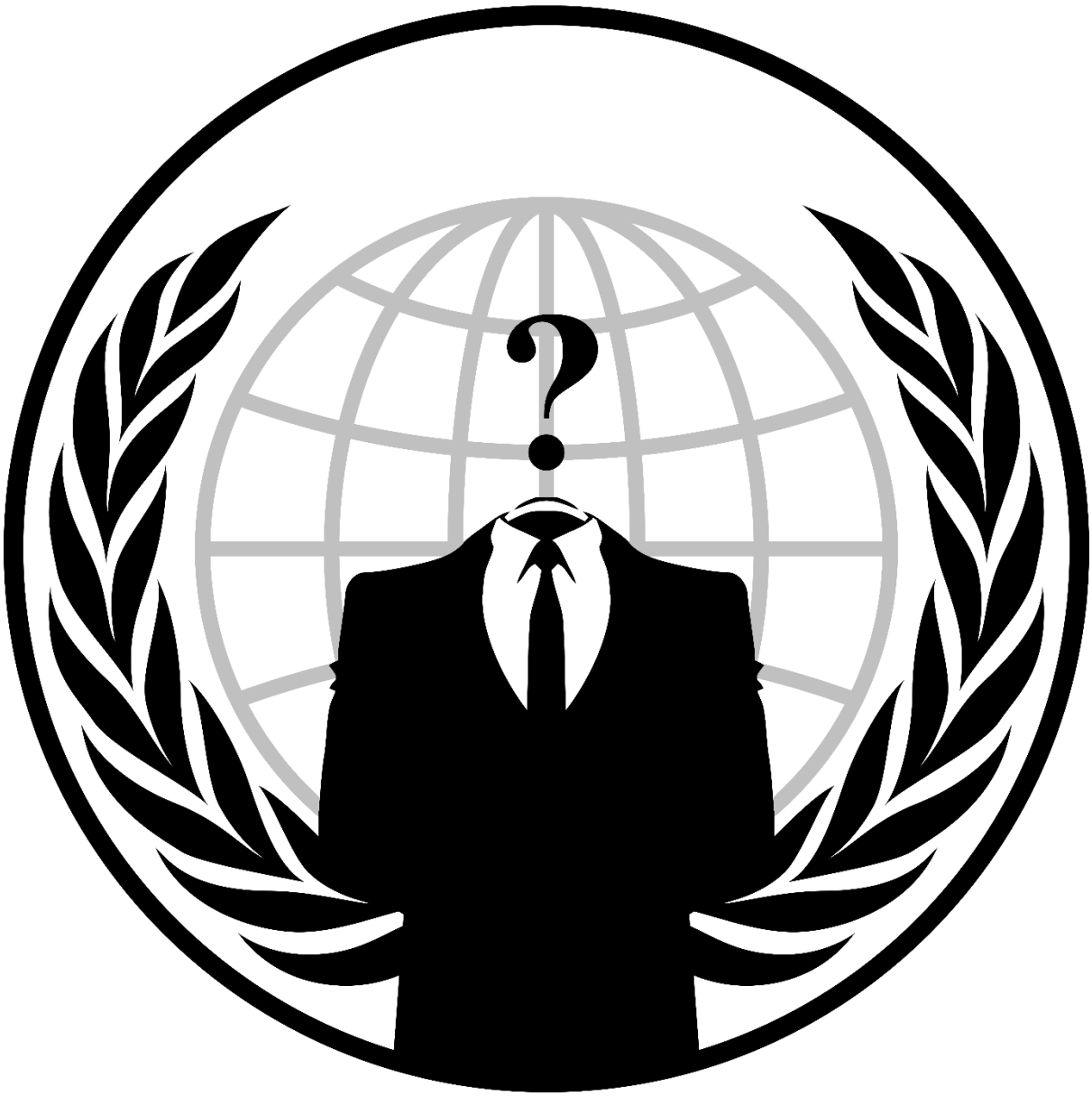

“There’s no patch for stupidity, or rather, gullibility,” Kevin Mitnick

Becoming a Black Hat Hacker

Mario Nascimento, a.k.a., darkArp

Index

What is a hacker

Definition

The term "hacking" often used to describe computer gurus with malicious intent that compromise computer systems and steal their data. However, hacking is much more than that. The original term refers to finding creative and unorthodox solutions to complicated problems. However, the preferred and current definition of this word is simply "to circumvent security and break into (a network, computer, file, etc.), usually with malicious intent "[1]. I underlined an important part of that sentence. Hackers don't necessarily have malicious plans. Let's look into this.

Types of Hackers

There are several types of hackers:

➢ Whitehat Hackers

These hackers are also called "Ethical Hackers." They work for companies or private individuals and with their authorization, attempt to hack into the company's servers. They then follow that attempt with a report stating what their server's vulnerabilities are, how they got in, what security implications those vulnerabilities have and how to fix them, as well as how much it could cost to do so.

➢ BlackHat hackers

These hackers are also called "Malicious Hackers," or "Criminal Hackers," or "Crackers." They do not ask for permission before hacking into other servers, and they do it so that they can disrupt the company's activities or steal information/money. They break the law for their personal gain.

➢ GreyHat Hackers

These hackers do not have a malicious intent, but they certainly don't always follow every law. They might break into computer servers without authorization but report back to the company revealing their vulnerabilities. They do break laws but not for personal gain, rather, they break them for fun or laziness.

What this article is about

In this paper, I will be disclosing all the steps involved in becoming a black hat hacker. It is important to understand how Blackhat hackers operate and think so that we can stop them. Not only am I going to talk about all the knowledge that a Blackhat requires but I'm going to provide concrete details of what I would, personally, do, buy, build, create, etc. if I wanted to become a malicious hacker. Furthermore, I am providing a demonstration of a remote exploitation against a fully patched Windows 10 x64 machine with Avira AntiVirus and MalwareBytes, performed and tested on the 15th of April, 2017, where I disable every security measure in place.

[1] http://www.dictionary.com/browse/hacking

Where to start?

I will be assuming the person becoming a Blackhat has no previous knowledge about hacking and only knows the basics of computer systems, much like an average person would.

The best place to start is learning. Learning should be done before anything else, or you are risking getting caught breaking the law unintentionally or creating bad habits and preconceptions that negatively affect your learning curve.

There are many ways you can learn computer systems, programming, and security free of charge on the internet. I will be linking *my opinion* about how you should be learning each skillset and the order you should be learning them in.

Since the person is becoming a Blackhat Hacker, there is absolutely no reason to follow the law, which means learning should be free. There are almost no penalties when it comes to downloading ebooks from torrents. Due to legal reasons, I will not be providing links to torrents of any particular ebook one would be able to find them very quickly. (I would just make sure I had an antivirus running since many torrents contain viruses).

Torrenting Websites

There are many different websites from which one could download torrents. The most popular ones are enough to get an aspiring Blackhat Hacker get started:

- http://www.torrentz2.eu
- http://1337x.to/
- http://extratorrent.cc/

Book-specific websites

- http://www.freebookspot.es/
- https://ebookee.org/

Best source for everything

- https://www.google.com

Don't forget that Google is still the best source of information in the world.

Books

1. *Computer Architecture, Fifth Edition: A Quantitative Approach (The Morgan Kaufmann Series in Computer Architecture and Design) 5th Edition* (ISBN-13: 978-0123838728)

I believe the best place to start is with the basics of how a computer works. Computer architecture is a crucial subject to learn before programming or networking. By knowing computer architecture, you can then grasp the concepts of programming much better, and networking knowledge will be easier to understand. There are many different books on computer architecture, but I would start with this one.

2. *Computer Networking: A Top-Down Approach (7th Edition)* (ISBN-13: 978-0133594140)

Next in the list is computer networks. Networks are one of the essential parts of information systems and must be learned.

3. *https://learnpythonthehardway.org/book/*

Python is, in my opinion, the first language one should start learning. It is very straightforward and easy for beginners. There are many good Python books. I recommend the one above.

4. *Linux For Dummies, 9th Edition 9th Edition* (ISBN-13: 978-0470467015)

Learning Linux is imperative, so I suggest you get started with it. This book had many positive reviews, for I believe should work very well to get you familiar with the UNIX environment. However, learning Linux should be more practical than theoretical, meaning you should mess around with it after learning the basics.

5. *C Programming: A Modern Approach, 2nd Edition 2nd Edition* (ISBN-13: 978-0393979503)

C should be the next language to learn. C is the building blocks of most other languages and allows you to code at a semi-high level while still being able to delve into the lower level.

6. *Advanced C Programming by Example – January 14, 1998*

Yes, C is just that important. I recommend not only sticking to these books but learning more. These are just to get you started.

7. *Hacking: The Art of Exploitation, 2nd Edition 2nd Edition* (ISBN-13: 978-1593271442)

This book is one of my favorites. It combines the knowledge of computer architecture and programming and introduces you to the world of computer security. This book is the on that will change the course of your learning from developer to a security expert.

8. *Unauthorized Access: Physical Penetration Testing For IT Security Teams 1st Edition* (ISBN-13: 978-0470747612)

This book is a bit outdated. However, it should get you started in network security as well as physical penetration.

9. *CEH v9: Certified Ethical Hacker Version 9 Study Guide 3rd Edition* (ISBN-13: 978-1119252245)
10. *Social Engineering: The Art of Human Hacking* (ISBN-13: 978-0470639535)

This book introduces you to Social Engineering in a very simple way but also includes advanced knowledge. It is perfect to cover the basics of Social Engineering, which is a key aspect of any hacker.

11. *The Hidden Dimension (Anchor Books a Doubleday Anchor Book)* (ISBN-13: 978-0385084765)

Another great book to further your knowledge of interpersonal relationships and psychology to better conduct Social Engineering attacks.

These books are just to get you started. As you progress through them, you will find many references to other sources of information and gain the required knowledge to search for other sources yourself and further advance your knowledge. I recommend continuing learning basic languages like HTML, Javascript, CSS, MySQL, PHP, .NET languages, C++, Java, among others. After that, you should try learning more and more about each and everything you already know. There's always more to learn about what you already think you know. Learning about virtual machines is also crucial.

The beginning

Now that we have the knowledge part out of the way let's get to the interesting part. So I want to become a Blackhat, and I already have the basic knowledge. How do I proceed?

Hardware and Software

The first order of business is deciding my hardware. I want to be able to get cheap and untraceable hardware but also powerful enough to do the business.

➢ Equipment:

- **Cannon Powershot G10 ($160 on Amazon):** This is one of the most used cameras for monitoring and reconnaissance. It's small, has an excellent quality-price ratio and is very discrete. You can select any other camera that does the job.
- **USCAMEL® Military HD 10x42:** Binoculars are also essential for surveillance, and I would purchase this one but any good quality medium price Binoculars would do.
- **Raspberry Pi(s):** Raspberry Pi is a requirement these days. It allows you to perform a broad range of attacks while being very discrete. I would purchase two Raspberry Pis. One of them I would build it as a TOR/VPN router (more on that later) and the other one I would change as necessary for different attacks.
- **Dell Latitude:** When it comes to actual computers there are many to choose from, and there's not a correct choice. It needs to have a high processing power, a lot of RAM, support USB and be a little bit discrete. It is needless to say that **a Dell Latitude** doesn't have much RAM or processing power. However, it is cheap, and it will be enough against many targets. I would like to spend as little money as possible in the beginning. After all, I am aspiring to become a Blackhat; I will just hack into people's bank accounts and gain some money to purchase a decent computer later on.
- **Alfa AWUS036H:** These are cheap USB wireless network adapters that support injection. These are very stable and work out-of-the-box with UNIX-based systems. It is the first choice for Wireless hacking. The important part is the chipset (RTL8187L).

➢ Operating Systems and configurations

As most people know, Linux is the preferred operating system for hacking. However, it is not the only one that should be used. I would personally install a Unix-based operating system (Arch Linux, Ubuntu or Fedora) set up with LVM encryption with a big encryption key (the one I currently have is 106 characters long). Set it up with LUKS disk encryption as well, with the self-destruct password set-up. I would use LVM on LUKS since that enables use to self-destruct very fast and also has the advantage of unlocking the LVM-volumes with one password. I would then install VirtualBox (or any equivalent software like VMware Player or Workstation) and set up Windows virtual machines and Kali Linux as well. I like Kali Linux because it includes many of the most used software for all types hacking, saving you the hassle of installing each one yourself.

Anonymity

There are many tweaks and settings to perform to enforce my anonymity. I would NOT use proxychains or any other software to anonymize myself. We will be discussing why later on when we talk about VPNs and TOR. I would use the Raspberry Pi I built as a TOR router to relay all traffic through Tor before it reaches the computer. Doing so ensures that no leaks happen. Using hardware is

generally more stable than using software. I would make sure ALL of my purchases were made using cash so it couldn't be traced back to me. If possible, avoiding purchasing at places with a lot of surveillance and/or avoiding cameras altogether and/or disguising myself subtly.

Using "Zombies."

To fully anonymise myself, I wouldn't rely solely on TOR (tor has been cracked and will be discussed later). I would use Zombies. A Zombie computer is a computer you have previously compromised and is under your control. You can then use it as a pivot to perform attacks on another target. You can set up a network of Zombies as your proxies to further anonymize yourself. It is safer to connect to your Zombies before connecting to the TOR network so that the Data transferred to your computer is less likely to get compromised. However, before you can compromise other people's computers you won't have any Zombies to connect to so relying only on TOR will have to suffice.

When using zombies, it is always good practice to wipe all logs clear and completely uninstall and clear all traces of yourself in at least one of the zombies. Following this rule will make sure that even if, by analyzing the connections of each zombie and reverse engineering the virus you used, they can trace back the other zombies, the tracks will stop after reaching the one you wiped. If one of the zombies have no record you were there or virus to reverse engineer, then there's no way to connect it with you. Doing this causes you to lose a zombie, but it's totally worth it since you can get many zombies per day using high-spread viruses.

Routers

I wouldn't want to tweak my personal routers since I don't want to be using them at all to perform my attacks. My real IP address is likely to get caught at some point in my life as a Blackhat Hacker, and if that translates into my home router, I am in serious trouble. What I want to do is use my wireless hacking skillset to compromise other people's routers or public hotspots, preferably far away from where I live but not too far, I wouldn't want to draw attention to the fact that I was traveling to a far location. These details aren't very important when I start hacking, but they become necessary after I perform hundreds of attacks. If the place of the attacker was pinpointed in ten of these hundreds of attacks and I was to draw any unwanted attention, the fact that I traveled far away to the places where the attack originated ten times would be too much of a coincidence.

Browsers

When any of my Linux machines I don't need to use any browser, to be honest. When I do need to use a browser (example, reconnaissance), I would either use the TOR browser or the Epic browser (both on windows virtual machine). I shouldn't need to use browsers for anything else other than researching my targets' websites or details. All browser attacks such as SQL injection and Cross Site Scripting can be performed with software like burp suite. The TOR browser and the Epic browser give the most anonymity by stripping away most of the features that track us (DNS cache, cookies, javascript, etc.).

VPNs, TOR, Proxies, etc.

Most people believe their identities are protected when using the TOR network and no-logging VPNs and free proxies and other anonymity tools. The truth is none of these tools guarantee you anonymity. They help, but all of them have flaws.

VPNs

First, we need to understand how a VPN works. A VPN or Virtual Private Network is a secure system you connect to through a secure tunnel. All the traffic between you and the VPN is encrypted. Furthermore, this tunnel adds safety towards hackers because it automatically reroutes if it detects a

breach. This makes it tough for an attacker to gather information about what you are sending the VPN or receiving from it. However, there's no protection when this data leaves the VPN into the Internet. Someone listening to the traffic after it exits the VPN will be able to capture all the data anyway.

There's another aspect to it as well. Let's talk about the two main types of VPNs:

- With logging policy
- With no-logging policy

VPNs with logging policy are obviously not to be trusted if you seek anonymity. The question is: Can you trust those with no-logging policy? The answer to this issue is personal, as there is no way to know for sure they do not log. I would never trust VPNs with this kind of policy because I just can't believe in their word. VPN companies are **companies**, meaning money drives them. They have the opportunity to earn A LOT of money by selling the information they log from you to third party survey and marketing companies. Why wouldn't they do that? They say they have a "no-logging policy" but who confirms that? Maybe it's just me who's paranoid, but I'd rather stay that way than to get caught because I was stupid enough to trust on VPN companies. Think about it, people who want privacy use VPNs. Why wouldn't the government make many of these "no-logging" VPNs to give people the notion of privacy while monitoring each and every one of them?

Still, sometimes you need to use a VPN. Let's say you are gathering information on a target and you realize that it only accepts IP addresses from a particular country. In this case, it is better to grow your botnet to include zombies from that country and perform your attacks from them. If that's not possible, and you can't travel to the country in question, or you're in merely in a hurry for some reason, and you must use a VPN I would use a paid version of VPN paid by a harvested PayPal account or a stolen credit card. Doing so minimizes the chance of me getting caught if the VPN company is subpoenaed. Even if I'm using a VPN, I will always connect to my network of zombies because if the IP address of the computer attached to the VPN gets caught, it will at least be the IP of the "exit zombie," for lack of a better term. Furthermore, following the aforementioned rules when using zombies[2], no one will be able to track your real IP by following the connections.

TOR

Let's understand how the Tor network or Onion Network works before we discuss its limitations and security flaws. When you connect to the Internet, you are connecting to your router, which then connects to the server of the website you want (in a very simplistic way). The server knows your IP address. However, the Tor network works by relaying traffic through nodes, which are volunteered by people. This means that your data is passing through multiple computers/servers with several layers of encryption before the request reaches the website you want to browse. There are three main divisions to make:

- The Entry node (the first node)
- The Middle nodes
- The Exit node (the last node)

The Entry node is the first computer you connect and to which you send your data. This machine doesn't know your final destination (the website you want to browse), but it does know your identity.

[2]" When using zombies, it is always good practice to wipe all logs clear and completely uninstall and clear all traces of yourself in at least one of the zombies.(…)", refer to page 5, category "Using Zombies".

The Exit node is the last one, the one which connects directly to the site you requested. This node has no idea who you are; it only knows the place you want to reach.

The middle nodes are there to separate The Entry and the Exit nodes. Each middle node only knows the information about who connected to them and who to connect with next.

The Tor network is in theory very secure. What happens, though, when a third party control the entry node you used, as well as the exit node? Well… It's pretty easy to figure out who you are. Many law enforcement agencies have tried to control a big portion of Tor's nodes in hopes of being able to track down Tor users. The most notable ones have been located. Tor has been cracked, meaning it's not safe. It is still useful because not only does someone need many resources to track you down but it requires time. By the time the law enforcement agencies get the IP of the "exit zombie," which connected to the entry node of Tor, you, hopefully, are already done with your exploitation and have covered all your tracks.

Jondonym and other proxies

Every other proxy service suffers from the same problems as VPNs. We don't know who's monitoring them. I wouldn't, therefore, use any proxy service or free proxies.

Anti-Virus/Firewall/Security measures

A big part of being a successful Blackhat Hacker is not to get hacked while in the middle of an attack, or at all. That is why all the attacks should be performed under virtual machines. While you are conducting these attacks, your computer is more vulnerable because you often need to disable security measures to be able to utilize some attack vectors. The primary system being Linux you shouldn't have trouble with viruses, malware or any security threat to your system provided the user configurations are correct. Using hardware or software security measures such as firewalls are not practical because it draws too much attention to yourself and takes up space (in the case of hardware) and gets in the way of your attacks. If my Windows or Kali Linux machine gets compromised while I am performing an attack then so be it. There's no way (given proper configurations) that my hacker can compromise my host machine. Plus, it extremely unlikely that such occurs because that would mean the hacker would need to trace all the connections through the proxy network and my zombies to get to the virtual machine before I cover my tracks. That takes longer than a normal exploitation would last, not to mention that cracking the TOR network can only be done by very few parties.

Tools

When hacking, I will use any tool that gets the job done. It is usually better to craft your own tools. Creating your programs has the advantage of doing exactly what you want, without limitations (other than your capabilities). You don't have to wait for updates; you don't have to ask for features; you don't have to ask to make certain features optional because they use up too much memory; etc.

However, if there's a tool out there that does what you want, there's no need to code one that does the same thing. No one needs to reinvent the wheel. Your goal should be to build a better wheel or use the already-created one if it that's all you need.

I will be listing valuable tools that I would consider using:

- **SQLmap:** This tool is splendid for SQL injections. However, this should only be used either if you aren't able to do manually perform the attack, or if the attack can be performed faster by using it or combining the tool with other ones (like burp suite).
- **Burp Suite:** This tool is fantastic. It works very well for testing SQL injections (often used alongside SQLmap)
- **Nmap:** The holy grail of port scanners (not that port scanning is all that useful nowadays anyway)
- **Metasploit Framework:** The holy grail of exploitation. You can use many of the built-in modules, or you can create your code and import it into the framework, allowing you easy access to them and perfect integration with other exploits. There's a paid version (Metasploit Pro), which I would also get. You can get it for free every 14 days by applying for a free trial. They only let non-free emails so you can't use @gmail, @hotmail, etc. to register. However, you can create a free trial account at http://www.fastmail.com, for example, for 30 days and that email will work to apply for a free trial of the Metasploit Pro Framework. Repeat the process every 14 days or whenever you want to extend your free trial license.
- **Wireshark:** The best tool for monitoring traffic. Learning more about it will give you a boost in your abilities to compromise systems and gather information.
- **Aircrack-ng Suite:** You can't ask for better when it comes to Wireless hacking.
- **Reaver:** Also a part of wireless hacking.
- **Social Engineering Toolkit (SET):** This tool is useful when you want to perform fast social engineering attacks such as phishing links. It doesn't give you as many options as the Metasploit Pro Social Engineering attack does in certain aspects, but it's a great tool.
- **Ettercap, Bettercap, and MITMf:** When performing man-in-the-middle (MITM) attacks Ettercap is one of the most known instruments. However, there are security measures in place like SSL and HSTS that prevent certain MITM attacks. That's where Bettercap and MITMf come into the picture. These tools incorporate many programs into it to try and bypass these security measures. To some extent, they are successful but crafting your tools here might be the best way to ensure success when it comes to MITM attacks.
- **THC-Hydra:** Old-school online password attack tool, allowing brute force and dictionary attacks against most protocols. Definitely a good one to have in your arsenal. You don't *always* need to create a brute forcing tool.
- **Hash-identifier:** Not big enough to deserve its bullet point but I thought it would be good to include anyway because it is used so much.
- **Hashcat and John the Ripper:** These are great tools to crack hashes.
- **Msfvenom, Veil-Evasion, and Shellter:** Most traditional instruments to create payloads, encrypt them and attempt evading antiviruses. They don't produce incredible results. If you want to create a FUD virus, you should do it yourself because these tools won't achieve the result you desire.

Covering your tracks

If compromising the system and stealing data/disrupting the system is important, clearing your tracks is just as important or even more so. Some of the ways to erase your tracks have already been mentioned but not in great detail.

Clearing Event logs

The first place to start is the compromised system's event logs. They will register all the activity such as your connections to the system (as well as your failed attempts). There are many ways to do so. If you are in a meterpreter session, you can issue the command "*clearev"* which will remove all the event logs. However, if you are not in a meterpreter session or you don't want to be obvious you can remove only the security logs on the event logs.

➢ Windows:

You can remove the security logs in Windows by spawning an interactive PowerShell and issuing the following command: "*wevtutil cl security".* If you desire to erase all the logs, you can issue the following command instead: *"wevtutil el | Foreach-Object {wevtutil cl "$_"}"*

It is also worth mentioning that clearing the PowerShell command history can be useful so that the system administrators don't know what you have done. To view the history, issue the command "*Get-History".* You can then delete specific IDs by sending the command "*Clear-History -ID* [id]" (example: "*Clear-History -ID 10").*

➢ Linux:

Linux log files are stored in the */var/log/* directory. The following command allows you to open a text editor and view these logs: "*gedit /var/log/messages".* You can then proceed to delete all of the logs or, if you don't want to be obvious, delete only the ones that have to do with your activity.

In Linux, it is also worth mentioning that clearing the command history is more important than in Windows machines because unlike PowerShell, Bash keeps a record of your commands even after the process is killed. To view the history, we can issue the following command: "*gedit /root/.bash_history"*, Where *root* is the name of the user you are impersonating. There are two options to deal with the bash history:

- **Prevent it from being stored:** You do this by limiting the size of the history to zero. The size of the bash history is determined by the environment variable "*HISTSIZE*". By default, this variable is set to 1000. You can change it using the following command: "*export HISTSIZE=0",* making the history size equal zero. This has to be done before you start typing in commands in the machine. To make these settings permanent you can add the export command to the "*/root/.bashrc"* file, where *root* is the user for whom you want to make the settings permanent.
- **Erase it completely:** You do this by completely shreading the history file and replacing it with zeroes: "*shred -zu /root/.bash_history".*

Clearing Router logs

Routers are Unix-based machines. You can access them the same way you would access a computer. However, most routers won't allow command-line access by default, which means you will either have to rely on the victim's router allowing it or exploit the router to spawn a shell connecting you to it. After that's done, you can browse to the location of the logs and delete them. Note that you don't need to remove the records from your primary victim's router, only the ones from the same zombie computer you wiped the event logs on. In case you aren't able to access the zombie's router, cleaning them from your router (the one your main machine is connected to) will have to suffice.

You don't have to access the router from a terminal to clear its logs. There are two other ways to do it:

- Connecting through the router's configuration page (usually 192.168.1.1 on home networks) and searching for the logs tab and clearing it. By default, you will only be able to access this page if you are connected through Ethernet cable. If you are connected by wireless, access to this page will be denied. The solution is to take remote control of one of the users in the network who has an Ethernet cable connected and perform the rest of the actions from him.
- Unplugging the router from power will also clear all logs.

Clearing other logs (IDS, etc.)

There can be multiple other security measures in place that monitor the network and store log files. It would be too many to go over here. You would have to correctly and thoroughly gather your information about the target and determine what security they have in place and go from there. Find out where that system usually stores their logs and then delete them. If the victim has a system that prints logs (very rare but I once pentested a company that had that system) it can become a huge problem for a beginner. You might only realize it when it's too late, or worse – not at all. In my case, I realized it before it was too late. I found out at the reconnaissance phase that the system printed out a full page worth of logs every time the logs filled up an entire page. What I did at the beginning of the pen test was crash the printer service so it wouldn't print and at the end deleted all logs and restarted the spooler service, effectively bypassing that measure.

You may also face a system that stores records and backs them up on another machine. In this case, you must hack into the other computer and delete the files there as well.

Erasing and overwriting free space

After cleaning up one of your zombie machines and, therefore, stopping people from continuing your real IP trace any further, you have to erase the computers free space. Those logs were deleted, but in fact, they weren't. Every data in your system is composed of 1s and 0s. Deleting data means you replace all 1s and 0s to 0s. Example: Let's say you have the following piece of data: "10010110111101111101". You delete it by transforming it into "00000000000000000000", therefore, deleted. However, data in your system can help me very long. Each binary digit is a bit, 8 bits is a byte. So one kilobyte would be 1000 bytes or 8000 bits, which would mean a combination of 1s and 0s with 8000 numbers. Some of these logs files could have hundreds of kilobytes, hundreds of thousands of numbers to replace to 0 if we were to delete them. There's also another disadvantage to deleting files this way, which is decreasing the life of your hard drive. The more you write, the more it decreases in a lifetime. To solve this, when you click delete, what happens is the section in memory

where the data for that file is gets marked for rewriting. Meaning if you download more data into the disk, that section of memory is allowed to be rewritten by it. When a section of memory is marked for deletion, you can still recover all the data that was there. Only after you rewrite the memory addresses, you can say that the file is unrecoverable.

This means that we also need to delete the free space of our zombie after deleting all our evidence, to make sure none of the logs or other files we might have used are recoverable. Let's find out how to do this in Windows and Linux.

➢ Windows

To erase the free space in Windows, issue the following command at a command prompt:

"*cipher /w:**driveletter**:**foldername***" or simply "*cipher /w:**driveletter***"

This command can take a long time, so what you could do is schedule erasing the free space as a task in the future at the time you desire. You can schedule tasks in the following way:

"*SchTasks /Create /SC DAILY /TN "Name of Task" /TR "path to the file you want to run" /ST 10:00"*

This command schedules a daily task that executes a particular file. In this case, we may want to put the "*cipher /w:**driveletter***" in a batch file hidden on the target computer. However, running a batch file would be suspicious because a command prompt would show up on the victim's machine. We can solve this by calling the batch file using a VBS script. We would code the VBS script as follows:

```
Set WshShell = CreateObject("WScript.Shell" )
strCurDir    = WshShell.CurrentDirectory
WshShell.Run chr(34) & strCurDir & "\free.bat" & Chr(34), 0
Set WshShell = Nothing
```

Now that I have created a file called *free.vbs* with the above commands and another file called *free.bat* with a single line: "*cipher /w:%SystemRoot%\system32\winevt*", I can now proceed to schedule the task by issuing the following command in the command prompt:

"SchTasks /Create /SC DAILY /TN "Erase Free Space" /TR " WScript.exe %TEMP%\free.vbs" /ST 17:53"

NOTE: I placed both the VBS and Batch files on %TEMP% location as an example.

Now, every day, my victim will start erasing all free space without knowing, protecting us in case he gets tracked down.

The log files for Windows 2000, Server2003 and Windows XP are stored in the following directory: *"%SystemRoot%\System32\Config"*

For Windows Vista, Server 2008 and above: *"%SystemRoot%\system32\winevt\logs"*

➢ Linux

Erasing free space in Linux machines isn't as easy because it depends on the file system in place. Usually, the command performed to clear the logs ("*shred -zu /root/.bash_history*") should replace everything with zeroes unless the file system in place doesn't overwrite data.

I was hacked!

You will eventually, at some point in your life, be hacked. Either as a Blackhat Hacker or a Whitehat Hacker. Maybe the company you work for was hacked. Whatever the scenario may be you most likely want to do two things:

- Find the hacker
- Secure your system

How to find the hacker

Tracking down a hacker can be a very tedious and non-rewarding task. Most of the times, if the hacker did his job well, not only will you have a hard time tracking him down, you will be chasing ghosts. Oftentimes, hackers don't delete logs or hide their presence. Rather, they use it. They change the records to match someone else, change their presence to incriminate someone else.

Even if, by a miracle, the attacker was found, it is still a long way to go before he can even get convicted. Who's to say it wasn't used as a zombie by another hacker to perform it? He can also report that it was stolen weeks before he starts his attack. If he gets caught, he has a get-out-of-jail card.

However, regardless of whether the hacker made any mistakes or not, how would the authorities begin to track him down?

Log files

The first thing is checking log entries for the attacked server machines. Many attackers either don't wipe out all records or don't erase the free space leaving the logs a couple clicks away from being recovered. Even the best hacker in the world might not have time to wipe the logs clean. In this case, they are recoverable and can trace back to either a VPN or a TOR network node or a zombie computer. Before we proceed, however, there's another possibility that is worth looking into, if we are dealing with an experienced hacker. He might have changed the logs to point in another direction. We have to verify if the records were modified in any way. There are many different methods we can apply to determine this such as checking file size, timestamps, checking inode ctime changes, checking the last modification time, amongst other.

Despite all your efforts, it is still possible for a hacker with root permissions to a system to bypass these checks and make it look like the file wasn't changed and the records are accurate. This will result in the authorities being chasing ghosts for the first weeks. When they realize that the VPN IP address in the logs had nothing to do with the attack they will start looking for any mistakes you might have made while trying to erase your tracks. Maybe you forgot to change the records in one folder in one of the machines; maybe you didn't edit all the entries regarding your IP address in one of the logs, and they will go over everything again until they find a discrepancy. When they do, and they often do because let's be serious, no hacker can erase ALL tracks correctly in a timely manner after exploiting a big target; you'll always make a mistake. When they do, they will investigate the IP address they received and know everything about it. In case it's a VPN they will ask for the records with a court order for that particular time. In case it's a TOR node, odds are they control it, as well as the entry node you used so by comparing the time differences they can pinpoint your zombie computers or your VPN. In case it's a zombie they will ask (demand) the owner's ISP for his records and details, seize this machine and analyze it thoroughly. They will be looking for logs, erased files (such as viruses), etc. If

they recover anything that will lead back to the other zombie, they repeat the process until the computer of origin (the hacker's) has been found.

Authorities will have many more resources than private individuals to be able to track hackers down. They can issue court orders for VPN or ISP records; they have access to large databases that can help identify the attacker and access to fast computer systems. Federal agencies also have the privilege of using state-of-the-art forensic tools developed just for them, which can usually get results faster than open-source tools used by everyday hackers. This is more than enough to add a huge advantage to the IT experts that work for the authorities and federal agencies when compared to private individuals or companies. This is why it is generally a better idea to let these agencies track down the attackers.

Software and behavior

When a hacker compromises a computer, they perform certain actions. Authorities have to carefully use forensic tools to extract everything they can on the computer to determine what the hacker did on that system. Just like an autopsy, by finding out what happened, maybe they can find out more details about the attacker. Maybe he added a task that connects to a specified IP address (as a backdoor), or he used valid credentials that belonged to him at some point, or he sent files to an email that was created using his real IP address (or a zombie's), etc.

Other

There are other many other ways for a hacker to get caught. Believe it or not, some hackers leave their name inside certain files or access their Facebook account from the computer they performed the hack, or tell a friend he is going to hack into that company. There are many stupid and non-tech related ways that hackers get caught. A good hacker always knows better than to do any of these things.

Forensic Tools

I would like to talk a little bit about the tools used to analyze a computer system carefully.

There are many different tools available, some open-source, some paid and others restricted to law enforcement agencies and restrict companies.

- EnCase (https://www.guidancesoftware.com/encase-forensic): Tool usually restricted to law enforcement use. Very powerful, customizable, flexible and includes report-building features for automated analysis.
- Forensic Toolkit or FTK (http://accessdata.com/solutions/digital-forensics/forensic-toolkit-ftk?/solutions/digital-forensics/ftk): The competition of EnCase. Law enforcement agencies also use this tool all around the globe.
- The Sleuth Kit (https://www.sleuthkit.org/sleuthkit/): It's open-source! It's more of a library of commands you can use to analyze disk images.
- Autopsy (https://www.sleuthkit.org/autopsy/): This is a free tool as well, and it adds a GUI to The Sleuth Kit and makes the forensic process a lot easier. This is a very powerful tool.
- dcfldd (http://dcfldd.sourceforge.net/): A small tool used to create a copy of a disk image. It is used before the forensic analysis starts. You don't want to run forensic tools directly on the customer's computers; you want to create a copy of it first and analyze that copy.

The Forensic method

Although we have discussed how logs are important and how many other fingerprints may have been left in the system, we haven't discussed how this analysis should be done. Here, I'm presenting the usual steps of the forensic method from the moment Law Enforcement agencies are contacted**.**

1) **Live analysis:** This happens by running the Forensic Tools on a live boot while the system is running. This is the first step in an incident report before confirming it. After incident confirmation, a dead analysis is performed.
2) **Dead analysis:** Running forensic tools on an image file of the entire file system of the computer. These are the steps taken to conduct a dead analysis:
    a. Create a bit-by-bit copy of the entire computer system as an image and hash it to ensure integrity, example: "*dcfldd if=/dev/sda of=/media/case020234.dd hash=md5 bs=256 noerror*".
    b. Start Autopsy or FTK or EnCase and import the image and verify the hash to ensure integrity.
    c. Start scanning, recovering deleted files, analyzing raw data from suspect files, etc. Here is where the expert will examine the log files as well as any suspicious files.

Random Questions

If you'd have to choose a VPN from a certain location, what country would you choose?

Well, to be honest, any country as long as it's not the USA. The United States of America is arguably the most vigilant and monitoring country in the world. If I had to choose a VPN, I would just stay away from that and choose an Italian or Malaysian one. I wouldn't recommend using VPNs because of the reasons I outlined in *page 5, VPN section*.

Beside all above (online anonymity solutions) services, are there any hardware solutions for attackers to prevent their identity leak?

Technically yes: for example the TOR/VPN router I had discussed before. However, you can't fake TCP connections, it's just impossible, so any method you use can, at least theoretically, be traced back to you unless you cover your tracks entirely after performing an attack.

Is it possible to trace back an attacker who uses a VPN with "no-logging policy"?

As I've discussed before, I have no way of verifying if they really don't keep any logs. However, if they don't, there are still ways to trace back that attacker, in the form of DNS leaks or any fingerprints left in the victim's system.

Is it possible to trace back an attacker who uses TOR network?

Yes. The TOR network has been cracked. As I've discussed before on *page 6 section TOR,* you can locate an attacker if you control both the exit and entry nodes used. However, the probability of you getting caught using TOR – given you didn't make any stupid mistakes – is pretty low.

What are the most mistakes attackers do, that lead them to get caught?

- Forgetting to delete all logs either in the main victim's machine or at least one of the zombies';
- Spending too much time connected to the victim: This gives law enforcement of computer security personnel the time they need to track you down;
- Getting lazy or overconfident: This one is a little bit vague, but I should include it. Many excellent hackers get caught simply because they get too overconfident after performing hundreds of hacks and never getting caught and start making very simple mistakes.
- Getting in over their head: Sometimes beginner or even moderately experienced hackers think they can get away with breaking into highly secure environments and aren't prepared to deal with all the security measures in place.

If you wanted to hack into someone's system, from a particular country, with specific content on his system, how would you approach this without using a wide spreading virus that could check the IP and the content from the hard drive?

This one is tricky, but a little bit of creativity goes a long way. I would probably try to find connections between all people who have the specific content I'm searching and belong to the country I'm

targeting. For example: Let's say I'm targeting Portuguese people that have Photoshop installed. I would create Portuguese websites with Photoshop tutorials in the form of PDF files containing viruses. That way I don't check their IP or the contents of their hard drive, but I'm positive most of my victims will be Portuguese and have Photoshop installed.

Remote Exploitation Demonstration

As promised, I am sharing a video of a remote exploitation with some explanations. The virus and scripts created for this demonstration were custom-made and I will not be disclosing the source-codes because they are in use in active pen-tests.

Do not mind my "thin" voice and I apologize for the fact that I am not good at creating video content, I am more of an improviser, so any scripted material will make me struggle. With this said I hope you enjoy the video and find it informing.

Errors in the video

There were some unfortunate mistakes here and there, which is why the video is very long. However, the end result was still achieved. The mistake that made the video last longer was in the .reg scripts I created to deactivate Windows Defender. It could be solved by simply recreating those scripts without the mistakes. Another way to do it is to use "reg delete" and "reg add" commands in a shell, making the process less prone to error.

https://youtu.be/DDbRluJzs2A

Final thoughts

Computer security is a very vast field and it requires a lot of dedication and effort to master any subject. Many people who are interested in computer security ask me what the best field is or what field has better results in the real world. The answer to that question is all. It's often not about the field but about the person. This said, there is a particular field that is becoming very big and growing roots into every other field in computer security: Social Engineering.

Computer security experts are creating more secure software every day and whenever a critical vulnerability is found it is usually patched within hours or days. However, who's patching security flaws in people?

As Kevin Mitnick said, you can patch a vulnerability on a system but there's no patch for stupidity, or rather, gullibility. People are the most targeted system in the world and will continue to be so for very long. People who work on help desks, for example, only want to aid others, they are not looking for malicious intent on every people they meet in their work, and you can exploit that very easily and gain access to a company quickly.

All in all, due to the fact that computer security is a huge field, whether you want to become a Blackhat hacker or a Whitehat you will always do your best in a team. Teams composed of people specialized in many different areas adds versatility and knowledge to the table, allowing you to really accomplish the original meaning of the word *hacking* (finding creative solutions to difficult problems).