIRTY-FIVE
DOCUMENTATION & REPORTING

Once you have completed your research, you will need to compile your intelligence into a professional report. Using the correct tools and techniques for documentation throughout your investigation will make generating your final report a relatively painless process.

Investigative Note-Taking

Like any form of research, note-taking is essential when conducting a professional OSINT investigation. As mentioned in the previous chapter, I prefer to take my notes in both paper and digital formats. The paper scratchpad is for quick identifiers, reminders, and at times a rough diagram. The digital notes are for pasting copied text, images, video, or other files. My hand-written notes tend to be unstructured, whereas collecting digital content often benefits from a pre-structured destination. The following are my recommendations for solid digital notebook applications for OSINT work, along with some of their respective pros and cons.

Standard Notes (standardnotes.org)

Standard Notes is the best privacy focused note-taking application with AES-256 encryption and has a very clear privacy policy. It has open-source versions for Mac, Windows, IOS, Android, Linux, and web browsers making it a great fit for OSINT work. You can use Standard Notes completely offline, but if you choose to make an account even the free tier supports sync and end-to-end encryption. The premium version adds advanced features such as multi-factor authentication, automated backups to your own cloud service of choice, and some aesthetic options, such as themes. Figure 35.01 displays typical usage.

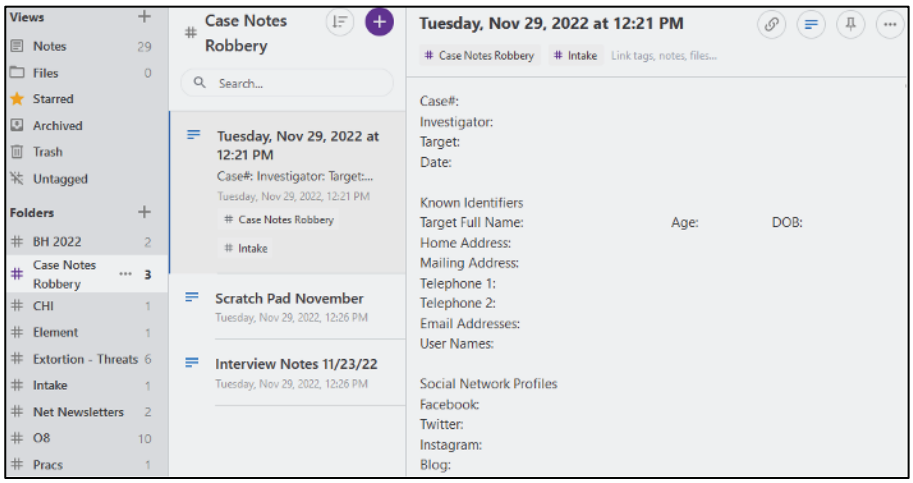


Figure 35.01: The Standard Notes application.

OneNote (onenote.com/download)

OneNote is the power option for digital note-taking at the price of a privacy compromise, but it is worth mentioning because many of us work for organizations that use Microsoft Office as their enterprise office platform. I recommend installing the standalone version of OneNote unless your agency requires use of the cloud-based Office 365 ecosystem. The default version is 32bit. If you want the 64bit build, look for "Other download options" at the bottom of the page listed above. Creating a fresh Microsoft account will prevent the installation from cross contaminating your other MS Office projects. Microsoft will offer to provide you with a new Outlook.com address, but a Proton Mail address would be the better choice for most people.

The desktop version of OneNote is supported on Mac, IOS, Android, and most major browsers, but each will require authentication with the account you used for installation. Microsoft, like other major platforms, forces you to sync to their ecosystem whenever possible. I have had little luck getting even the "offline" versions of their Office products to load without an internet connection. What makes OneNote unique is its ability to organize notes in a hierarchical structure, similar to pages within sections within notebooks. From an organizational perspective, it functions exactly like an old-school case binder of paper reports and printouts. I can create a new notebook titled "CopperThieves" with tabbed sections for each suspect. Each section is then broken down into my desired pages: Bio, Facebook, Twitter, Vehicles, Employment, etc. These pages can be populated from scratch, or I can preload them with my case templates or a simple spreadsheet.

Another feature that makes OneNote beneficial for data collection is its ability to accept just about any filetype as insertions or pastes into the pages. I can drag a pdf report from the Department of Licensing and drop it on the "Vehicles" page of a target and OneNote will ask if I want it added as a printout or file. The printout option adds a human readable image of the file, while the file option embeds the pdf file itself. I like to add both image and file by dragging in the pdf twice. That gives me a visual representation of the pdf contents while also saving a copy of the report in its native format. Double-clicking on that pdf file after import would open it in a browser window or pdf reader of choice.

I keep an original section prepopulated with my templates. For each new case, I right-click the section tab at the top of the window and select "Move or Copy". I select the destination notebook and click "Copy". If I select the same notebook where my original is, it will paste the copy into a new tab, but add a number to the section name, such as "Template 1". Now I can double-click that tab and rename it appropriately. I can also export entire notebooks by clicking on "File" and then "Export". Notebooks can be exported in a proprietary format, whereas sections and pages may also be exported as pdfs, docs, and several other file formats. This is convenient should you want to export sections from your digital notes to include in your report appendix.

Although OneNote can be used in an online capacity via browser from any operating-system, implementation on Mac and Linux is clunky at best. For MacOS users you may have better luck with a OneNote clone called Outline (outline.app/mac). Linux users have the option of a project called P3X (github.com/patrikx3/onenote) which will allow you to create, open, and edit OneNote files in your browser.

CherryTree (giuspen.com/cherrytree)

CherryTree is my note-taking application of choice for Linux. It ticks all the boxes we like such as being open-source and offline. What separates it from other Linux options is its ability to support both hierarchical notes and some limited support for storing images, tables, and other filetypes. The following steps will install CherryTree to your Linux Original VM.

- Open a terminal window by clicking on the shortcut in your Dock.
- Type the following into terminal: `sudo apt install -y cherrytree`
- Press enter and you will see CherryTree and all dependencies installed.

Open CherryTree from the Applications menu and add to your favorites, if desired. CherryTree has vast functionality, but the core feature of which we want to take advantage is the hierarchical node structure. Think of Nodes as notebook sections and SubNodes as pages within those sections. On the left side of the CherryTree window is the "node tree", which shows everything in your current notebook. To add a new section, click on "Tree" in the top toolbar and then "Add Node". Give your node a logical name such as "Facebook" and select "OK". To add a page or "SubNode" to that section, right-click on the node and select "Add SubNode". Name it appropriately and click "OK". Most of the functions we care about can be completed via the "Tree" menu in the toolbar or by using the right-click context menu.

You can create your own templates with the export feature. Once you have a notebook structure complete, click "Export" and then "Export To CherryTree Document". In the next window, choose "All the Tree" and click

"OK". For storage type, leave the top option selected and click "OK". Type in a name for your template, such as "OSINT_TemplateSept19" and click "Save". If launching a new investigation on a clean VM, the following steps will add your template within the application.

- Download or copy your template file to the desktop of your VM. The template file will have a file extension of .ctb.
- Within CherryTree, click "File" and "Open File".
- Browse to your desktop, select your template, and click "Open".
- Click "File" then "Save As".
- Leave the storage type as "SQLite, Not Protected (.ctb)" and click "OK".
- Type in your case name/number as the new filename and click on "Save".

CherryTree is not without its downsides. It is not nearly as user-friendly as OneNote when it comes to drag and drop and its support for inserting non-text filetypes is inconsistent. However, it is a hierarchical digital notebook within our Linux VM and one with much greater privacy than we could ever obtain using a Microsoft or other closed-source office application. There is a windows installer available as well, although I recommend using the Linux version in your VM whenever possible. A CherryTree template is included in the files download page.

Joplin (joplinapp.org)

Joplin is a free and open source note taking application which is supported on Mac, Linux, and Windows. Joplin is a great option for anyone wanting a simple digital notebook with an intuitive structure of notebooks and pages. Markdown format is supported for notes as well as the ability to create notes in a "to-do" format which adds a checkbox to the title in the navigation panel. Notebooks may be exported in markdown or HTML formats, with pdf export as an additional option for individual note pages. Although fairly simple by design, Joplin does support modern features such as search, tagging, plugins and small quality of life features such as dark mode. The following resources install various Joplin applications.

- Desktop Clients: joplinapp.org/desktop/
- Mobile Applications: joplinapp.org/download/#joplin-app-for-mobile
- Joplin Terminal Application: joplinapp.org/terminal/
- Plugins: github.com/joplin/plugins
- Browser clipper extension: joplinapp.org/clipper/

Joplin supports optional notebook sharing and synchronization via various cloud services such as OneDrive, S3, and their own premium Joplin Cloud service. The privacy-minded can take advantage of their self-hosting support with NextCloud or a Joplin server. If you choose to utilize any of the syncing options, make sure to enable end-to-end encryption and set a strong master passphrase at "Tools" > "Options" > "Encryption".

Cryptpad (cryptpad.fr)

Cryptpad is a privacy respecting online office suite with support for online collaboration. While it lacks the more advanced features of something like the Microsoft Office ecosystem, it is very simple to use, and it does not collect and sell your data. Your documents are encrypted by default on the Cryptpad servers. There are built in comment and chat features for real-time collaboration on shared documents.

- Cryptpad as a guest: cryptpad.fr/pad/
- Cryptpad account: cryptpad.fr/register/
- Documentation: docs.cryptpad.org/en/

As a guest you can create, store, and share documents for up to 90 days. Setting up a free account requires only a username/password and will net you an additional 1 GB of persistent online drive space for storing and sharing your documents. The team at Cryptpad is not collecting your contact information, which is great for privacy,

but they cannot reset passwords, so store your credentials appropriately. There is also a premium tier which includes more storage and faster support should you require either. As with any online service, we recommend maintaining a local backup of any files stored on the Cryptpad servers.

Advanced Text Editors

You may want to consider adding Interactive Development Environments (IDEs) to your arsenal of documentation tools. These are applications built for programmers and web developers for writing and testing their code. While full IDEs can be very complex, there are a handful of "light" IDEs or advanced text-editors which fit well into your workflow. The editors handle basic note-taking quite well. They also provide additional capabilities such as built-in terminals and mark-down support, which is exactly what we need when customizing your OSINT tools as previously explained. There are significant efficiency gains to be had when using a tool that supports note-taking, running scripts, and editing code all within a single interface. While many readers will continue to rely on office suites from major publishers to prepare your final reports, these light IDEs are capable of handling much of your investigative workflow.

The advanced text editors explained here were chosen based on the balance of functionality; ease of use; and their status as open-source applications. Each is available for macOS, Windows, and Linux which makes them ubiquitous across all major platforms. VSCode is very powerful with a large variety of extensions which add near limitless functionality, but at the cost of being tied into a proprietary Microsoft product. VSCodium strips away the Microsoft piece, but at the cost of some functionality. I recommend testing each and using the editors which feel right to you and make your workflow smooth. For those not ready to move to a more advanced editor, consider the previous recommendations for basic text editors and mainstream office applications.

VSCode (code.visualstudio.com)

VSCode is very close to being a fully functional IDE and it has a huge following in the programming and web development communities due to its extensive catalogue of available extensions. Extensions are plugins which add additional features and customizations such as interface themes. While the source code is open, the application itself is a proprietary free Microsoft product and its primary weakness is the standard telemetry and data collection that accompanies most Microsoft products. Therefore, VSCode is a good option for someone already working in a Microsoft enterprise environment. In addition to simple text capabilities, VSCode supports mark-down. This makes it a good tool for both notes and also for working on your custom tool files. It has a very responsive interface and does not bog down when dealing with large files. It has arguably the largest collection of extensions of any editor in its class and it supports running code should you want to customize some scripts. The standard install includes a terminal so you can run CMD, PowerShell, or bash from within the application. Real-time collaboration features can be added via the Live-Share extension. VSCode is available for macOS, Windows, and Linux.

macOS: `brew install visual-studio-code`
Windows: `choco install -y vscode`
Linux: `sudo snap install --classic code`

Following installation, immediately review all settings and disable telemetry. You can accomplish this by selecting "File" > "Preferences" > "Settings" or "Code" > "Preferences" > "Settings". Using the search box, submit a query for "telemetry" and uncheck "Telemetry: Enable Telemetry setting". Now search for "@tag:usesOnlineServices" and review the settings which make calls to online services such as those querying for extension updates. Disabling these will reduce your exposure to Microsoft but may limit some functionality such as automatic updates and notifications. Although the diverse offerings when it comes to extensions present a great deal of power and customizability, it can be a little overwhelming at first. I recommend the following extensions for note-taking and basic work with HTML and other common web filetypes.

HTML Preview (github.com/george-alisson/html-preview-vscode)
Default Browser (github.com/peakchen90/vscode-open-in-default-browser.git)
PowerShell (github.com/PowerShell/vscode-powershell.git)
Python (github.com/Microsoft/vscode-python)
Prettier Code (github.com/prettier/prettier-vscode.git)
Markdown PDF (github.com/yzane/vscode-markdown-pdf)

VSCode also offers a real-time collaboration extension (github.com/MicrosoftDocs/live-share). Data is end-to-end encrypted using the SSH protocol. In order for all users in a session to have read-write privileges, they must use a Microsoft or GitHub account. If you require live collaboration functionality, I recommend setting up a covert account on GitHub instead of using a Microsoft account. If you do not require real-time collaboration with team members, there is no reason to install Liveshare. There is also a web client available at vscode.dev, but be sure to disable telemetry via "Settings" > "Application" > "Telemetry".

VSCodium (vscodium.com)

VSCodium is an open-source clone built from the VSCode source. It removes the Microsoft data collection in exchange for a slightly more complicated configuration process. VSCodium offers most of VSCode's capabilities. If you are not already using Office or other Microsoft products, this is the best choice for you. Although the bulk of data sharing is disabled by default, it is recommended that you review all application settings and look for any which use "online services". There is no substitute for reviewing configuration settings yourself. VSCodium presents a slightly less polished user experience. Some extensions require additional installation steps and there is a much smaller community from which to draw support. VSCodium is available on macOS, Windows, and Linux using the package managers which were previously explained.

macOS: `brew install vscodium`
Windows: `choco install -y vscodium`
Linux: `sudo snap install codium –classic`

Anatomy of an OSINT Report

Although some investigators choose to write their reports as they go, I prefer to use handwritten and digital notes to complete the formal report at the conclusion of the investigation. I have created a series of templates in Microsoft Word and Adobe Acrobat, each of which are available in the digital files download, which provide a polished documentation framework for the most common mission scenarios. Although they vary in format, each contains a common structure including face-sheet, narrative, analysis, and appendix.

Face-Sheet

The face-sheet is typically one or two pages and is the most important piece of any report. It is meant to quickly convey the most important intelligence. The mark of a good face-sheet is that a boss or client can glance at it and instantly know the case name/number, who collected the intelligence, investigative timeframe, primary subject identifiers, and a concise set of key findings. The key findings can be in bullet or paragraph format, but should be no more than a half of one page. If you choose to include a table of contents for the report, it is often placed under the case title and prior to key findings. Figure 35.02 (upper-left) displays an example of a simple face-sheet while Figure 35.02 (upper-right) displays a full report version.

If you are limited on time or supporting a field operation, such as fugitive apprehension, the face-sheet may be the only form of documentation required. It is not unusual to receive "profile" requests while other investigators are purely looking for assistance in identifying accounts, addresses, phone numbers, and associates.
This also makes for a convenient quick reference sheet during future cases or incidents involving the same subjects. Events also get intelligence sheets which concisely address need-to-know details such as locations, hashtags, pertinent groups, and operational contacts. It should have everything you would want to effectively

monitor public social media posts and live video streams. Figure 35.02 (lower-left) displays an event intelligence sheet.

Narrative

The narrative follows the face-sheet and it tells the story. This is your opportunity to describe the path you took to reach your key findings. I like to organize my narrative in either chronological order or match the order of findings as listed on the face-sheet. I always prefer to tell a story in the order in which things happened. This can be vital if your report will be later used as part of your courtroom testimony.

Write in the first person and as concisely as possible, just as you would for any written statement for a police report or other legal document. Write just enough to provide context and a clear understanding for each piece of discovered intelligence. This section can be a couple of paragraphs to several pages, depending on the complexity of the case.

The narrative should address any discoveries that impact the investigative findings. It is important to acknowledge any exculpatory evidence and ensure that it is represented in your reporting. This is especially true for those working in the criminal justice and government sectors. Likewise, if your report takes the form of an intelligence briefing, your audience may expect confidence levels associated with your conclusions. Use terms such as "unsubstantiated", "likely", or "highly likely", rather than expressing the chances in percentages.

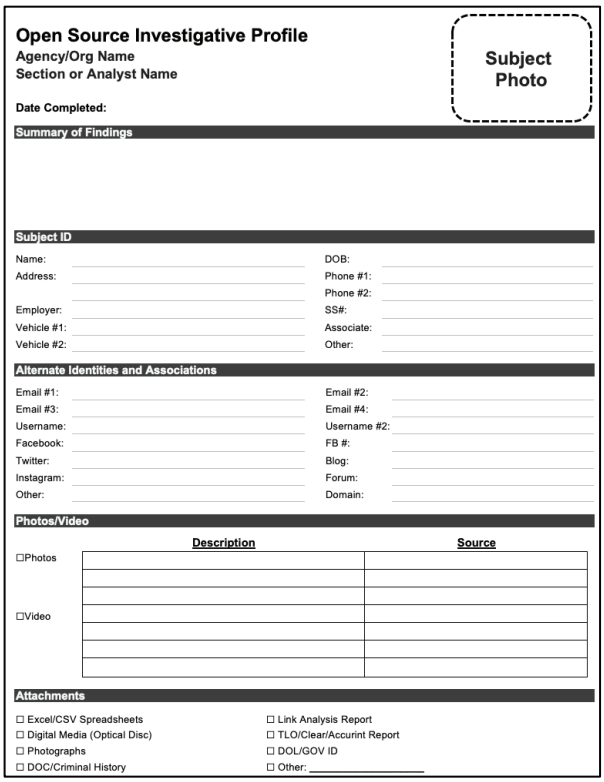
Consider adding a section describing the best-practices used during the online investigation. Key points to be made here are compartmentalization from other casework; thoroughness of research; software and services used; and a brief statement of training and qualifications. Think of this section as demonstrating why your work should be trusted. Figure 35.02 (lower-right) displays a partial example of a report narrative.

As a reminder, digital templates of all reports referenced within this section are available online at **https://inteltechniques.com/osintbook10/**. When prompted, enter the username of "**osint**" and your password of "**book1443**" (without quotes).

Link Analysis & Timelines

Not all reports will contain an analysis section, but this is where I present graphical components that support understanding of the preceding narrative. These include timelines of key events and link charts mapping out relationships between individuals, groups, locations, and internet sites. In my organization, this type of specialized work is often handled by civilian analysts who are proficient in software such as Maltego or I2 Analyst's Notebook. I provide them with a copy of my face-sheet, narrative, and a hand drawn map of the case entities and corresponding relationships. They use these to construct a more professional, visually appealing, and understandable graphical representation of the criminal organization or series of events. Not every investigation warrants this level of investment and not every investigator has these resources available, but they are a nice touch on major cases. When I do not have analyst resources available, I leverage one of the following free and user-friendly link visualization tools.

Draw.io (www.diagrams.net/integrations.html)
Visual Investigative Scenarios (vis.occrp.org/account/metro)
Gephi (gephi.org)
MindMup (mindmup.com)
NWU (knightlab.northwestern.edu/projects)



Open Source Investigative Profile
Agency/Org Name
Section or Analyst Name

Date Completed:

Subject Photo

Summary of Findings

Subject ID

Name: DOB:
Address: Phone #1:
Phone #2:
Employer: SS#:
Vehicle #1: Associate:
Vehicle #2: Other:

Alternate Identities and Associations

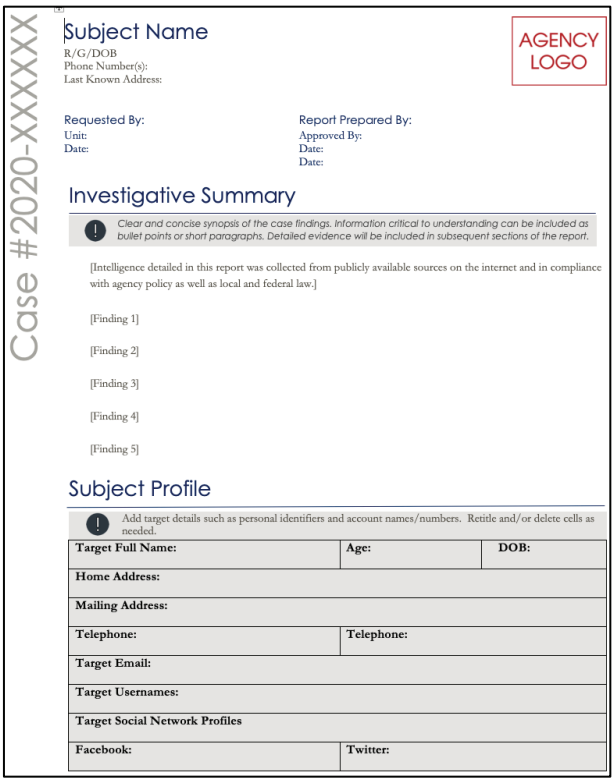
Email #1: Email #2:
Email #3: Email #4:
Username: Username #2:
Facebook: FB #:
Twitter: Blog:
Instagram: Forum:
Other: Domain:

Photos/Video

	Description	Source
☐Photos		
☐Video		

Attachments

☐ Excel/CSV Spreadsheets
☐ Digital Media (Optical Disc)
☐ Photographs
☐ DOC/Criminal History
☐ Link Analysis Report
☐ TLO/Clear/Accurint Report
☐ DOL/GOV ID
☐ Other: _____



Case #2020-XXXXXX

Subject Name
R/G/DOB
Phone Number(s):
Last Known Address:

AGENCY LOGO

Requested By:
Unit:
Date:

Report Prepared By:
Approved By:
Date:
Date:

Investigative Summary

Clear and concise synopsis of the case findings. Information critical to understanding can be included as bullet points or short paragraphs. Detailed evidence will be included in subsequent sections of the report.

[Intelligence detailed in this report was collected from publicly available sources on the internet and in compliance with agency policy as well as local and federal law.]

[Finding 1]

[Finding 2]

[Finding 3]

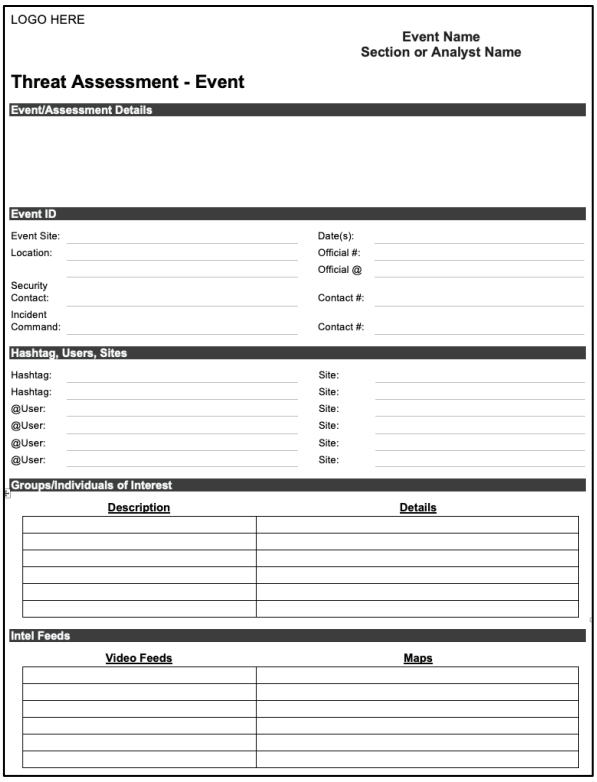
[Finding 4]

[Finding 5]

Subject Profile

Add target details such as personal identifiers and account names/numbers. Retitle and/or delete cells as needed.

Target Full Name:	**Age:**	**DOB:**
Home Address:		
Mailing Address:		
Telephone:	**Telephone:**	
Target Email:		
Target Usernames:		
Target Social Network Profiles		
Facebook:	**Twitter:**	



LOGO HERE

Event Name
Section or Analyst Name

Threat Assessment - Event

Event/Assessment Details

Event ID

Event Site: Date(s):
Location: Official #:
Official @
Security Contact: Contact #:
Incident Command: Contact #:

Hashtag, Users, Sites

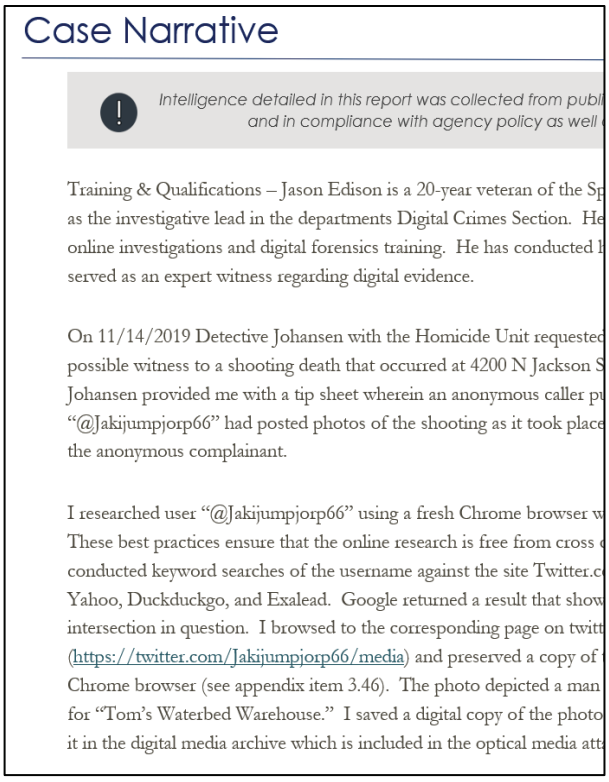
Hashtag: Site:
Hashtag: Site:
@User: Site:
@User: Site:
@User: Site:
@User: Site:

Groups/Individuals of Interest

Description	Details

Intel Feeds

Video Feeds	Maps



Case Narrative

Intelligence detailed in this report was collected from publi and in compliance with agency policy as well

Training & Qualifications – Jason Edison is a 20-year veteran of the Sp as the investigative lead in the departments Digital Crimes Section. He online investigations and digital forensics training. He has conducted served as an expert witness regarding digital evidence.

On 11/14/2019 Detective Johansen with the Homicide Unit requested possible witness to a shooting death that occurred at 4200 N Jackson S Johansen provided me with a tip sheet wherein an anonymous caller p "@Jakijumpjorp66" had posted photos of the shooting as it took place the anonymous complainant.

I researched user "@Jakijumpjorp66" using a fresh Chrome browser w These best practices ensure that the online research is free from cross conducted keyword searches of the username against the site Twitter.c Yahoo, Duckduckgo, and Exalead. Google returned a result that show intersection in question. I browsed to the corresponding page on twitt (https://twitter.com/Jakijumpjorp66/media) and preserved a copy of Chrome browser (see appendix item 3.46). The photo depicted a man for "Tom's Waterbed Warehouse." I saved a digital copy of the photo it in the digital media archive which is included in the optical media att

Figure 35.02: Document templates available at https://inteltechniques.com/osintbook10.

I formerly used the free, stripped-down version of Maltego called CaseFile for my link charts. I have since moved on to better open-source options that are far less resource intensive and less of a privacy concern. Remember, when working in our VM, we are borrowing resources from our host machines, so we need to use lightweight applications whenever possible. I prefer Draw.io for most investigations. It is most commonly used online as a browser-based diagram solution. When you browse to the site it will prompt you to create a new diagram, but first click on "Change Storage". Select "Device" and check "Remember this setting" to establish that we will be saving our work locally rather than with Google or Microsoft. Once you select either a blank diagram or one of the many templates, you will be ready to start building a link chart representing your investigative diagram. Before moving on, consider a more private option by installing the offline desktop application in your Linux VM. The following steps should be conducted in your OSINT Original VM before launching a cloned investigation machine.

- Navigate to github.com/jgraph/drawio-desktop/releases.
- Click the Linux "deb" link and download the installation file.
- Right-click on the downloaded file and choose "Open with Software Install".
- Click the "Install" button and provide your password.
- Open the application from the Applications menu and add to favorites, if desired.

Once the application is running, choose to create a new diagram, and a window will open offering you several templates. Some of the options in the network category work well for OSINT, or you can start with a blank project. Draw.io is very intuitive, allowing you to select icons on the left and drag them into your chart. Any linear icons (lines or arrows) are connectors and can be used to represent relationships between entities on the diagram. Double-clicking on text will allow you to edit the labels. The toolbar at the top and the right-click context menu offer many additional options for editing or adding components to the link chart. Save and export options are in the "File" menu. Draw.io supports export in the most common filetypes such as pdf, docx, png, and html. When you first install Draw.io the default settings will present charts which appear a bit outdated due to the art style of the standard icons. Changing the following settings can fix this.

- Click "Extras" at the top and select one of the additional themes. I use the lighter options, as seen in Figure 35.03, for anything being printed, but I find that dark works well visually.
- At the bottom of the "Shapes" panel on the left, click "+More Shapes...".
- Browse through and check any sets that look useful and click "Apply". One I always include is "Web Icons" under "Other". The "Web Icons" and "Web Logos" work very well for OSINT charts and the icon styles tend to be more modern than the default selections included in the offered templates.
- There is also a + button in your toolbar that will allow you to import your own images into the chart or even draw shapes freehand.

One small quirk that I have noticed when working with Draw.io in my VM is that when I choose to save my project the pop-up window is hidden behind the chart window. The easiest way to bring it to the front is to click the "Activities" button at the top left of the VM window which will show you all open windows. Click the file section window and it will pop back on top of all other open applications. The Activities button is a great way to find lost windows if you have several applications or terminal windows open at once.

Timelines and Event Maps

Draw.io is very flexible and you could use it to create timelines or event maps. However, I like to have multiple options for any task. The following descriptions are of two other applications for dedicated timeline and mapping tools.

Event Viewpoint (eventviewpoint.com)

Event Viewpoint is a free, browser-based timeline and mapping application. It will allow you to create events made up of a location, designated span of time/date, and event notes. You may add images and view your case as a list, timeline, or geographical map. An example is seen in Figure 35.04. You will need to sign up for a free account using a non-attributable email address. Event Viewpoint is not open source and does collect some user data so make sure to use a clean browser and VPN. I never use these types of browser-based applications to work with sensitive data.

Time Graphics (time.graphics)

Time Graphics is a premium browser-based timeline tool and you will need to make an account using a valid email address. Only premium users can save privatized projects, but you can select a two-day free trial. The interface is driven by an intuitive right-click context menu and will allow you to add events, pictures, video, and notes. Figure 35.05 shows an example. You can export your project in several formats including pdf, docx, and json.

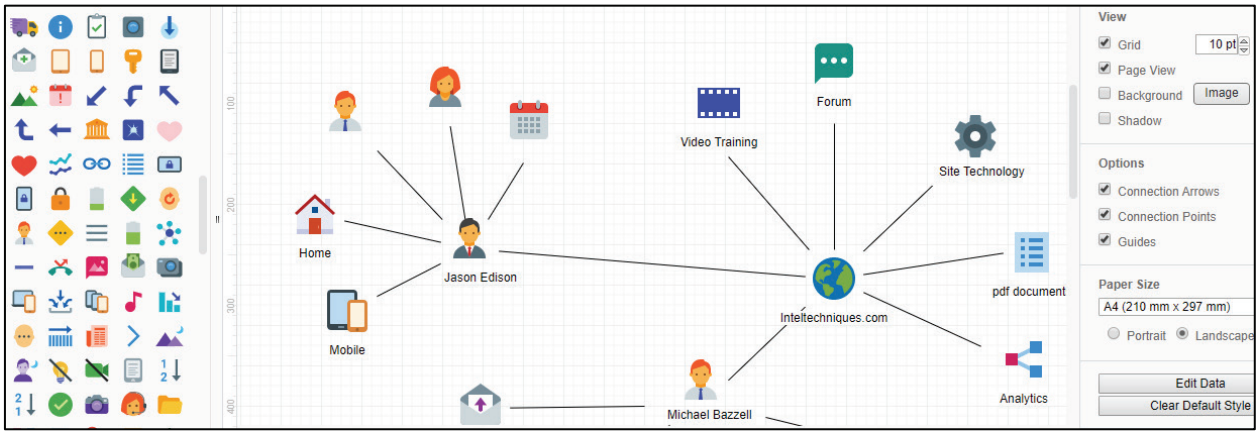


Figure 35.03: The Draw.io light theme.

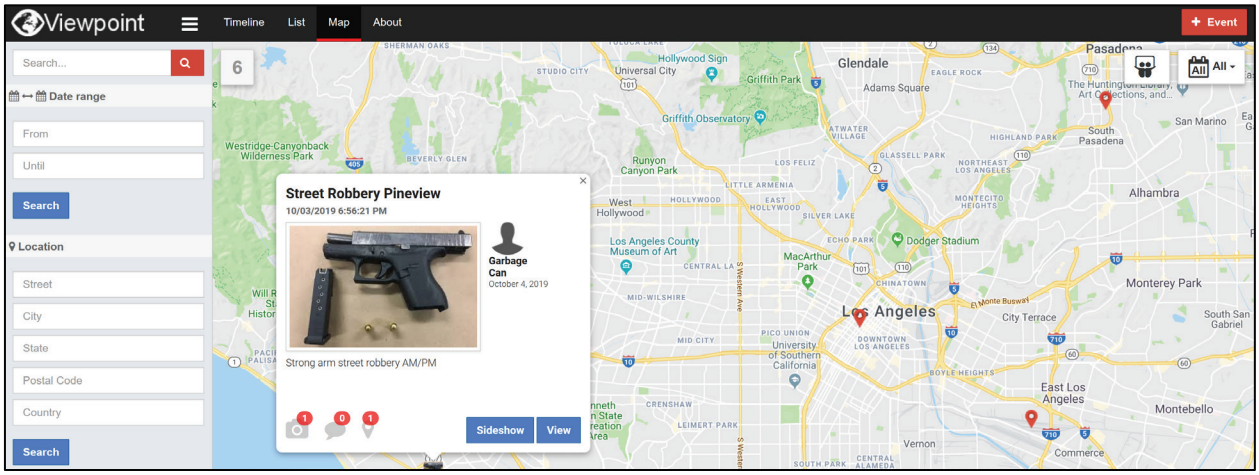


Figure 35.04: The Event Viewpoint timeline and mapping application.

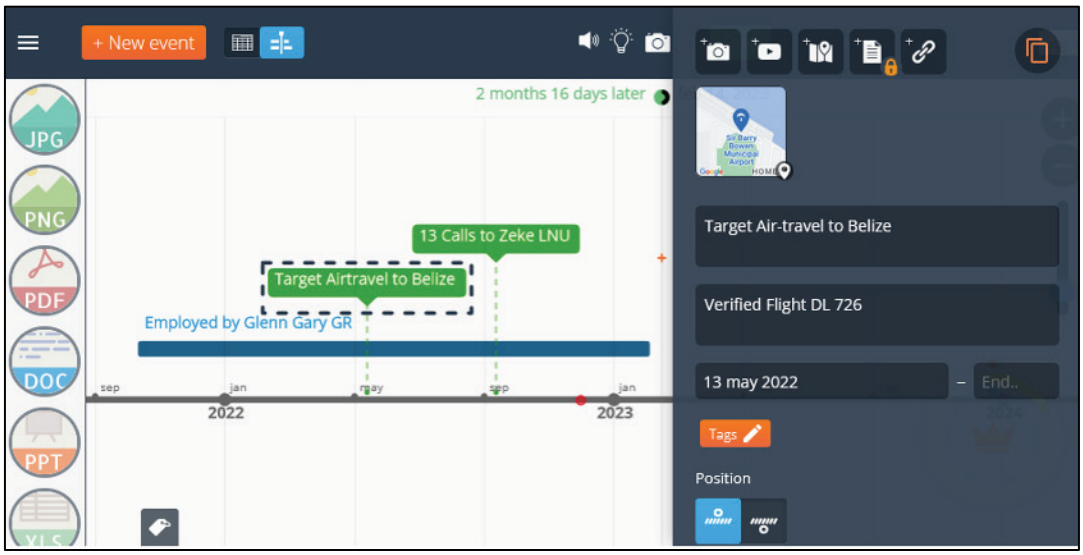


Figure 35.05: The Time Graphics application.

Appendix

The appendix of a report is made up of all the supporting documentation and media captured during the investigation. This will be screen capture files, still frames from YouTube evidence, spreadsheets of social media contacts, and any other digital evidence that can be displayed in a document format. These are working copies only, with the digital originals preserved on external media or other format approved by your agency's digital evidence policy. Many of your tools are going to generate pdf files which you will then need to include in your report appendix. If you have Adobe Acrobat DC you can easily combine the working copies of your pdfs into a single document for easier distribution. The free versions of Acrobat do not support combining or conversion of multiple documents. I prefer an open-source option that does not require an account with Adobe, such as the following.

PDFsam Basic (pdfsam.org/pdf-merge)

PDFsam Basic is a free and open-source application for merging multiple pdf files. It is available for Linux, Mac, and Windows, but the Linux version is the perfect fit for our investigative VM. Installation in Ubuntu Linux is straightforward, as follows.

- In your Linux VM navigate to pdfsam.org/download-pdfsam-basic/.
- Click on "Deb package for Debian based Linux distributions" to download the app.
- Right-click the file and choose "Open with Software Install".
- Click the "Install" button.

This installation provides a new icon in the Applications menu. Opening the application presents a user-friendly menu with many options. The "Merge" function will compile pdfs into a single file with the following steps.

- Choose "Add" or drag multiple pdf files onto the window.
- In "Merge Settings" > "Table of Contents", select "Generate from file names".
- Click "Run" on the bottom left side.

A new combined pdf document will be created in the same directory as the source documents. PDFsam Basic has many more features that will allow you to manipulate your pdf documents, such as removing or rotating a single page within a long document. The main menu is accessible by clicking on the house shaped icon on the top right of the application. To the right of that, the icon made up of three stacked horizontal lines will take you

to the settings page. Here you can change the default output directory, should you want to make it your case folder. The premium versions of PDFsam are closed source, costing roughly $60 a year, and primarily offer more options to edit and convert the documents into other filetypes such as docx. I find the free version sufficient for my needs.

Sourcing & Annotation

It is important that in our documentation we include annotation indicating the source of each piece of intelligence. The following are some of the most common methods for adding sourcing to reports:

- **Footnotes** – This is one of the most common methods of including sources in documentation. It has the added benefits of looking professional and also being intuitive.
- **Tables** – Tables make for easy organization and association between specific identifiers (accounts, usernames, etc.). If your report includes ratings, such as seriousness of data exposure or confidence levels, you may wish to color code cells and include a color key.
- **Embedded Links** – Embedding the links in the listed identifier is an option but is often not ideal because you lose that link if you print out a hard copy of the report.
- **Combination** – Footnotes can be included in tabled entries or the links to the source may be listed in the table directly depending on formatting preferences. Including embedded links in your tables along with footnotes or full URLs is an option that some choose. Just ensure with embedded links that you are not directing your client to any questionable sites that could expose them to threats or an adversary.
- **Appendices** – Your report appendix may include full captures, lists of sources, or both. Keep in mind that even if you include sources in your appendix, you may want to include them in the profiles portion of your report as well for convenience. If your client has to hunt through the appendix any time, they wish to see an exact source, they may find the report cumbersome to digest and make use of.
- **File Attachments/Supplemental Documentation** – Larger captures of raw pages, images, or videos may be provided in a zip file or other archive. This should be in addition to proper sourcing in the report, not in place of. Many clients will not be interested in digging through the raw data and we want the archive of captures to be a bonus, but at the same time we do not want them to have to dig though the raw data to find the source URLs.

Quality and Diversity of Sources

In addition to how we memorialize and present our sources, we need to also bear in mind the value of the sources themselves. We often say in our line of work that information does not become intelligence until it has context. Sourcing our findings adds context while also providing an avenue for further validation/verification. The sources may also lend to the level of confidence we have in our findings. Just how likely a certain assumption or belief is to be true. Whenever possible we should include multiple diverse sources for each data point in our report. For example, if we find our target's address on a people search engine such as truepeoplesearch.com, that alone would be a fairly low confidence finding as people search engines are notorious for inaccuracies and false positives. However, if we locate and properly source the address from a county tax parcel database, that will increase the confidence level and overall value of that particular finding.

Hunchly (hunch.ly)

Hunchly was mentioned in prior chapters and is one of the only paid OSINT tools I use daily. If you are a Hunchly user, the report builder can be used to create your appendix even if you are not planning to use Hunchly to prepare the entire report. The first step is to set aside the evidentiary copy of your recovered data. This is accomplished by clicking on "Export" and then "Export Case". This will organize the raw data into an archive which can then be transferred to optical or other storage media. We may want a visual representation of this data included in the appendix of our report, which can be accomplished with the following steps.

- On the main toolbar, choose "Export" and then "Open Report Builder".
- Title the report to match your case, such as the target name followed by the case number and add your agency logo if you choose.
- Selecting “Add Section” will open a new window which will allow you to select specific Hunchly capture data to add to your report, such as notes, screenshots, saved images, and results from any keyword queries.
- If you wish to add a list of all captured pages, select “Add capture history” on the top right of the Hunchly application window.
- On the top right, if you select “Export report”, a pop-up will allow you to export a PDF or Word document. Toggle on the optional table of contents if you wish, select the document type, and then click on "Export". This will open a "save as" window and allow you to browse to your case directory where you can rename it "Photo Evidence" and click save.

The primary advantage of using Hunchly to generate your appendix is that it will add the capture date, hash, and URLs to each preserved page and image. This is beneficial when using the appendix as a hard-copy visual reference for the full evidentiary case export. Figure 35.06 displays the Hunchly Report Builder.

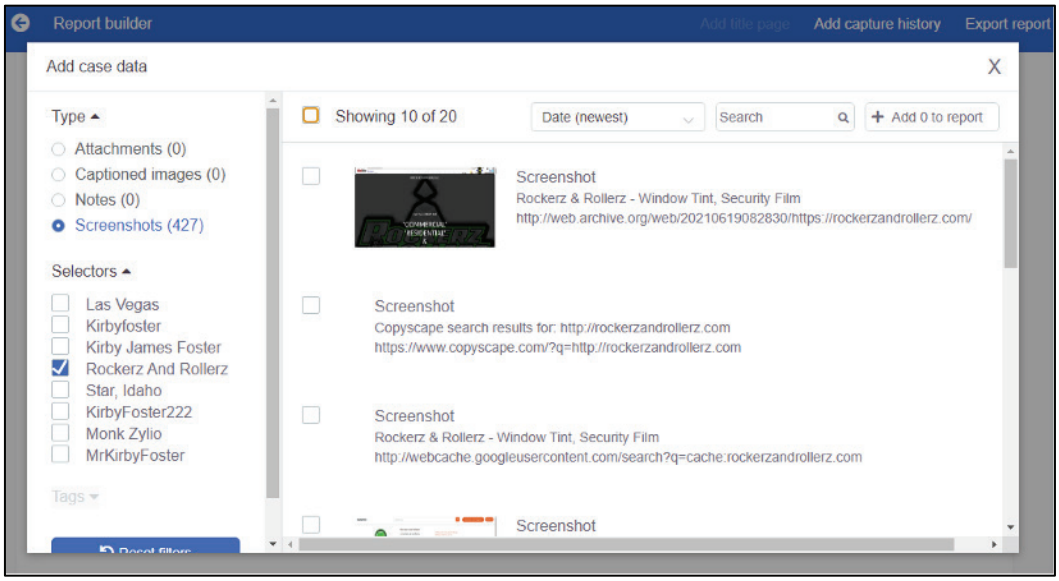


Figure 35.06: The Hunchly Report Builder interface.

Document Sanitization

Although I prefer to submit paper reports with an attached optical disc of supporting digital evidence, you may be constrained to the documentation procedures defined by your organization. If your agency uses any of the full-feature proprietary products such as Word or Acrobat, there is going to be metadata included in the digital version of your report that could unnecessarily expose your account details. While we would never want to remove metadata from evidentiary documents, removing hidden author data from your investigative report prior to submission is a recognized best practice.

To accomplish this in Microsoft Office products select "File" > "Info" > "Check for Issues" > "Inspect Document". This will open the inspector window with a list of possible metadata. Check all boxes and click "Inspect". This will execute a scan and any concerns will display a red exclamation point. Typically, these will be in the "Properties" or "Header, Footers, and Watermarks" sections. I often want to keep footnotes and do not remove this section. However, next to "Properties" click "Remove all". This will delete most of the metadata such as the username of the author. This is by no means a complete cleanse, but is an easy way to remove low hanging fruit.

To perform a similar metadata removal in Adobe Acrobat, click on "Tools" and under "Protect & Standardize", click "Redact". A new set of options will show up just below the top toolbar. Click "Sanitize Document" and a window will open with the options “Selectively remove” or “Remove all”. Clicking on “Selectively remove” will open a panel on the left side of the application showing a running scan of your document. Under "Results" you can see what types of data were found. You should uncheck boxes next to any items that you do not want removed. When you click "Remove", anything with a check mark will be deleted permanently from your document. The alternate option, "Remove all" will sanitize the document as if you had left all boxes checked for the various data types. I prefer the former method for most projects, as I like to have more granular control over what is removed.

Whether using the selective or remove all option, changes are not applied until the next time you save the document. To do this, click "File", then "Save As" and give your cleaned document a unique filename, such as Policy_v6_san.pdf. Whenever possible, I try to use naming conventions that convey the state of the file such as adding "san" or "clean" to indicate it has been sanitized.

In the toolbar, you will see an option to "Sanitize Document". Think of this as the express option for removing metadata as it will use a set of default removal choices to perform the same task that we previously did. I prefer the former method for most projects, as I like to have more granular control over what is removed.

It should be noted that removing metadata can potentially break formatting or corrupt the entire file; therefore, it is wise to create a backup copy prior to metadata removal. You can also use third-party scripts and tools to clean documents, such as the Metadata Anonymization Toolkit (https://github.com/jubalh/MAT) for Linux. I find that these tools are more prone to breaking documents created in Microsoft and Adobe products. Thus, for those types of documents, I use the built-in removal tools described above.

Keep in mind an old, but still very effective, method for removing metadata from non-evidentiary documents is using a scanner. Just print your report and scan the paper report back to a digital format using a scanner. The new document will have metadata based on the scanner/copy machine rather than your own accounts or workstation. If this scanner is an enterprise grade copier, it likely saves this data temporarily to a hard-drive and you would never want to use this technique with sensitive documents.

Case Archiving & Cleanup

Upon completing a report, I collect my analog and digital case work and archive it for future use. The length of retention of these notes, virtual machines, screen captures, and reports are dependent on the policies of my organization or the expectations of the client for contracted work. When appropriate I keep handwritten notes for six months and digital documentation for at least three years. Should the case be later adjudicated, I have the documentation available. It is also not uncommon to have a current target resurface in a future investigation; in which case those digital notes will give you a head start on that new case.

A final warning for users of OneNote or other digital notebooks. The search capabilities built into OneNote are very useful when querying subjects for any previous work your team has completed. However, hoarding data long-term can leave you with an awful mess if you do not have a system for archiving old cases. I have an archive notebook for each year and within it sections broken down by month. Any completed cases get moved to the archive, and at the end of each month I review any remaining sections to assess their status. Then, at the end of each year, I go through all my notebooks to tidy up any loose ends and consider purging any archived notebooks that have gone untouched for longer than three years. Another approach is to export older notebooks and save them to optical discs or other long-term storage. In your own agency, please ensure that you implement an organization system that works for your workflow and one that is also compliant with retention/audit policies.

Large-Scale Continuous Investigations

Many of my investigations span weeks or months. Most are extremely sensitive and must possess complete isolation from any other investigations. This goes beyond a clean virtual machine. They demand management of dedicated Facebook, Twitter, email, and other accounts, which can become difficult when providers detect my behavior as suspicious. In these scenarios, I assign a dedicated mobile device and cellular data account to each investigation. This may sound ridiculous and expensive, but we can provide this extreme layer of protection at minimal cost. My process is as follows, and only applies to investigations which cannot afford to be compromised by case-contamination or suspended accounts.

First, I need an unlocked mobile device. This will never be used outside of the investigation or for any personal use. I usually buy refurbished Android devices at local cell phone repair shops for $20 each. These are generic, low-powered, and overall undesired units which have very little value. You may also find similar new phones in grocery stores, pawn shops, or online through eBay or Amazon. I then purchase Mint Mobile SIM cards from either Amazon or mintmobile.com. These are $2.00 each, but include a $5.00 credit for service and a one-week free trial. Three months of service is $15 per month ($45 total), and T-Mobile is the data provider. I purchase the phones and service with prepaid gift cards or Privacy.com virtual cards.

I insert the SIM into the device; download the Mint Mobile application over Wi-Fi; register an account under an alias name; and start my trial. Since this is prepaid service, there are no verifications or credit checks. I choose a local telephone number issued by Mint Mobile which can be used for the countless verification text hurdles I am likely to face over the course of the investigation. When Facebook demands a real cellular number, I give this out freely. When Gmail blocks my account as suspicious, I can unlock it with a verification text. I no longer dread the suspension notices typically received when relying on VOIP numbers, VPN connections, and burner email accounts. This cellular number is my ticket out of most negative situations.

I can also place applications on the device when an emulator is not appropriate. As an example, Snapchat and Tinder usually block Genymotion and other virtual Android environments. With this device, I can install the native apps, launch a GPS spoofer, and conduct my investigation without roadblocks. My device appears "real" and I bypass scrutiny from the providers.

At the end of the investigation, I remove the SIM and place it and the phone in a sealable plastic bag with holes punched for use in a binder. These can be found in any office supply store. The phone and SIM are part of the investigation. The service will expire on its own and I have a clean digital trail. If necessary, I can provide the device and account details as part of discovery in court. I have no concerns if an expert witness wants to clone the machine for their own analysis. If the number should become exposed in a data breach, it is not a problem. It will never be used again. If you plan to replicate this technique, I advise preparing now. You do not want to be shopping for a device and waiting for delivery of a SIM card while you should be investigating your target.

CHAPTER THIRTY-SIX
POLICY, ETHICS, & DEVELOPMENT

There is much controversy over the use of social media and personal data for investigative purposes. Therefore, it is critical that your organization has a clear and concise technical investigations policy in place. This should be a one or two-page document and it must include at a minimum a training standard, an approval process, and an appropriate use policy.

Avoid using language tied to specific technologies or platforms, as those will change rapidly over time. In-house council may request language pertaining to Facebook or another specific third-party platform. They may recommend building policy around specific tools or technologies. It is our responsibility to demonstrate how the rapid changes in technology will make such a policy ineffective and almost immediately irrelevant. Moreover, limitations imposed by overly specific regulations will likely confine us to difficult options.

Responsible investigative policy should focus on the appropriate use of techniques and technologies. Review and borrow heavily from the mission statement and boilerplate language that your organization already applies to traditional investigative procedures. Any existing policy relating to training, equipment, enforcement, supervision, or chain of custody will have language that will fold easily into the framework of your online investigations policy.

Most organizations place online investigations policies within a section covering the overall social media policies. That section is usually broken down into a statement of intent, social media definitions, official use, personal use, and finally investigative use. I believe that operationally it makes more sense to group online investigations policies under the "Investigations" section of your agency manual. This is consistent with our argument that purpose and use should drive policy, rather than platform. Open source intelligence gathering is just another form of lawful investigation and we want to align it as such. An online undercover operation should be conducted and scrutinized in a manner similar to a covert assignment on the street. Comparing online investigative procedures with how the agency might use social media for public relations makes little sense and yet most organizations group them together.

I believe that for most agencies an online investigations policy should be no longer than two pages. The following is the basic framework for an appropriately non-specific policy. The statement of intent and definition sub-sections may be omitted if those are addressed within your general social media regulations.

```
SECTION 1.0 - SOCIAL MEDIA - INVESTIGATIONS - STATEMENT OF INTENT

[This statement articulates the importance of establishing guidelines for
appropriate use of online accounts, services, and software for the purpose of
online criminal investigations. It should be brief and reflect the mission
statement of your organization. Consider borrowing language from existing
policies related to appropriate investigative tactics.]

1.1 DEFINITIONS

[If not addressed elsewhere in the section]

Browser - Software which translates various forms of internet code into human
understandable formats including, but not limited to, plain text, graphic images,
and animations.

Private - Content is private when transmitted or conveyed in a manner that shows
reasonable measures and intent to limit access to specific individuals, and
where it is reasonable to expect that only those individuals will have access.
```

Public - Content is public when shared on a site or service in a manner wherein a reasonable person would expect that it is accessible to a broad or non-specific audience.

Post - Submitting information to the internet or a social media site.

Site - A page or series of pages hosted on the internet.

Social Media - Various online platforms, sites, and services facilitating the "posting" or sharing of information on the internet. Examples of social media content include text, images, audio, and video.

1.2 APPROPRIATE USE

Investigative accounts and tools will be used for agency purposes only. Investigative social media profiles, software, and services will not be used for personal purposes. Agency resources will be used in compliance with local, state, and federal law. Personal equipment, services, and accounts will not be used for investigative purposes.

Investigations will abide by all legal restrictions relating to private vs. public data and consistent lawful search and seizure. Where required by law, legal orders will be obtained from the appropriate magistrate and/or jurisdictional authority.

1.3 APPROVAL PROCESS

Social media accounts and online identities used in an investigative capacity will be reviewed and approved by the unit supervisor to ensure they are within policy. Ongoing casework will be periodically evaluated by the unit supervisor for compliance with agency policies and investigative best practices.

1.4 TRAINING

Personnel will only participate in the creation and use of investigative social media accounts after successfully completing an approved Covert Online Investigations training course. Unit supervisors are responsible for ensuring that all personnel utilizing investigative technologies are provided with appropriate training and supervision.

1.5 RETENTION/AUDIT

Data obtained during online investigations is handled and stored per the agency's existing digital evidence and retention policies. Where operational constraints necessitate unique data management procedures, the unit supervisor will review, approve, and log any non-standard measures along with a written justification.

All operational documentation and investigative logs are subject to review by the Office of the Inspector General [or appropriate body of oversight].

1.6 AGENCY FORMS (if applicable)

- Authorization & Training Verification
- Covert Online Investigations
- Unit Compliance Log - Online Investigations

1.7 ACKNOWLEDGEMENT OF TECHNOLOGICAL ADVANCEMENT

```
The state of the internet and social media technologies evolve rapidly, and it
must be acknowledged that the pertinence of overly specific policy will limit
the effectiveness of said policy. It is for these reasons that appropriate and
responsible use, rather than specific products or technologies, are the
foundation of this section.

1.8 REVISION HISTORY

*Legacy versions of this policy which are typically presented in reverse
chronological order.
Rev. 19.32 [Link to official archive if applicable] [Date of inception]-[Date
Rescinded]

1.9 APPENDIX

[Appendices are used for informational material that is helpful, but not directly
related to the implementation of the policy. These can be references to related
policies, procedures, or case law providing the foundation for investigative
best practices.]
```

For smaller teams or organizations lacking an official policy structure, an alternative format could be a set of standard operating procedures (SOPs). The content will be essentially the same, but less formal in structure. Take the previous policy example, delete the definitions and headings, and paste the remaining content into your agency's memorandum template. Then, have it approved by a commander or manager higher in rank than the front-line supervisor. The important thing is to have some type of documented standard for these types of investigations. Eventually a controversial incident or high-profile case will expose your team to scrutiny and a well-structured policy will go a long way to demonstrating transparency and professionalism.

If your organization requires language around "interaction", such as "friending", make this as non-specific as possible. Some agencies also restrict the use of covert social media accounts, which will extremely limit the effectiveness of your OSINT work. You have to educate management and legal advisors on the importance of covert accounts in building actionable intelligence. Decades of case law supports the use of "undercover" operations for traditional investigations and online investigations should be treated no differently. If you are forced to accept some restrictions related to covert accounts or infiltration, push for them to be contained within SOPs rather than agency policy. Continue to lobby for a reasonable standard utilizing appropriate concepts and language that is not overly specific. Common phrases of this type of language include "for lawful purposes", "consistent with agency training and standards", and "when feasible and reasonable".

For those in the public sector, specifically law enforcement, always be thoughtful regarding any reasonable expectation of privacy. U.S. courts have ruled consistently in favor of online undercover infiltration tactics based on the fact that a reasonable person knows that others on the internet may not be who they purport to be. Where we can get ourselves into trouble is when we cross over into what could be considered private communication. One example is recovering an unlocked smart phone from an arrestee and noticing that the phone is receiving social media messages. You would not want to pretend to be the phone's owner and engage in conversations without appropriate court granted authority, such as a search warrant. Another scenario is a victim allowing you to use their social media account for direct communication. Some jurisdictions consider any communication outside a "group chat" private and you may be required to obtain a legal order to continue any type of private or direct messaging using a third party's identity, even if it is with consent. Much depends on the jurisdiction in which you are working, which can get complicated when your cooperating victim and the suspect are in two different states, or in some cases, different countries.

I am not an attorney and the best path is always to solicit advice from legal counsel that is familiar with the local, state, and federal laws affecting your jurisdiction. Common sense goes a long way and I always ask myself if there is any way a target could argue a reasonable expectation of privacy. Additionally, being good stewards of

privacy is important when building trust with our communities. We never want to portray any hint of recklessness during intelligence gathering. Practice good documentation during the process in order to build favorable case law and maintain as many of our tools as possible.

Child Sexual Abuse Material (CSAM) Policy

In the course of our investigative work, there are times where we might locate content that is, by its very nature, illegal. If we are investigating crimes such as child exploitation, this might be intentional. We should be fully trained on how to appropriately document that digital evidence. Investigative units which do that type of specific work are less scrutinized than those who unintentionally stumble into digital contraband. These people often find themselves unprepared to deal with the ramifications. Those who work specifically in roles investigating child exploitation will already have more detailed processes and policies in place.

Although there is a range of digital contraband which could potentially fall into this category, the most critical issue to address in your process and policy is Child Sexual Abuse Material (CSAM). Within the open source intelligence community, specifically the crowd-sourced programs investigating missing or exploited persons, we see an increase of investigators locating material with a reporting requirement. I am not an attorney. I recommend investigators within the U.S. becoming familiar with **Title 18 Part 1 Chapter 110 Section of 2258A** of the U.S. legal code (https://www.law.cornell.edu/uscode/text/18/2258A).

There are a couple of important distinctions when it comes to CSAM in regard to semantics. Many laws in the U.S. and elsewhere still use the term "child pornography", which is widely considered as dated and inaccurate. Although you may need to use that terminology in the charging documents required by your jurisdiction, I recommend using the term "child sexual abuse material" in your notes and incident documentation. This, or the acronym CSAM, are generally accepted as the preferred terminology. First, let's address what you should do if you stumble onto content which you feel may be CSAM, or otherwise require reporting responsibility. The following are recommendations for those who do not already have a policy.

1) Do not capture it: We typically want to capture any online content for preservation when we find potential evidence of a crime. However, there are two primary reasons to avoid capturing CSAM. The first is to protect the victims. The last thing we want is unnecessary copies of harmful material being created, and possibly leaked, due to mishandling of data. Next, many jurisdictions define mere possession of CSAM images and video as a felony offense. Although your intentions may be noble, collecting this material could create legal consequences. If you are trained to handle evidence in these types of investigations, follow your defined procedure. Otherwise, it is best to only record URLs and other non-graphic material.

2) Notify a supervisor: Due diligence is essential in these incidents. One of your first actions should be notifying a superior to show that everything is above board. This can also ensure that all appropriate procedure and policy requirements are met. They may have additional responsibilities related to the scenario of which you are unaware. Even if mistakes are made along the way, we must show that we made every effort to act in good faith.

3) Report the incident to law enforcement: Reporting requirements will vary between jurisdictions, but ethical responsibilities for reporting are universal. Although your local law enforcement agency may not have the capabilities to investigate the incident fully, you will have created a trail of documentation which demonstrates personal due diligence and effort on behalf of your organization.

4) Report the incident to NCMEC: Report the content to the National Center for Missing & Exploited Children (missingkids.org/gethelpnow/cybertipline). Memorialize this reporting in your notes and consider saving a screenshot of your submission.

5) Report it to the involved platform: The timing and appropriateness of this will depend on the platform. If the platform itself is potentially involved in the criminal activity, we do not want to tip our hand. Also, we want

to make sure we notify law enforcement first so they have an opportunity to preserve evidence. Most platforms will retain digital evidence in CSAM incidents, but not all. Law enforcement will typically serve them with a preservation letter to request that an offline backup is maintained in the course of executing content removal.

6) Document the incident: Create a concise record to memorialize encountering the contraband material. If you are uncertain what to document, consider including the date, time, investigator, supervisor, URLs, and description of the contents. You want to convey an understanding of why you believe the material may be contraband.

7) Wellness: Even for seasoned investigators, exposure to CSAM or other materials depicting victimization can be very disturbing. There can be lasting personal emotional impacts if not properly addressed. It is entirely appropriate to speak with a professional counselor or other support personnel following an exposure to images, video, and even secondhand accounts of inhumane acts. It is important to take care of yourself, subordinates, and colleagues following a CSAM exposure, whether due to an investigation or an unintentional discovery while doing other online research.

The largest pitfalls in properly handling this material are in the reporting. Again, we do not create copies of the photos or videos unless we are trained to do so. We want to document enough information to direct authorities to intervene, and yet we want to be certain that we are not making things worse for the victim(s).

The previous recommendations are not a replacement for preparing a proper and well defined procedure for handling illegal digital material. Although I am not a believer in having policy just for the sake of policy, this is an area of potential liability. You would be wise to also consider a written policy reflecting and supporting your prescribed best practices. You might need it to defend yourself or a colleague against allegations of criminal or negligent behavior. If that occurs, a written policy shows your organization is professional and squared away. The following is a very general framework for a CSAM policy.

Sample CSAM Policy

```
SECTION 1.1 – REPORTING EXPLICIT MATERIAL

Employees in the course of their normal duties may encounter materials on the
internet which by their very nature demand action in regard to notifications and
proper handling. This policy aims to provide a baseline set of procedures which
support the proper handling of any incident involving illegal digital content
and/or specifically child sexual abuse material (CSAM).

1.1  DEFINITIONS

[If not addressed elsewhere in the section]

CSAM - Child sexual abuse material consists of any still, video, or audio data
depicting any sexually explicit conduct involving a minor.

Media - Visual or audio material stored in an analogue or digital format.

Minor - An individual whose appearance would lead a reasonable person to believe
they were under 18 years of age.

1.2  REPORTING RESPONSIBILITIES

Instances of illegal online content to include CSAM will be reported to the
appropriate legal authority. Additionally, employees will notify a supervisor
immediately upon discovering said content. The supervisor will document the
```

incident and ensure that material which is exploitive in nature is not further propagated.

1.3 POST INCIDENT PROCEDURES

Equipment exposed to sensitive and potentially illegal material will be examined by appropriate technology personnel to ensure that any digital contraband is removed and that the workstation or systems in question are appropriately sanitized prior to further use.

1.4 DOCUMENTATION

Personnel will not copy or otherwise reproduce any material that is believed to be child sexual abuse material. Every effort will be made to capture location information such as internet addresses, IP addresses, and/or domain names without including any media that could cause further exposure or harm to the depicted victim(s).

1.5 EXCEPTIONS

This policy does not supersede any policy or procedures related specifically and directly to investigative units with child exploitation investigations as a primary duty. Such units must have specific exemption granted by [Unit or Section Commander].

1.6 TRAINING

All employees are responsible for reviewing and acknowledging this policy. Any units assigned with online investigations as a primary duty will complete an additional body of training at the discretion of their unit commander.

1.7 REVISION HISTORY

[Legacy versions of this policy, which are typically presented in reverse chronological order]

Rev. 20.10 [Link to official archive if applicable] [Date of inception]-[Date Rescinded]

1.8 APPENDIX

[Appendices are used for informational material that is helpful, but not directly related to the implementation of the policy. These can be references to related policies, procedures, or case law providing the foundation for investigative best practices.]

Once you have a policy drafted it is wise to have it reviewed by your in-house council or other legal advisory body. Remember to provide training for employees on both the policy and procedures. A baseline of awareness will save you a fair amount of stress when an incident involving CSAM occurs during an investigation or intelligence operation.

If you are a student or non-professional OSINT practitioner, consider starting a conversation with your instructors and peers about the proper handling of illicit materials. If you join a crowd-sourced or charity event related to crimes against children, please ask the organizers for direction and education regarding appropriate procedures for responding to occurrences of CSAM or other digital contraband. As with most other aspects of online investigations, the key is having a plan and showing that a reasonable level of due diligence was applied.

Ethics

Privacy is certainly the cornerstone of ethical considerations related to open source intelligence gathering, but I will also be discussing issues of intent and deception. Privacy is a broad issue and will pertain to almost every portion of our case. Deception becomes pertinent when we engage in active covert measures such as undercover accounts or "friending" targets.

We each value privacy in our own lives and we should always be as mindful as possible in regard to the data we collect during our investigations. Early in an investigation, you should clarify the scope of engagement. How deep are you going to dig and what is the appropriate balance between purpose and level of intrusion? Here are two scenarios from the law enforcement world that represent opposing ends of the spectrum.

Scenario #1: I receive a report of online threats against a public official and am asked to make an assessment and if necessary, identify the owner of the account. I locate the account and posts in question and immediately see that although they are mean spirited, the comments do not articulate a true intended threat. A common example of this is similar to, "I hope you get cancer". It is a terrible thing to say, but there is no threat implied. At this point, I can report back to my boss that there is no threat. If necessary, I can show her a screenshot of the post. I have accomplished the mission and there is no justification to dig into the details of that person's life.

Scenario #2: We receive a tip from an internet service provider that a specific IP address is pushing traffic containing a large amount of child pornography. I investigate the IP address and find that it is a listed Tor exit node. That means the person operating the router at the associated residence or business is allowing people on Tor to funnel their internet traffic through their device. So now I know there is a fair chance that the person at that residence is not directly involved in the child pornography and may be completely unaware of its presence. I do not stop my investigation. I dig up every piece of public data that I can on the people controlling that router.

My decision to press forward on the second scenario is based on the seriousness of the crime. For a situation like crimes against children, you need to exhaust all reasonable means up until there is a certainty that the persons of interest bear no public threat. I am still obeying all laws, but it is far more reasonable for me to sort through the entirety of someone's public data if failing to do so might cause extraordinary harm to someone. The seriousness of the crime being investigated justifies a far broader scope of investigation and level of intrusion.

While some data on the internet is clearly public, there is questionable data that is publicly accessible and legally obtained, despite the owner intending it to be private. An example of this might be an Elasticsearch database that was improperly configured. If we can access it via a browser and providing no credentials, we should be in good legal standing. However, we are taking advantage of a mistake on behalf of the owner rather than something being intentionally shared. My feeling on these types of data sets is that our intentions make all the difference in deciding if it is appropriate to use them.

Examples of strong use cases would be defensive vulnerability assessments or investigations into crimes against children. We always want to balance the degree of impact on another individual's privacy with the greater good being served by the mission. If I am merely investigating a misdemeanor crime or gathering intelligence about what looks to be a peaceful community event, I am going to be far less aggressive in accessing any sites or data that were clearly made public in error.

In the workflow section, we talked about identifying your mission goals at the beginning of any OSINT engagement. Establish the reason why you are collecting intelligence on this target. This not only helps you to get organized in your approach, but also raises the question of intent. It might be that I am following orders or fielding a request from a colleague. The target might be related to a person of interest on a larger investigation. They might even be a future employee going through a background investigation. All of these are typical and reasonable business purposes for exercising our intelligence gathering skills.

I don't always know all of the details of a case, but I need to be able to articulate the reasonable operational justification for digging into a person's life. This is not a high bar, but just as I do not follow a stranger on the street for no reason, I too show care and consideration when conducting investigative work on the internet. A good rule of thumb is that any OSINT work that provides personal gratification is a red flag. We don't stalk exes or someone that caught our eye on the train, even out of innocent curiosity. Likewise, we don't cyber-stalk the guy in the car behind us in heavy traffic, who expresses his feelings towards us with a crude gesture. We are professional investigators and always leverage our skills in a professional manner.

One of the clearest examples of using OSINT for ill-intent is the practice of doxing. Doxing is using online research to profile an individual and then posting that information publicly on the internet with the intent of creating fear or embarrassment. The goal is to make the person feel exposed and show that you have power over them. It is cyber-harassment and it can even rise to the level of being a chargeable criminal offense.

Typically, my OSINT work is the result of a public safety event or statutory obligation, which allows for well-defined ethical grounds and clear justification. When I do run into "gray areas", it is usually when doing contract work for private sector clients. It is perfectly acceptable to conduct investigative work on behalf of a for-profit organization, but we do need to be clear on the intended use of any gathered intelligence. Often this is very straightforward. Common use cases are the vetting of vendors or potential employees. Another is vulnerability assessments on both the organization or C-level management. How the client intends to use our work product matters. We never want to be an unwitting party to a harassment or cyberstalking situation.

Deception is any behavior that misleads another person to believe something that is not true. This seems wrong when taken at face value, but there are occasions when deception is ethical and warranted. Take for example using social engineering to get your home address removed from a site-run data-mining company. You are doing no harm to others, but you are most certainly using deception to accomplish your goal.

The two most common forms of deception in open source intelligence are the use of covert accounts and the practice of infiltration or "friending" target individuals or groups. Let's look at each of these and their ethical considerations.

Covert Accounts: The use of covert accounts violates the "terms of service" of most social media platforms and online services. However, covert accounts are critical to successful queries into social media platforms, and not authenticating with these services would hamper the effectiveness of our work. If our intent is good, we have articulated the reasonable justification for the use of these accounts and stayed within the confines of our investigation, this deception will not be invasive to the other users. We are one of many anonymous accounts.

Infiltration: Infiltration, such as "friending" individuals or joining social media groups, is far more invasive than merely using covert accounts to run queries. When you join a group, you are, to a small degree, changing the dynamics of that group. When you interact with a target on Facebook or in a forum by commenting on their posts, you are directly affecting them, even if it is in a very small or positive manner. Just as we demonstrated when making a comparison of scope, infiltration has its place in an ethically conducted OSINT investigation. Joining a criminal forum to gain the confidence of, and to deanonymize, child predators is a case where the greater good significantly outweighs the level of intrusion. I do not like limiting these types of tactics by policy, as that type of framework tends to be too rigid. However, I do want to always be able to articulate my justification before interacting with targets online.

Finally, I believe that we have an ethical responsibility to engage in ongoing training in both OSINT and privacy. Lack of familiarity with both technique and legal requirements leads to mistakes. Sloppy casework does a disservice to us all. We owe it to the people we serve, whether they are clients or victims of crimes, our best efforts and discretion in making sure that we uncover the truth and bring it to light. A big piece of protecting our tradecraft is showing that we wield our tools and talents with restraint and thoughtfulness.

Professional Development

One of the most common inquiries I receive is regarding the establishment of a career in open source intelligence. This often comes from individuals in the private sector who are new to online investigations and those in the public sector looking to move into OSINT as a specialization. Although some concepts are universal, let's look at each of these scenarios. I will share some considerations for your new career.

Formal Education

If you look at practitioners in the open source intelligence field, you will find a wide variety of educational backgrounds. While there are universities that offer degrees and programs specific to analysis, intelligence, or investigations, I do not think you need to necessarily have that specific educational background. Many successful colleagues have degrees completely unrelated to this line of work, and many more received their education from the military or other organizations which provide internal training. Some employers will require a degree or a certain number of years of secondary education, but these prerequisites tend to be arbitrary. Many will accept work experience in place of educational requirements.

Some of the most important skills in this line of work can be non-specific to our field. The ability to organize information and communicate well, both verbally and in written form, are indispensable. There are very few assignments or contracts which do not involve generating some type of written report. You can be the best researcher and investigator in the world. However, being unable to package and present your findings in an understandable and professional fashion will prevent you from successful employment. Working to improve your written and verbal communication skills cannot be emphasized enough. Your resume is essentially an intelligence report representing your own capabilities. Make sure it is concise, well organized, thoughtful, and free of errors.

Experience

Experience is arguably the most valuable resume item and yet it is probably the most difficult qualification to gain. Many new analysts and investigators run into the chicken-and-the-egg issue. Most employers are only looking to hire experienced professionals, but it is difficult to gain experience until you are employed. There are several paths to gaining experience, but some will require managing your expectations or adopting a non-traditional approach. I have seen colleagues and students successfully build experience in the following ways.

- **Military Service:** Many colleagues received their initial intelligence training and experience in the military. It is also one of the only paths which builds education and real-world experience while also affording you a modest living. Military service may also provide opportunities for language studies and exposure to other cultures, both of which are very valuable in our field.
- **Law Enforcement/Gov**: Many LE and government agencies have units or entire branches which provide intelligence operations, investigations, or other OSINT products on a daily basis. The downside to this approach is that those types of positions often require working through the ranks. You may need to provide many years of service prior to being eligible for placement into one of these specialized teams.
- **Private Sector:** Although the intelligence units in larger firms are looking for specific experience and educational backgrounds, there are other departments, such as loss prevention, which can be a foot in the door. The requirements tend to be lower but still include opportunities to build and use investigative skillsets. These types of positions may not be your end-goal, but they are a step in the right direction.
- **Create a Program**: If you already work at an organization which does not have an OSINT function but could benefit from one, you might consider writing a proposal for adding that capability. Put together a demonstration for management and make sure to include success stories using OSINT. Think outside the box and understand that it will take time to gain traction.

- **Volunteer:** Find a person, team, or organization that does this type of work and volunteer to assist them whatever possible. If you can pass a background check, and you are able to work for free, unpaid internships are fantastic opportunities which can lead to full-time employment. This can often be at the same organization.
- **The Entrepreneur:** This is the path that I do not typically recommend. However, it is the most common. Starting your own company, or picking up a private-investigations license with little to no experience, is not likely to get you very far. If you are early in your career, you are better served surrounding yourself with experienced peers in order to benefit from their knowledge and learn from their mistakes.

Even if you are inexperienced, the ability to work as a team, eagerness to learn, and a strong work ethic are the qualities most employers seek in new hires. Wherever you build your experience, start creating a reputation as being someone excited to learn and willing to do menial tasks to support the mission. Early in your career, expect to do a lot of work that is outside of your area of interest. If you expect to be paid well to do interesting tasks, but you have little work experience, you are probably going to be disappointed. I have worked many assignments and roles which had very little to do with the intelligence field but were invaluable experiences. Be gracious, patient, eager to learn, and those disinteresting entry-level positions and internships will turn into something resembling a gratifying career.

Building a Program

Some of you may already be working in the intelligence or investigations field, but are looking to expand your capabilities. Others may be looking to make a pitch to your boss for more staffing and resources. Building an OSINT program or team can be challenging, but the following are recommendations based upon my own experiences.

The first step should be crafting a proposal which demonstrates the return on an investment from bodies and operating budget toward an expanded capability. The content of your proposal will depend somewhat on your specific organization and field. Consider the following.

- Articulate that an OSINT capability is an industry standard and most agencies are using specialized teams to increase efficiencies and success. A small team of well-trained investigators will contribute more than hundreds of untrained investigators.
- Articulate how increased situational awareness will reduce liability and risk. Field operators will have better intelligence for critical incidents which enhance the chances for positive outcomes.
- Propose a timeline and metric for measuring success.
- Estimate equipment, training, and staffing costs to begin the project. Consider utilizing low-cost alternatives such as surplus equipment. Early in my career, a fair amount of OSINT work was conducted on previously seized laptops and recycled desktops.
- Prepare a sample policy and standard operating procedures.
- Prepare a communications, documentation, and approval plan.
- Research location options for the team if it needs to be centralized.

Management tends to be most concerned about risk and cost. Your proposal should contain credible research to support the argument that public and private entities are employing dedicated open source intelligence teams to reduce risk and improve performance. Provide them with a prepared logistics plan and expect management to have reservations and objections. Convince your audience that they are behind the curve, but have the opportunity to lead the pack. If you are successful, control your expectations and understand that management will likely make their own changes to your plan. The goal is to establish a foothold. Understand that a fully capable OSINT unit will take time to develop.

Order 0138471223

OSINT Specific Training

Learning and honing any skill takes time. The length of time depends on your level of focus and interest, but this can be greatly influenced by the quality of learning materials. I believe that someone who is motivated can learn just about anything on a near zero budget if they are willing to devote time, attention, and dedication. Materials such as paid courses, and even this book, simply do the research for you and accelerate the learning process. Therefore, a good learning strategy takes advantage of the resources you have available.

For those with a tight budget, you should expect to do a lot of research and self-learning. However, there are many benefits to this. Autodidactic learning is the process of figuring out something or solving a problem on your own. Conducting your own problem solving can make you a stronger investigator. Performing investigations and gathering intelligence are often exercises in problem solving. When we are given all the answers, we do not strengthen those important areas of our brains. The beauty of open source intelligence is that it builds upon itself. We are learning to get better at conducting online research, which is a skillset that we can use to locate new techniques for improving our online research. This book, a browser, and determination can get you quite far towards building your own solid skillset.

Where do live courses, bootcamps, and other premium training programs come in? If you or your employer have a healthy budget for training, and you have less time to spend doing your own research, premium training may be worth the investment. For most people, live, hands-on training is the fastest way to learn something new or to improve existing skills. Live courses tend to be quite expensive. The area between live training and self-study is where online video training resides. It affords the opportunity to watch concepts and tactics demonstrated, but often lacks the hands-on labs common to live sessions. You can do a lot with a book and determination. Having a dedicated instructor and more materials just gets you there quicker.

IntelTechniques provides a variety of training tools, including this book which is affordable for almost anyone. For those who want to accelerate their learning curve, there is online video training at **IntelTechniques.net**. If you are looking for hands on instruction, we also offer live training sessions. Regardless of your budget and goals, make sure to do your research before settling on a training program. There are dozens of OSINT "experts" claiming to have the best courses and techniques. Use the research techniques in this book to get a peek at previously released content or to find communities where people are discussing the courses and instructors. Identify your goals, do your research, and execute the work to get where you want to be.

CONCLUSION

I hope the techniques presented here have sparked your interest in finding new avenues of research and investigations. With patience and diligent effort, this book will be a helpful reference guide to assist in performing more accurate and efficient searches of open source intelligence.

Permanently documenting these techniques on paper ~~may~~ will provide outdated content. Technology changes quickly and methods must adapt to keep up. Ten years from now, this book may be an amusing piece about how we once managed our online data. To keep up to date with the changes in various OSINT strategies, please subscribe to my blog at *IntelTechniques.com*; or monitor my Twitter (X) account at *IntelTechniques*. The chances are good that as you read this, new content has been posted about the very topic you are researching.

I am often asked my opinion about the future of OSINT. In short, YOU are the future, not artificial intelligence or online services. Occasionally, I am asked to advise intelligence collection companies during the creation of a new "all-in-one" OSINT solution. I decline these requests because the easy solutions are usually short-lived. Constant changes in technology, and automated data collection restrictions, make these commercial products less powerful over time. I do not believe these can ever replace your analytical brain. The truly valuable and powerful OSINT techniques are going to require manual work from an individual investigator. Your OSINT analysis skills cannot be replaced by a machine. Please go do good things with these methods, and never allow a program to become more beneficial than you.

With this book, I am passing the torch. YOU are now the search tool. As you find new resources and modify your files, let the community know over Twitter or other networks. When you configure a new Linux application in your custom virtual machine, tell us how you did it. When an online OSINT service disappears, I am eager to hear how you resolve the issue. I am truly excited to see how we all adapt to the impending OSINT changes sure to come.

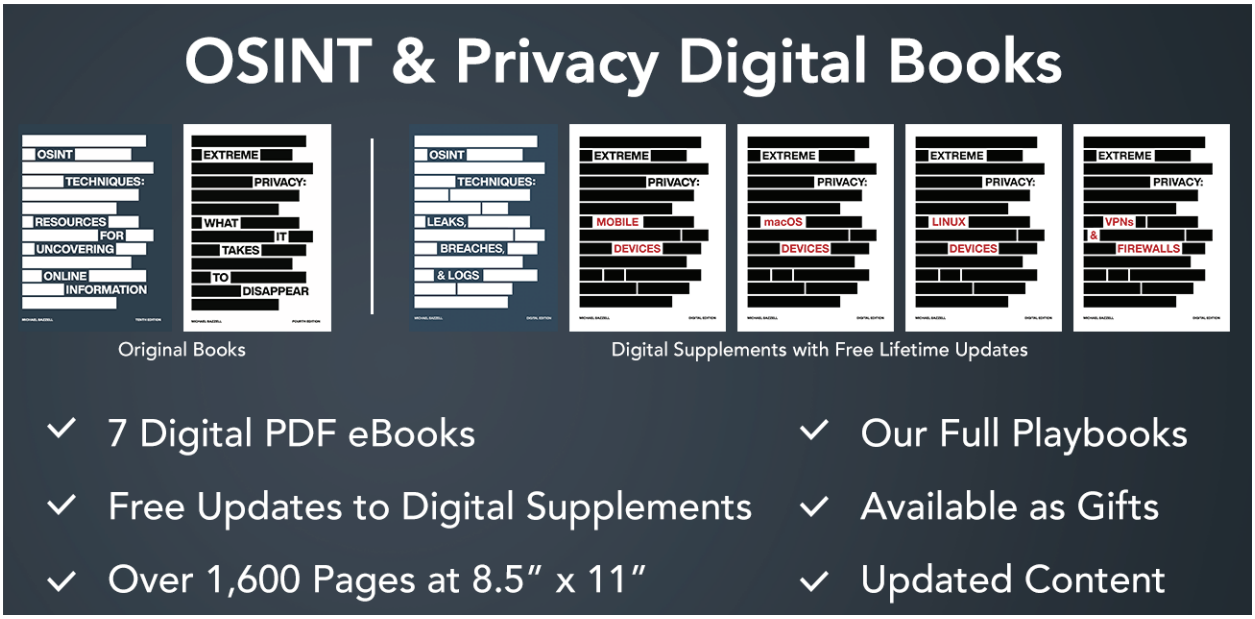
Finally, remember that each of these investigation techniques could be used against you. When you find your own information exposed, take action to protect yourself. Personal defense against OSINT is as important as offense. If you would like to stay updated on these topics, please consider my other series, *Extreme Privacy*.

Thank you for reading. ~MB

A special **THANK YOU** to my editors. You make me appear smarter than I am. This book would be a mess without your input and guidance. I owe you all more credit than I can possibly give within this closing thought.

Order 0138471223

The Next Level



We now offer seven digital eBooks related to OSINT and Privacy. More details (and bundle discounts) are available at **https://inteltechniques.com/books.html**. All five digital supplements (not the digital versions of the original print books) include free updates as things change. Your purchases support the research required to keep our methods current. Please consider an updated guide for yourself, or as a gift to someone else.

OSINT Techniques, 10th Edition: 36 chapters | 260,000 words | 550 pages | 8.5" x 11" | $30 - This textbook will serve as a reference guide for anyone who is responsible for the collection of online content. It is written in a hands-on style that encourages the reader to execute the tutorials while reading. The search techniques offered will inspire researchers to think outside the box when scouring the internet. Digital downloads include offline search tools, custom Linux scripts, and detailed report templates.

Extreme Privacy, 4th Edition: 22 chapters | 320,000 words | 517 pages | 8.5" x 11" | $30 - This rewritten privacy manual is PROACTIVE. It is about starting over. It is the complete guide that I would give to any new client in an extreme situation. It leaves nothing out and provides explicit details of every step I take to make someone completely disappear, including legal documents and a chronological order of events. The information shared in this book is based on real experiences with my actual clients, and is unlike any content released in my other publications.

OSINT Techniques, Leaks, Breaches, & Logs: 9 chapters | 55,000 words | 162 pages | 8.5" x 11" | $20 - This digital (PDF) supplement to OSINT Techniques presents a new approach to our tutorials. It is not a replacement for the printed book, but a much more thorough guide about Leaks, Breaches, & Logs. It provides our entire playbook which we use to locate, acquire, clean, store, and query various online data collections valuable to our investigations. We also explain all daily, weekly, and monthly tasks required to maintain your data collection. All updates are free and delivered digitally.

Extreme Privacy, Mobile Devices: 16 chapters | 65,000 words | 152 pages | 8.5" x 11" | $20 - This digital (PDF) supplement to Extreme Privacy presents a new approach to our tutorials. It is not a replacement for the printed book, but a much more thorough guide about mobile devices. It provides our entire playbook which we use for our clients when we need to issue new devices which must be private and secure. We also explain all maintenance and best practices for this new lifestyle. All updates are free and delivered digitally.

Extreme Privacy, macOS Devices: 10 chapters | 40,000 words | 111 pages | 8.5" x 11" | $20 - This digital (PDF) supplement to Extreme Privacy continues a new approach to our tutorials. It is not a replacement for the printed book, but a much more thorough guide about macOS devices. It provides our entire playbook which we use for our clients when we need to issue new macOS devices which must be private and secure. We also explain all maintenance and best practices for this new lifestyle. All updates are free and delivered digitally.

Extreme Privacy, Linux Devices: 10 chapters | 39,000 words | 101 pages | 8.5" x 11" | $20 - This digital (PDF) supplement to Extreme Privacy continues a new approach to our tutorials. It is not a replacement for the printed book, but a much more thorough guide about Linux devices. It provides our entire playbook which we use for our clients when we need to issue new Linux devices which must be private and secure. We also explain all maintenance and best practices for this new lifestyle. All updates are free and delivered digitally.

Extreme Privacy, VPNs & Firewalls: 9 chapters | 34,000 words | 88 pages | 8.5" x 11" | $20 - This digital (PDF) supplement to Extreme Privacy continues a new approach to our tutorials. It is not a replacement for the printed book, but a much more thorough guide about VPNs and firewalls. It provides our entire playbook which we use for our clients when we need to issue network-wide firewalls which must be private and secure. We also explain all maintenance and best practices for this new lifestyle. All updates are free and delivered digitally.