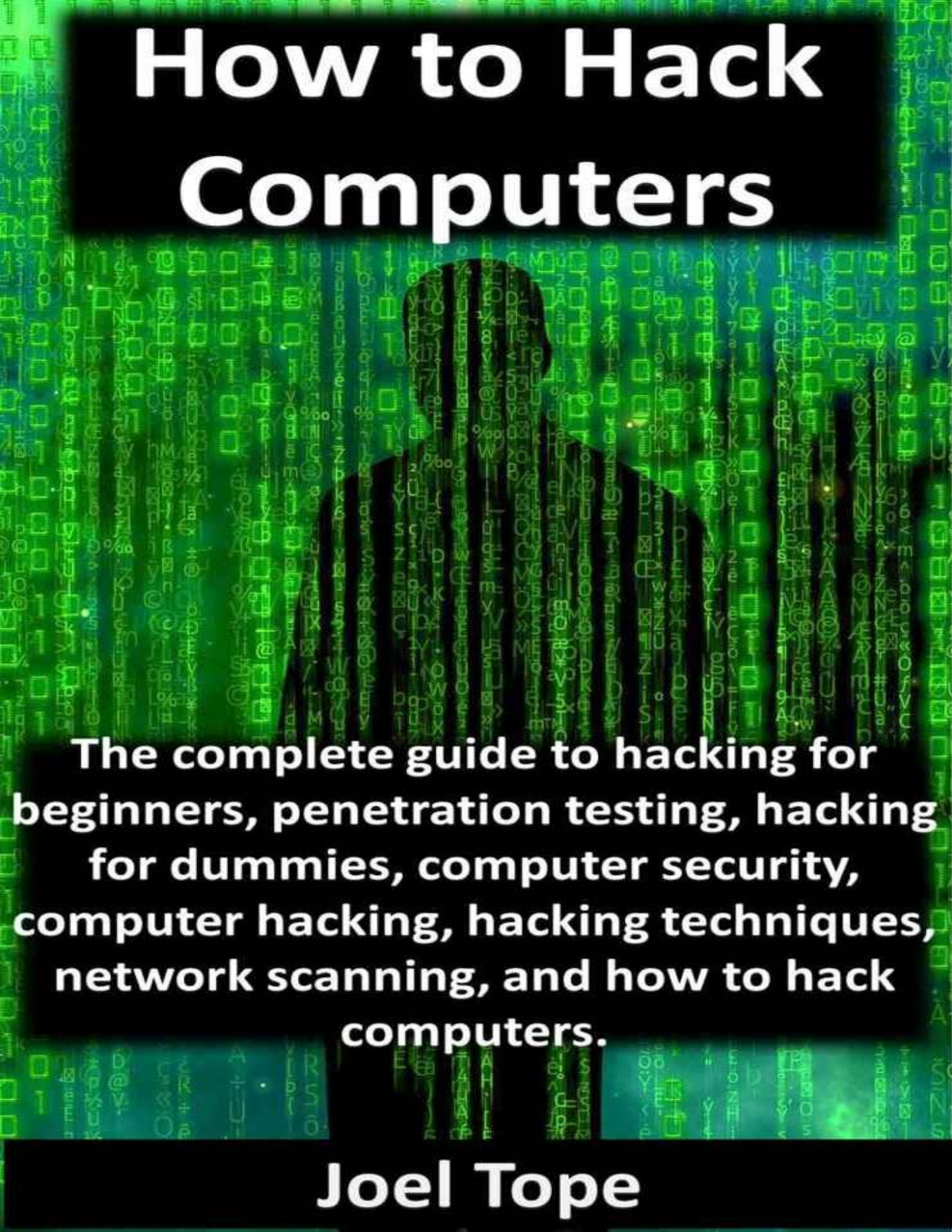
How to Hack Computers



The complete guide to hacking for beginners, penetration testing, hacking for dummies, computer security, computer hacking, hacking techniques, network scanning, and how to hack computers.

Joel Tope

Contents

How to Hack Computers

A Guide to Hacking Computers for Beginners

Joel Tope

Chapter 1 – Introduction

The general public usually has two competing viewpoints of hackers. Some people revere them as brilliantly minded individuals while others look down on them as petty criminals. While both perceptions could be true for many expert hackers, the public's perception has been twisted and contorted by what they see on television dramas and in the movies. Because your average user doesn't understand how a computer or the Internet works from a technical perspective, they can't hope to begin to understand what hackers actually do.

In fact, the term 'hacker' usually carries a negative connotation to it. Ask any non-technical person what a hacker is, and they'll give you a response such as, "They're the bad guys that steal people's credit cards, listen to my phone calls, and work with terrorist organizations." For some reason – likely accredited to entertainment media – hackers get a bad rap and most people would instantly assume that their behaviors are illegal. These stigmas couldn't be further from the truth, because the reality is that there are many types of hackers. Some of them are good, some of them are bad, and some lie somewhere in between. There is no single motivation that drives every hacker and no blanket statement that you can use to accurately describe every hacker in the world. Also consider that hacking isn't an inherently evil practice and you can do it legally. Some people even like to do it for a hobby. More practically, however, some people get paid big bucks as consultants to try to hack into a corporate network in an effort to find security holes. Be forewarned, though. If you start abusing your knowledge it is a slippery slope to the dark side, and nothing good ever happens once you're there.

If your curiosity has gotten the better of you, if you just want to be able to understand what's going on in the movies and the news, or you have a goal of becoming a competent hacker, I want to personally introduce you to hacking and guide you to achieving your goals. The problem most people have when they

want to start hacking is that they find material that isn't written for novitiates. Once you get the basics under your belt and you can actually apply the knowledge you will learn in this book, you'll find that you are much more educated than your peers and that technology is actually pretty exciting. As the tools hackers use have changed over the last couple decades, people that take an interest and develop a passion for hacking have changed as well. Though technology is only getting more complex with each passing year, the tools hackers utilize are becoming more sophisticated – making the learning curve much less steep for newbies.

In this guide, I am going to teach you a lot of valuable information about hacking such as:

-What hacking is and what hacking isn't.

-Hacking terminology and hacker culture.

-Types of attacks and the most famous hacks of all time.

-Ethical considerations and fair warnings about becoming a hacker.

-Fundamental concepts that will serve as a foundation to build hacking skills.

-How to install Linux operating systems using VMWare to setup hacking tools.

-Step-by-step guides for ping sweeps and port scanning.

-How to map network topologies and perform reconnaissance techniques.

-How to use advanced software to find security holes.

This is designed to be an all-inclusive guide that will not only give you an understanding of the basic technical concepts you will need to become a hacker,

but also introduce you to some fascinating software and show you step-by-step how to use it. I’m sure most of you want to get started hacking right away, but I urge you to spend time learning the basics before moving on to some of the more challenging attacks discussed in this book.

What it Takes to Become a Good Hacker

One of the reasons some hackers become so successful is because they have a passion for what they are doing. Their personality drives them to tackle extremely difficult challenges, which is why some hackers break systems just to see if they can. If you are going to want to become a prolific hacker, it takes the same two things as any other skill you want to build: time and practice. If you can't figure something out in the first two minutes, don't give up. Some of the pros will spend weeks or even *months* planning and executing their attacks. And once you get the basics under your belt, you're going to be able to implement these techniques in a matter of minutes. Arguably, I would say the hardest part for a newbie is getting their environment setup. Past that, things start to get easier and you can really start to sink your teeth into how the technology works. Before we get to the juicy details, we should begin with an overview of hacking so you understand some rudimentary concepts and perceptions about hacking.

Chapter 2 - An Overview of Hacking

To your average computer user who doesn't understand much about Internet and network security, hackers are shrouded in a cloud of mystery. Most people don't understand what they do or how they do it. And the movies don't help to demystify them, either. Countless action movies portray a character that takes the role of a hacker that can break into top secret computer systems to save the world. When the camera pans over their computer screens, you see them typing strange letters and numbers into a command prompt that, for all you know, is a foreign language. Humorously enough, the hackers in the movies frequently use a tool called NMAP, which I will show you how to use later in this book. If you've seen *The Matrix Reloaded*, *Dredd*, *Fantastic Four*, *Bourne Ultimatum*, *Die Hard 4*, or *The Girl With The Dragon Tattoo* (among countless others), you have already seen actors using NMAP to facilitate their hacking endeavors in the movies.

But what exactly is hacking? Hacking means a lot of different things to a lot of different people. It is an umbrella term used to describe hundreds, if not thousands, of various techniques that can be utilized to use computers and information systems in unintended ways. At its core, hacking means using a computer to gain unauthorized access to another computer system or data that is protected or restricted. This is the most conventional meaning of the word hacking. Once a hacker has gained access to an unauthorized system, he or she then has the ability to steal information, change configurations, alter information, delete information, and install further malicious code to capture even greater control over the target system. The list goes on and the sky is the limit regarding what an experienced hacker can do once they find a way into a computer system.

However, there is a lot more to hacking than clicking a button to attack a computer. You will need to use tools and scanners to map the local network

topology and use reconnaissance techniques to gather information and look for vulnerabilities. The good news for newbies is that these tools are highly automated today. In the past, hacking software hadn't been created that aggregated vast amounts of code and tools into simple and easy to use commands. As such, hackers in the past needed highly intimate understandings of the technologies they were trying to break and it was difficult to do so. Having an extremely deep understanding of technology today will certainly help you become a better hacker, but my point is that these tools are becoming increasingly easy to use. In fact, there are young kids and teenagers that are too curious for their own good and take advantage of highly sophisticated tools to break into systems they have no business accessing. Understand that these tools simplify the hacking process considerably. If a teenager can hack into a system using simple tools, guess what? You can too!

But what does it take to excel as a hacker? Well, most hackers have several things in common. First of all, they are experienced software developers and can craft malicious programs and viruses that further their cause. Furthermore, most hackers are competent Linux users. Linux operating systems are extremely secure and provide virtually limitless access to the latest penetration and security tools – for free! In addition, some Linux operating systems such as Kali Linux were designed for the sole purpose of hacking and network penetration. Linux can be scary for newbies, but I will show you how to run Linux and use some special tools later in this book in a simplified and easy to understand manner. Lastly, hackers almost always have a working knowledge of networking topics such as IP addresses, ports, and the dirty details of how different networking protocols operate. Some tools even exploit vulnerabilities in these network protocols, and the knowledge of these exploits combined with the ability to craft computer programs is what makes some hackers truly formidable.

Some of these techniques are outside the scope of this book since this guide was created for beginners, but if you really want to excel as a hacker you would do well to study and practice these concepts. Though we won't touch on software development in this guide, I will certainly show you step-by-step how to install and use some various hacking tools that the pros take advantage of and teach you

the basics of networking addresses and protocols.

Chapter 3 – Attack Types and Famous Viruses

Most of you have probably heard of viruses, worms, malware, key loggers, rootkits, and Trojans before, but what the heck are these things and how to hackers utilize them to steal people's data and disrupt their computer systems? Each of these tools are a little bit different from each other, but they all have one similar goal: to enter a target's system to provide the attacker with information he or she doesn't already have access to. No, I'm not going to show you how to craft nefarious computer software, but you should have a well-rounded understanding of these topics if you have any hope of calling yourself a hacker.

First and foremost, you need to understand the concept of computer viruses because they are one of the most popular terms thrown around in discussions about cyber security and hacking. A computer virus is a piece of malicious code or software program that is able to infect a target system and then make copies of itself on other local computers. They are aptly named because they reproduce much like a virus in real life, and they facilitate their operations by attaching themselves to computer programs. Typically they either render a computing system completely useless or they seek to destroy data. Again, you'll hear about computer viruses in the movies a lot, so we'll take a look at some of the most famous computer viruses of all time after defining the other terminology.

A worm is very similar to a virus, and it's true that the line between a virus and worm gets muddied and blurred. The largest difference is that worms are not attached to a computer program. They exist independently on the host system, and they often take advantage of network resources to spread to other hosts on the network they have compromised. Sometimes worms are also classified as malware, because there are only minute differences in the terminology. Colloquially, these terms are interchangeable but their meanings vary slightly in academic settings.

Perhaps you have already experienced the negative consequences of malware. One of the most popular ways that malware is distributed is through the medium of online downloads, whereby a downloadable file has been corrupted with malware that the user then downloads and installs. You'll see this frequently with most files hosted with P2P (Peer-to-Peer) file sharing programs such as Bit Torrent. Malware gets its name by combing two other terms: MALicious softWARE. It can also be used as an umbrella term used to describe many different types of attacks, and it could mean any software that is used by an attacker to create access to a target's data, block them from their data, or change information on their computer.

Furthermore, a key logger is yet another type of malicious program, and as you might have guessed its sole purpose is to log the keystrokes of the user who has been infected. This is absolutely disastrous for the target user, because an attacker will be able to record and view *every single key* that the target types on their host system. This includes usernames and passwords, Google searches, private instant messaging conversations, and even payment card data. If an attacker has successfully installed a key logger, the target is at the mercy of the attacker. There's no telling what the attacker could do next – they could hack into the target system by using the information they gathered such as usernames and passwords, steal money using their payment card data, or use their host system to carry out attacks on other hosts on the same network.

Next, you should also be familiar with the idea of a rootkit. Rootkits are extremely dangerous because they serve to edit background processes in an effort to hide the malicious activities of an attacker. This will help viruses, key loggers, and other malicious code exist for extended periods of time without detection on the target system. They can even serve to hide software that would have been otherwise detected and quarantined by security software.

Last but not least is the infamous Trojan horse, sometimes called a Trojan virus or a backdoor virus. They are extremely problematic because they can be slipped into innocent-looking applications and they are very hard to detect without the right security software. There could even be a Trojan horse lurking in the depths of your personal computer right now, and they are frequently used to gain complete control of a target system.

Now that you have a basic understanding of the different types of malicious code hackers employ to do their bidding, you should know about some of the largest and most famous computer viruses of all time. Some of them are actually other types of malicious code such as Trojan horses, but people still refer to them as viruses. Any expert hacker will have heard of these famous attacks before, so you should know them as well.

Also, if you get the inkling to try your hand at using one of these methods on your own by hunting around on the Internet for freely distributable code that will allow you to attack a target system, just know that you're setting yourself up for a disaster. Humorously enough, some hacking newbies try to find rootkits and key loggers to attack hosts. But here's the catch – some hackers actually facilitate their attack by taking advantage of people who want access to these types of programs.

And the end result isn't pretty. In the end, the newbie hacker might actually install an expert hacker's virus and unknowingly infect their own operating system! And don't forget that there are ethical and legal implications as well. Many, if not all, of the people responsible for these famous attacks were severely punished. So don't try to research and implement these types of viruses at home!

1. Code Red

I know what you may be thinking, and no, this has nothing to do the movies. When people think of hacking in the movies, they think of top secret military bases getting hacked by a teenager and raising their alert level to ‘code red.’ Believe it or not, it is rumored that the two engineers who discovered and named this attack were merely drinking the disgusting cherry-flavored soda when they first identified the worm back in 2001. This worm was pretty darn nasty, and its targets were servers that were running the Microsoft IIS software for web servers.

This attack relied heavily on an exploit found in the code that left servers vulnerable to a buffer overflow issue in an older version of code. However, it was a huge problem and very difficult to detect because it had the ability to run solely in memory (RAM, or short term storage as opposed to long term storage such as a hard disk drive). And things got out of hand pretty quickly, too. After it had compromised a system, it would then try to make hundreds of copies to infect other web servers. Not only that, but it gobbled up a ton of local server resources that all but crippled some of the target systems.

2. Sasser

Sasser is another worm designed to target Windows (noticing a pattern here?). It first found its way into the spotlight back in 2004 and was created by a legendary and infamous hacker named Sven Jaschan who was also responsible for another famous worm named Netsky. One reason this worm made Internet security headlines was that it had affected more than a *million* targets! Yet again, this worm took advantage of a buffer overflow vulnerability that caused target systems to crash.

It also made it nearly impossible to reboot your computer without removing the power cable and it caused many computers to crash completely. To be fair, most people saw this worm as a nuisance as opposed to a serious threat. But it cannot be denied that it caused massive and widespread disruption. It even infected critical infrastructure devices that caused networks to perform very poorly. Like other types of worms, it used its target computers to propagate and multiply itself to other computers.

But one of the biggest problems with this worm is that users didn't upgrade their operating systems after a patch had been created. Both public and private sector organizations were affected like news stations, transportation systems, healthcare organizations, and even some airline companies. But what was the end result? The damages were collectively chalked up to be approximately $18 *billion* dollars! What happened to the infamous Jaschan, you ask? Fortunately for him, he was still young so he received a slap on the wrist considering how much damage he did. He ended up with a suspended sentence lasting 21 months.

3. Zeus

The Zeus virus was really a Trojan horse created to infect (can you guess which operating system?) Windows machines in an effort to force them to carry out varying procedures that were deemed to be criminal activity. Most typically, it would be used to carry out key logging activities and man-in-the-middle attacks that would allow an attacker to first sift through web browsing information before sending it to the intended web server. It most frequently infected hosts by utilizing innocent-looking applications as a transport medium into the intended targets, but the attack also employed phishing techniques.

After it had been discovered in 2009, it had ruined thousands of individual file download and FTP accounts from the largest banks and corporations. Those involved include Amazon, Bank of America, Oracle, and even Cisco. The attack also allowed the hackers to steal usernames and passwords to social media sites, email accounts, and banking information.

4. The I Love You Attack

The ‘I Love You’ attack is so impressive and revered in hacker communities because it created a whopping $10 billion dollars in estimated damages. What’s more impressive is that researchers believe that 10% of *every* computer connected to the Internet at the time was infected with this virus. Infecting 10% of the Internet with a computer virus is staggering to say the least. Things started becoming so terrible that some of the larger organizations as well as governmental agencies around the world started shutting down their mailing systems in an effort to avoid becoming infected.

5. Melissa

This naughty virus was supposedly named after an exotic dancer the creator, David L. Smith, had once known. Supposedly, the very root of the virus was an infected text document that was uploaded to the alt.sex Usenet group with the appearance of being a collection of usernames and passwords for subscription and membership-only pornographic websites. But once a user downloaded this Word document, all hell would break loose and the virus would activate.

To start, the virus would look at the first 50 addresses in the infected host's email address book and start sending those addresses emails. In turn, this would severely disrupt email services of large enterprises and governmental bodies. Furthermore, the virus would even corrupt documents by adding references to the television show *The Simpsons*. However, the original Word document was eventually traced back to Smith and he was arrested within a week of the virus's propagation. Although Smith only ended up serving 20 months of prison time and a $5,000 fine (he originally had a 10 year sentence) because he turned snitch on other hackers and helped the FBI make more arrests. To top it all off, it was estimated that the damages from his virus totaled approximately $80 million dollars.

6. The Conficker Worm

The Conficker worm first appeared in 2008 and it comes from an unknown origin. This worm was especially troublesome because it created a botnet (a group of infected computers networked together) of more than 9 million different hosts that harmed governmental agencies, large enterprises, and simple individual users alike. This worm makes the top 10 list because it caused damages estimated at a staggering 9 *billion* dollars. It was able to infect Windows machines due to an unpatched vulnerability dealing with background network services.

After a host had been infected with the worm, the worm would wreak havoc by preventing access to Windows updates and antivirus updates, and it could even lock user accounts to prevent people from logging in and cleaning up the worm. If that weren't bad enough, the worm would then continue its attack by installing malicious code that would make the target computer part of the botnet and scam users into sending the attacker money by holding their computer ransom. Microsoft and third party antivirus software providers eventually released updates to combat and patch this worm, but it did massive amounts of damage before a solution could be reached.

7. MyDoom

MyDoom was first seen back in 2004, and it was one of the fastest email worms to infect masses of computers since the I Love You attack. The creator of this attack is still unknown, but it is rumored that the creator was paid big money to carry out this attack due to the message included in the virus that read, “Andy, I’m just doing my job. Nothing personal, sorry.”

This worm was incredibly sly because it took on the appearance of an email error. After a user had clicked on the “error” to view the problem the worm would send copies of itself to people found in the email address book of the infected system. Furthermore, it would copy itself into peer-to-peer directories on the infected hosts to spread throughout the network. It is also believed that the worm is still lurking on the Internet to this day, and it caused approximately $38 billion dollars’ worth of damages.

8. Stuxnet

This attack has a somewhat political background as it is thought to have been created by the Israeli Defense Force in conjunction with the American government. While some of the past viruses were created out of malice, contempt, or the curiosity to see just how much damage a prolific hacker could create, this virus was created for the purpose of cyberwarfare. The goal was to stymy the initiatives of the Iranians to create nuclear weapons, and almost two thirds of hosts infected by this virus were located in Iran.

In fact, it is estimated that the virus was successful in damaging 20% of the nuclear centrifuges in Iran. More specifically, this virus targeted PLC (Programming Logic Controllers) components which are central to automating large machinery and industrial strength equipment. It actually targeted devices manufactured by Siemens, but if it infected a host that didn't have access to Siemens products it would lurk on the host system in a dormant state. Essentially, it would infect the PLC controllers and cause the machinery to operate far too fast – which would ultimately break the machinery.

9. Crypto Locker

This virus is another example of a Trojan horse that infected Windows machines, and the goal was to ransom target computers in exchange for money. This Trojan was very cunning because it had several different ways to spread to other computers. However, it was incredibly troublesome because after it had infected a host, it would then proceed to encrypt the hard drive with an RSA key that the owner of the computer never had access to. If you wanted your files to be unencrypted, you would have to pay money with prepaid methods or bitcoins to the initiators of the attack.

Many people were successful in removing the Trojan from their computers, but they still had one gargantuan problem: the files on their hard drive were still inaccessible because they could not be decrypted without the key. Fortunately the leader of the attack, Evgeniy Bogachev, was caught and the keys used to encrypt the targets' hard drives were released to the public. Apparently, the attack was successful in garnering $3 million from the ransoms, and it infected about half a million targets.

10. Flashback

I always love it when Apple evangelists claim to PC users that their computers are superior to Windows machines because their code is infallible and there is no way to get a virus on a Mac. While it's true that Windows machines are more susceptible to viruses, Macs aren't perfect either. Such was the case with the Flashback Trojan that was first observed in 2011. This Trojan used infected websites to inject faulty JavaScript code into the host browser, and it made infected Mac hosts part of a botnet. Believe it or not, this Trojan had infected over 600,000 Mac computers and a few of those were even contained at Apple HQ. Also, though numerous warnings and solutions have been created for this Trojan, many believe it is still lurking in the depths of the Internet and that thousands of Macs are still affected.

In Summary

Viruses, malware, and Trojan horses are just one facet of hacking, though. The truth is that these viruses were created by experts who had a deeper knowledge of computing systems than many of the security experts. All of the people who carried out these attacks were expert software developers and coders. If you think you want to become as infamous as these types of hackers, you're going to need to become an expert software developer. There's no way around it. However, I would hope that this section only opened your eyes to the potential some of these attacks have to cause widespread devastation and costly damages.

Again, please understand that the purpose of this guide isn't to teach you how to create a program that will harm other people's computers, rack up massive multimillion dollar damages, and leave you with heavy consequences such as prison time and ungodly fines. However, as a white hat hacker, you need to be aware that these types of attacks exist so you have a basic hacking vocabulary and some foundation knowledge.

I will, however, show you how to crack various passwords, map network topologies, exploit vulnerabilities, and scan targets for security flaws. In these types of examples, we will be focused on hacking into a single target host or network instead of trying to release a plague upon the global Internet. All of that in good time, however, because first you need to understand the different types of hackers that lurk on the Internet, ethical considerations regarding your use of the knowledge in this book, and the consequences of your actions should you misuse this information and get caught red-handed.

Chapter 4 – Ethical Considerations and Warnings

A book about hacking would be irresponsibly incomplete without a chapter giving you a fair warning on the consequences of misusing these techniques as well as the ethical considerations of hacking. To begin this discussion, you need to be familiar with two different terminologies that describe different types of hackers: black hat and white hat. I like the imagery these terms bring to mind because they always seem to remind me of *Spy vs Spy*.

Black hat hackers are what most people typically think of when they hear the word "hacker." A black hat hacker is the type of nefarious Internet user who exploits weaknesses in computing systems for personal gain or in order to disrupt an organization's information systems to cause them harm. He's the guy wearing a high collared shirt, sunglasses, and a fedora behind an array of 20 or so computer monitors or the nerd in the movies who can break into a top secret system illegally.

There really isn't any good that can come out of adopting a black hat approach to hacking, either. When you hear in the media that a financial institution just lost thousands of usernames and passwords or that a social media database was compromised that caused vast amounts of people to lose sensitive personal information, the attack was carried out by a black hat hacker. Recently, there was even a module of code contained in a WordPress plugin that was susceptible to an XSS vulnerability (a type of security flaw in websites with caching plugins) that was being exploited worldwide by the extremist group ISIS. If you are reading this book because you have dreams of causing mass disruption and chaos, I would highly advise you to reconsider. However, understand that security and penetration tools aren't inherently good or evil. One could argue that they are much like firearms in the sense that the weapon is an inanimate object and it is only as good or evil as the person wielding it.

White hat hackers, on the other hand, are the complete opposite. They're the good guys who do everything in their power to find potential security flaws and correct the errors so the black hat hackers can't break a system. As you read this book, you need to consider all of the tools and techniques I show you from the perspective of a white hat hacker and use them responsibly. If you pursue white hat hacking professionally, you can add tremendous value to the organization you work for and make big money doing so. Some white hat hackers that have the CEH (Certified Ethical Hacker) certification make salaries well into the six figure range. Internet security is only becoming more important with each passing year, and a talented white hat hacker can use penetration testing tools and footprinting methods to identify disastrous security flaws on the organization's network and information infrastructure and patch them before they become a problem that would cost the organization obscene amounts of money.

Furthermore, you need to be aware of the consequences of misusing the knowledge you learn in this book. Though you likely won't get caught snooping around a network attached to an unsecured SOHO (Small Office/Home Office) wireless network in your neighborhood or at your favorite local coffee shop, you need to respect other people's rights to privacy. Think about it – how would you feel if you were sitting down for a cup of coffee while reading a book only to find out later that someone had attacked your Kindle over the coffee shop's network and stole your data? You would feel enraged, irritated, and violated. So remember the golden rule as you grow into a white hat hacker.

Also consider that using penetration tools on networks where you don't have any authority to do so could lead to some extremely negative consequences. Let's face it, you don't have the right to steal other people's personal information – it's illegal. Not only could you provoke civil lawsuits, but you could even face jail or prison time depending on the nature of your offense. If you choose to do it on your employer's network and you get caught, the best case scenario is that you would have some extremely uncomfortable questions to answer and the

worst case scenario is that you would become fired. It's just not worth it, so keep that in mind moving forward.

Instead of testing out these techniques on public or corporate networks, my advice would be to try these in your very own home. Even a small home network will provide a digital playground for you to test out your new security skills. All you would need to run through some of these demos would be a personal computer, a wireless router, and preferably a few other devices that you can attach to your network. In the footprinting section I will show you how to run ping sweeps and other utilities to perform reconnaissance and information gathering methods, so having several other devices will give you more "toys" to play with on your local area network (LAN).

By now I hope you understand that the word "hacker" is rather ambiguous. Years ago, it rightfully meant a black hat hacker. Today however, it could refer to any number of different types of people who are extremely knowledgeable about technology, and the term "hacker" doesn't necessarily mean someone who is trying to steal intellectual property or break into a restricted network. Calling someone a hacker is the layman's approach to describing a digital thief, but security professionals will often draw the line between the white hats and the black hats.

With all of the dire warnings out of the way, we can now proceed to the juicer and more pragmatic sections of the book you have all been waiting for and we can begin to learn how you personally can get your feet wet with hacking. To begin, understand that this book is written with the assumption that you have little to no understanding of rudimentary networking and security concepts. Because this book is written for beginners as opposed to seasoned Internet security professionals and expert hackers, you need to first have a basic understanding of network terminology, addressing concepts, and other fundamentals that you will be able to use as a foundation to build your hacking skills upon. So, let's get started networking fundamentals!

Chapter 5 – Networking Fundamentals

Understanding the OSI Model and Networking Terminology

The OSI Model (Open Systems Interconnection) is one of the best places to begin if you are lacking a working knowledge of networking concepts. Just about every one of the demos we will run through together is heavily based on the OSI model and network security professionals often throw around terminology and jargon related to different components of this model. Also, it will benefit you personally if you understand what level of the OSI model various attacks target and this knowledge is fundamental to understanding IP addresses and ports, which we will cover later in this chapter.

To begin, understand that the OSI model consists of seven different layers as follows:

7. Application – A computer application that creates data such as an email or instant messaging program

6. Presentation – The method of encoding data, such as ASCII text

5. Session – TCP ports (FTP, POP, HTTP, HTTPS, etc.)

4. Transport – TCP or UDP connections (among others)

3. Network – IP addresses and packets

2. Data-Link – MAC addresses and frames

1. Physical – ones and zeros (bits) transmitted across a cable

(*Note: If you don't understand some of the terminology described above, take a deep breath and relax. We'll get to that later.*)

I realize that this list may look odd because it starts with the number 7, but the first layer of the model is always represented on the bottom since each additional layer is dependent on its subordinate layer to encapsulate and transmit data. You can remember the first letter of each layer with the pneumonic ‘Please Do Not Throw Sausage Pizza Away’. We won’t go into great detail about the finer points of this model as we will really mainly be concerned with layers 2, 3, 4, and 5 from a hacking perspective, but you need a high level understanding of the OSI model regardless.

Each layer has its own specific function to facilitate data transmissions between two remote systems. As data (like the text in an instant messaging application) is generated on one device, it starts at the top of the OSI model in the application layer and gets pushed down through each subordinate layer until it becomes 0’s and 1’s on a cable at the physical layer. Each layer encapsulates data for transmission before sending it on to the next layer for further encapsulation. The process works much like Russian nesting dolls. Once the data has reached the physical layer, it gets transmitted as binary bits over a cable medium. Then, the receiving host unpacks the encapsulated data from each layer using the reverse process.

This model is fundamental to understanding data transmission, but how will this help you build a skillset for hacking? First of all, it is essential to understand this model if you hope to learn about different network protocols and TCP/IP ports. Also, terminology is often thrown around regarding a device’s or protocol’s function and what layer of the OSI model it belongs to. For example, MAC addresses are a layer 2 address while IP addresses are a layer 3 address. And ports – which I am sure you have heard of before – belong to layer 5. We will dig into all of these concepts shortly, but first you need to know about IP addresses so you can identify various hosts when you are hacking!

IP Addressing Essentials

Of the fundamental concepts we are discussing in this book, IP addressing is by far the most important. But what is an IP address? Well, and IP address is a number that serves as a unique identifier that helps computers differentiate between hosts connected to their network. The most common analogy to describe this concept is that of the post system. If you wanted to mail a letter to someone (send them data), you would first need to know their home's address (IP address) before your message could be delivered.

Whether you know it or not, you have undoubtedly seen IP addresses already. They consist of four numbers ranging from 0-255 that are separated by periods as in the following example: - 192.168.1.1

Also understand that an IP address is 32 bits long. We won't dig into binary math because it won't do much for our network penetration examples later in this book, but know that each number separated by a period in the address is called an *octet*. It is called this because each of the four numbers are 8 bits (1 byte) in length. However, this IP address lacks something called a **subnet mask**, so we don't know what network it belongs to.

Subnet Masks

Each IP address is composed of two portions: the network portion of the address and the host portion. A subnet mask determines how much of the IP address defines a network and how much of the address identifies a host on that network subnet. For the remainder of this book, just note I will use the terms **LAN** (Local Area Network) and **subnet** interchangeably. Consider the following four examples of subnet masks:

1. 255.0.0.0 (/8) – 8 bits (the first octet) define the network portion of the address.

2. 255. 255.0.0 (/16) – 16 bits (the first two octets) define the network portion of the address.

3. 255. 255. 255.0 (/24) – 24 bits (the first three octets) define the network portion of the address.

4. 255. 255. 255. 255 (/32) – This subnet mask indicates a host address. It does not indicate a network subnet.

Note that subnet masks can be written using two different notations. Consider the first example. 255.0.0.0 is just another way of writing "/8" because they both indicate that the first octet in the IP address (the first byte or the first 8 bits) describes the network portion of the address.

Did you notice how these four subnet masks are in multiples of 8? That was intentional because it makes our example much easier. The truth is that there are many more complex subnet masks such as /17, *21, or* 30 that lie outside the scope of this book because they require binary math. However, on private home networks such as the environment where you will be testing our demos, a *24 subnet mask is by far the most common. I'd even bet big money that your home network device uses a* 24 subnet mask. That is, unless you changed it – in which case you would already know about IP subnets!

So, now it's time to put two and two together. We are going to consider an IP address and a subnet mask together, determine the host and network portion of the address, and then determine the complete range of usable IP addresses for that subnet. Consider the following:

- IP Address: 192.168.1.1
- Subnet Mask: 255.255.255.0

All right, so let's chop up the IP address and define the network portion of the address. Can you work it out? When the subnet mask is applied to the IP address, we see that the first 3 octets determine the network subnet. So, 192.168.1.0 *24 is the network on which the host with the IP address 192.168.1.1 resides. That means that the last octet determines the host portion of the address. On the 192.168.1.0* 24 network subnet, this host has the address of "1." Furthermore, we can conclude that because each octet can range from 0 – 255 that other hosts on the 192.168.1.0/24 subnet can use addresses from 2-254 (you never use the 0 or 255th address). Usable addresses on this subnet include 192.168.1.2 – 192.168.1.254. Understand that if the 192.168.1.1 host was sending data to the host using the 192.168.1.2 address, they are communicating over their LAN since they belong to the same network.

Two Special Network Addresses

So why don't we use the 0 or the 255th addresses on a subnet as host addresses? Because these two addresses are special. The first one is called the **network address**. This address can't be assigned to a host because it defines an entire network. In our example above, this address was 192.168.1.0. Also, note that the last address on a network subnet is the **broadcast address**. This address is used to send information to every host residing on that network at the same time, so this address can't be used for a single host address either. In our previous example, the broadcast address is 192.168.1.255.

MAC Addresses

MAC (Media Access Control) addresses are layer 2 addresses, and they are globally unique. Each MAC address is contained on the network card of your computer, and it is composed of twelve hexadecimal digits (0-9, A, B, C, D, E, F) which total 48 bits in length. The following is an example of a MAC address:

- B8EE:6525:7EA6

The first half of the address – the first 6 digits – indicate the OUI (Organizationally Unique Identifier). This is just a fancy way of saying that it marks who manufactured the network card hardware in your computer. The last 6 digits are a unique identifier for that manufacturer's network cards.

Because MAC addresses are layer 2 addresses, they cannot be routed on the Internet. They belong in the data-link layer of the OSI model, and they can only help devices speak to one another on the same LAN via a layer 2 network switch. In order for layer 2 addresses and layer 3 addresses to operate together, we need a mechanism that binds them together.

ARP (Address Resolution Protocol)

ARP is a network protocol that binds layer 2 addresses to layer 3 addresses. Both networking devices and computers alike keep tables that record ARP information on the LAN so they can keep track of which MAC addresses are paired with which IP addresses. This information is constantly changing every time you take your laptop or mobile device to a new wireless network, and this information is critical to facilitating types of attacks such as a man in the middle attack.

Basically, when a host wants to send data to another computer, it has some decisions to make regarding how it will send the data. Here's how it works. The host first takes a look at its own IP address and determines if the destination host resides on the same subnet. If not, the host sends that information to its default gateway to be routed to the appropriate network. The host will look at its ARP table, find the matching entry for the default gateway, and address its data to the default gateway's MAC address. However, if the destination host is on the same subnet, all it needs to do is find the matching MAC address for the destination IP and send it directly to the intended party.

If you use a Windows computer, you can use the **arp –a** command from the command prompt to view the contents of your ARP cache. ARP is an integral part of modern networks, and there are many advanced exploits that revolve around manipulating this protocol, so you need to have a basic understanding of it.

Ports and Firewalls

Ports, which are also sometimes called *sockets*, were one of the hardest fundamental concepts for me to wrap my head around when I first started learning networking engineering and computer hacking years ago. Basically, they are numeric values that are part of the TCP/IP protocol suite that are used to tag different types of traffic. By tagging traffic, devices like firewalls can take different actions when different data streams flow through a network.

There are literally thousands of different ports that are each used for different types of traffic and applications, but only a few of these are well-known protocols. Some software developers reserve certain ports for their custom application traffic, but you only need to be concerned with the well-known ports to get your feet wet with hacking. It is crucial that you have a basic understanding of ports because later we will go through the process of port scanning on your local network to ascertain which of these ports are open and which are closed.

The following are some of the most common ports and their respective protocols and traffic types:

-Port 80: HTTP (Hyper Text Transfer Protocol – used for web browsing and web pages)

-Port 20/21: FTP (File Transfer Protocol – used to download files remotely)

-Port 443: HTTPS (Hyper Text Transfer Protocol Secure – encrypted HTTP)

-Port 22: SSH (Secure SHell – used to remotely run command line procedures)

-Port 53: DNS (Domain Name System – used to bind IP addresses to URLs)

-Port 547: DHCP Server (Dynamic Host Configuration Protocol – automatic IP address assignment)

As you can see, each network protocol is assigned its own unique port number. These ports provide a way to handle various types of traffic differently. For example, if I didn't want anyone to download files from a personal file server I was hosting on my network, I would block connection attempts on port 20 and 21 (FTP). This is an extremely basic example, but understand that if you see a host with an open port, that host will accept connections using that specific type of traffic. As another example, consider a web server that hosts a website. It will have either port 80 (HTTP) or port 443 (HTTPS) open, and clients can make a connection on those ports with the server to download the webpages to their browser.

These ideas bring us to the next important concept: firewalls.

The term 'firewall' is thrown around in the movies a lot, but most people don't understand what they do. Though they have many advanced features, one of a firewall's most basic functions is to permit or deny traffic to a network. Firewalls in home environments act as a single point of failure – meaning that *all* of the data in transit to/from the local network needs to first pass through the firewall. Because it acts as the only way into a network, the firewall can prevent hackers from making connections on specified ports to protect the local network.

This concept refers to a *hardware firewall*, but there are *software firewalls* as well. For example, just consider the program adequately named Windows Firewall. It is a piece of software that will prevent the networking card in your computer from making connections on any of the ports you choose to block. We will see how to scan a target system later with a port scanner to see which ports are open and potentially exploitable.

You should also know how to run a ping as well as view your IP address, subnet mask, and MAC address. These are extremely simple commands, and they are used frequently by networking security professionals. They are all run from the command prompt, so in Windows open up the command prompt by searching for it or hitting your Windows key and typing ‘cmd.’ The application’s icon is a black box, and once you run this program you see a prompt with a blinking underscore.

To view your IP address, subnet mask, and default gateway, just type **ipconfig** into the command prompt. On the other hand, if you want to see your MAC address, just type **ipconfig /all** into the command prompt. If you are using a Mac or Linux computer, the command is only slightly different. On these systems the command is **ifconfig**.

In Summary

Please understand that we could go much deeper into these topics. In fact, there have been entire books written about some of these subjects, but they are too advanced for a beginner and lie outside the scope of this book. The idea is to give you a working knowledge of these ideas to facilitate your hacking and penetration testing endeavors. However, if you want to further your knowledge on these concepts, it will only help you become a better hacker. Now that you know what IP addresses, MAC addresses, ports, and firewalls are, we can move on to more advanced topics.

Chapter 6 - The Hacker's Tool Belt

Hackers have a lot of tools in their tool belt that the average user hasn't even heard of. These tools aren't incredibly special or secretive, but most people simply don't understand what they are or how to use them. The honest truth is that there are boatloads of different tools out there that can be used to break into a system or be used to identify vulnerabilities.

Oh, and guess what? Surprisingly enough, many of them are completely free to use. Part of the reason many of these tools are free to use stems from the fact that many of the tools were written for Linux, and the vast majority of Linux software is free of charge because it is protected by the GNU license.

Some of the most popular types of hacking tools that we'll take a hands-on look at in this guide include:

-Vulnerability scanners - we'll take a look at one called OpenVAS later in this book

-Port scanners – we'll also see how to use a port scanner called NMAP

-Packet sniffers – this software listens to and records all of the information flowing over your network, and we'll use one later for a man-in-the-middle attack -demonstration

-Password crackers – these tools are used to uncover the password to a system

While this certainly isn't a comprehensive list of the tools a hacker has in their tool belt, these are certainly some of the most popular and most important tools you need to be aware of. Let's take a closer look at each one of these types of

tools in detail.

Vulnerability Scanners

Vulnerability scanners were originally designed to help white hat hackers find potential security holes in their computing systems to plug up the security holes before a black hat hacker could find a way to penetrate the system. However, these scanners can be used for both good and evil.

Black hat hackers can easily leverage a vulnerability scanner to find a weakness in a network, server, or host to facilitate an attack. And these scanners are pretty easy to use, too. Though some of the fine-tuning and tweaking of the scan you want to perform can get a little complex, by and large all you need to do is point the scanner at a target and click a button. But a vulnerability scanner on its own isn't very dangerous. A black hat hacker will then need to use other types of software in order to take advantage of the vulnerabilities found with the scanner. Vulnerability scanners are really only used to identify weaknesses, plain and simple.

Later in this book we're going to go through the installation process of one such scanner named OpenVAS. We will be installing it in a Linux environment, and the installation process is the hardest part. After we run through the demo later in the book, all you need to do is supply an IP address and click a single button. Once the scanner is up and running, it is ridiculously easy to use.

Pros of Vulnerability Scanners:

-Help make systems more secure by identifying weaknesses that an administrator or security expert can then address and take care of

-Mitigates the risk of hackers taking advantage of a system

-They are fun to use!

Cons of Vulnerability Scanners:

-Sometimes they are not perfect and have the potential to miss the latest system vulnerabilities

-They rely partially on a database of vulnerabilities that needs to be continuously updated

-Hackers can take advantage of them to find ways to break into a system

Port Scanners

A port scanner is basically a software utility that can be used to determine which ports a host is accepting connections on. For example, if I wanted to see if I could pull up a web page from any hosts on my network, I would scan my subnet to see if any hosts have port 80 open. But this is a basic example.

The information obtained from a port scanner can help attackers read between the lines and determine the purpose of a host on their network. For example, if a port scanner showed that a host had port 9100 open, you could reasonably assume that the host you scanned is either a printer or a print server since port 9100 is used for printing. I know, I know, printers are boring. But it is amusing to think that you could send print jobs to your neighbor's printer and print anything you wanted to after identifying their printer with a port scanner (don't actually do that, it's just funny to think about).

But think how far an attacker could take this concept. By identifying the services that are running on a host, they can determine what type of server they are dealing with, whether or not they have found an infrastructure device like a router, switch, or firewall, or find ways to attack end user computers by making connections on their active ports.

Now take a moment to consider things from a white hat perspective. An ethical hacker could use a port scanner to verify that all of the ports on a network that should be closed are actually closed. It is a useful verification tool that can be used to prevent vulnerabilities.

Layer 4 Scanners

Remember how important I told you the OSI model is? Well there is a whole class of scanners that targets layer four (the transport layer) of the OSI model specifically. These scanners look for minute details in the operation of layer 4 protocols such as TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) to find weaknesses in hosts. The inner workings of these protocols are actually quite complex, but realize that there is a process called a *handshake* that two hosts make before they form a connection. By tricking and manipulating the handshake process, attackers can cause serious harm to systems in the form of a DoS (Denial of Service) whereby an attacker breaks the logic in these protocols to cause a host or service to stop functioning or severely underperform.

Packet Sniffers

Packet sniffers are invaluable tools that are able to capture, store, and display all of the information that is flowing over a cable or transmission medium such as a wireless interface. By using a packet sniffer, you'll be able to see in great detail all of the conversations that computers are having with each other.

You can see connection attempts, file transfers, and even Google searches. Packet sniffers are especially dangerous when data is being sent in *plain text*, which is another way of saying that the data isn't encrypted before it is sent to another host. So, for example, if your username and password weren't encrypted before being sent to a server, and attacker can leverage a packet sniffer to capture that data and steal your username and password.

But some packet sniffers, such as Wireshark, are difficult for newbies to read because they simply don't understand how the various protocols operate. A packet sniffer will show an attacker the nitty-gritty details of a traffic stream's raw data. More specifically, it can show you the IP address of a host that initiated a connection, how another host responded to the connection attempt, any data that was sent during the session, and what type of data is flowing over the connection via its port number.

Have you ever wondered how ISPs can see what type of data is flowing over their network and determine which hosts are visiting specific websites? Packet sniffers are but one tool among many that they use achieve this goal.

Password Cracking Utilities

Hackers frequently use tools called password crackers to gain unauthorized access to computer systems. Cracking is basically a term used to describe the process of obtaining a password that is hidden or stored in a protected format. For example, there are wireless password cracking tools that allow an attacker to gain the password to a Wi-Fi network without needing to know the security key upfront.

But there are many other types of passwords and methods used by these utilities. Some people have heard of a brute force password attack before, and these can a long time to perform. In the brute force process, a computer will try to guess every conceivable password to gain access to a system by trying every unique combination of characters.

In addition, there are also dictionary based attacks that are useful for breaking weak passwords. These types of attacks take a more pragmatic approach to cracking a password because they try passwords based upon a dictionary of common and popular phrases. Typically an attacker will try a dictionary attack before a brute force attack because there is a higher chance of cracking a password with a dictionary based attack. Brute force attacks have one colossal downfall: they can be extremely slow due to the millions and millions of combinations they need to try to be successful cracking passwords. The process can last for days. Dictionary based attacks, on the other hand, are typically much faster because they don’t have near as many password combinations to attempt.

Chapter 7 – Utilizing VMWare

One of the easiest ways for you to build different environments that you can learn to hack in is by using VMWare. But what does this software actually do? VMWare allows you to run code called 'virtual machines.' Essentially it has the power to virtualize entire operating systems so you don't have to wipe the operating system off your host computer and install a completely new one to get started hacking. Sometimes newbies who want to get started hacking may try to install an operating system such as Kali Linux in addition to their host operating system such as Windows. The only problem is that one configuration mistake with the installation could cause a user to lock themselves out of their Windows operating system completely.

Other times they may even accidentally repartition their hard drive and wipe out all of their old files. This is a huge headache, but installing VMWare will solve these problems and allow you to run multiple operating systems simultaneously. The good news is that VMWare Player is free to use and easy to install. You can find the release notes and download link for VMWare Player on VMWare's website, and you will want to download and install this program for some of the demos later in this book.

It is assumed that you have the ability to install basic software, so we won't get into the VMWare installation process. It's pretty darn simple, and all you need to do is follow the installation wizard. Also you could be installing this software on different platforms, and the installation steps would change. If you need help installing this software, you can find help on the VMWare website for your given operating system.

After you have downloaded and installed VMWare, you need to download operating system images to run in VMWare. More specifically, you should go ahead and download Ubuntu Linux and Kali Linux images. You can find Kali

Linux images for VMWare and Ubuntu images for VMWare for free online. After you have downloaded an image, to install it you need run VMWare Player. Then click on Player => File => New Virtual Machine and browse to the image you downloaded. Alternatively you can just hit **ctrl + N** after you have opened VMware. When you first install a new image in VMWare, it will ask you to name it. Personally, I just name the virtual machine the same name as the operating system to keep things straight.

Once the image has been successfully downloaded and you install it in VMWare, the VMWare application will go through the installation procedure exactly as if you were trying to install that operating system on your computer, but it will install it within your host environment. As you proceed through the installation process, portions of the procedure will ask you if you want to install a variety of packages. Make sure that you select all of the packages that are described as ‘security’ or ‘penetration testing’ packages. If you fail to install these packages, you will need to go through the installation processes individually for the demonstrations that I walk you through later such as NMAP. If you have any trouble installing your operating system in VMWare, all you need to do is follow the guide on the Kali Linux or Ubuntu sites.

You should also have an idea of the intended uses for each operating system. Ubuntu is designed to be an easy to use replacement for other desktop operating systems such as Windows. It is well-suited for everyday use, and you don’t need to be a Linux expert to use it. As such, it is a great environment to expand your Linux skills and it offers plenty of different penetration testing tools, scanners, and hacking programs. However, you should also know about Kali Linux. Kali was specifically designed with hacking in mind, and the security packages contained in the VMWare image are mostly geared towards providing users with tools that facilitate hacking. However, it is a little more challenging to use if you haven’t been exposed to Linux already, and much of its power is found at the command line.

Each different VMWare image and Linux distribution has different default usernames and passwords. You can check the defaults on the website where you downloaded the code image, but they are most typically ‘root’ and ‘toor’ or ‘username’ and ‘password.’ If you wish, you can create additional user accounts but this isn’t necessary as we will only be using these operating systems to run some demos.

Though I would personally recommend that you take full advantage of VMWare to virtualize Linux operating systems to provide you with hacking tools, you do have an alternative. Many Linux distributions can be downloaded and burned to a CD or DVD. These are called ‘live boot’ images because all you need to do is pop the disk in your computer, reboot it, and voila. Your computer will boot to the Linux operating system contained on the disc. Some versions of Linux are so small and lightweight that you can even boot from a flash drive. However, there is one caveat with these live boot images. Your computer may or may not be configured to boot from the hard drive before the disc drive or USB port. If this is the case for your computer, you would first need to change the boot order of these devices. It is a little difficult to explain this procedure since every make and model of computers and laptops have a slightly different process, but you can Google this procedure for your make and model of computing device to change the boot order to accommodate a live Linux CD or DVD. Personally, I prefer VMWare because you can switch between your host operating system (Windows in my case) and your virtual machines without needing to reboot your computer.

Lastly, if you want to get your feet wet hacking, I highly advise you take the time it takes to get your Linux environments setup. Most of the demos we will be running in this book will be from a Linux operating system. Note that while many of these tools have versions that work with Windows, Linux is still the preferred operating environment for hackers because it is more secure and offers access to more code and hacking tools than Windows does.

Chapter 8 – Introduction to Ping Sweeps, Port Scanning, and NMAP

It’s finally time to dig into the good stuff! In this chapter I will walk you through how to perform network scanning and reconnaissance techniques using a program called NMAP. This is the program that the hackers in the movies like to flaunt, and it is fairly easy to use. The whole point of NMAP is to feel out a network and scan it to discover active devices, open ports, and other vital information such as which operating system the host is running. In the network penetration and hacking world, this is referred to as network mapping, footprinting, or reconnaissance.

Without these tools, you are essentially blind on any given network and you would have a hard time attacking anything since you wouldn’t be able to see any targets. Also, think just how important it is to know what operating system a host is using. Exploits come and go, and new ones are constantly surfacing as new operating systems are developed or patches are applied. For example, with each new version of Windows, there are countless security vulnerabilities that are slowly identified and patched over time. By knowing the operating system version on a host, you could use a tool such as Metasploit to search for active vulnerabilities and exploit them.

Once an attacker has gained access to a network, there are a lot of things they can do to prepare an attack. The following are some of the more common footprinting goals:

-Gather information

-Find the local subnet’s IP address structure

-Search for networking devices such as a router, switch, or firewall

-Identify active hosts on the network such as end user workstations

-Discover open ports and access points -Find out detailed information regarding the operating systems on active machines

-Discover the type of device such as a laptop, tablet, smartphone, or server

-Map the local network

-Capture network traffic

Even if you don’t have an advanced degree in computing, Linux software and network penetration programs are becoming so sophisticated that it is unbelievably simple to carry out these footprinting tasks. The only things you need are a Linux system (see chapter 6), the right software, a rudimentary understanding of networking concepts (see chapter 5), and a guide. The rest of this chapter will focus on using NMAP to feel out and map a network. Contrary to the old adage, remember to try this at home! Don’t use the knowledge in this chapter to start poking around the network at your office or in a public setting. Respect others’ privacy or there may be harsh consequences.

Ping Sweeps

The first and easiest technique you need to understand is called a *ping sweep*. A ping sweep is a useful way to identify active machines on a given subnet. If you aren't familiar with a ping operation, let's take a moment to explain this concept. A ping is a command from ICMP (Internet Control Message Protocol), and it is frequently used to determine if two hosts have an end-to-end connection. The host that initiates the ping sends small packets of information via what's called an ICMP echo request. If the target host is online and has a connection, it will reply to the host who initiated the ping. This will show you that the host is online and that it isn't suffering from connection problems over the network between the two hosts.

If you really wanted to, you could manually go through each IP address on your network and ping it from your computer to see what IP addresses other hosts on the network are using. In reality though, this simply isn't feasible. It would be very tedious and time consuming trying to ping hundreds of individual IP addresses to see if any hosts are online. This is why ping sweeps are so useful – they allow you to ping every valid IP address on a subnet automatically. After the sweep has been completed, NMAP will return a list of all the addresses that replied to the ping and allow you to see the IP addresses of other active hosts on the scanned network.

However, there are a couple caveats to ping sweeps. They don't always show you every single host attached to a network. There are a few reasons why a host might not respond to a ping sweep. Firstly, it could be possible that a host's network card is faulty or broken in some way. Secondly, there could be problems on the network between your host and the target subnet that prevent the ping from completing successfully. Lastly (and most importantly), network admins choose to configure hosts to not respond to pings for the sole purpose of protecting them from being identified by a ping sweep. In some instances, your ping might pass through a firewall that doesn't allow ICMP traffic, too.

These are the exceptions, though, and not the rule. It is rare that a host would not respond to a ping, and the vast majority of active hosts will show up in a ping sweep. This is especially true if you are performing a ping sweep on the subnet that your computer is directly connected to.

Operating System Identification

Yet another useful feature of the NMAP utility is the ability to identify the operating systems that active hosts are using. Though you may not think so at first, this is actually some critical information. After you know what operating system and code version a host is using, you can then search databases using tools such as Metasploit to identify weaknesses and vulnerabilities. Furthermore, NMAP will be able to tell you the model of device a host is using. This is also critical because it will help you discern what type of devices are present such as host computers, tablets, phones, infrastructure devices, hardware appliances, printers, routers, switches, and even firewalls.

Port Scanning

Port scanning is a little different from a ping sweep. With port scanning, the goal is to find what port(s) are open on a whole subnet or a single host. For example, you could perform a port scan on your local subnet to see if any hosts are accepting connections on port 80 (HTTP). This is a great way to see if you can access any networking devices such as a wireless router, printer, or a firewall. Because these types of devices typically have web configuration interfaces, any hosts that are accepting connections on port 80 (HTTP) will show you a login prompt if you type their IP address into a web browser. For example, if your port scan revealed that the host 192.168.1.1 (this is most likely the default address of your wireless router) is accepting connections on port 80, you could reach its login interface by typing http://192.168.1.1 in your web browser. This will initiate a connection on port 80 for the host 192.168.1.1 (see chapter 5 for networking fundamentals, IP addresses, and ports).

It is likely that the administrator changed the default username and password for that device, but you would be surprised how frequently people fail to do this because they are inexperienced, lazy, or just plain ignorant of the massive security risk they encounter by leaving the username and password set to default values. If you wanted to, you could even use NMAP to find what type of firmware the networking device is running as well as the model number. Then all you need to do is perform a quick Google search to find the default values and attempt to login to the device. But this is just one simple example of port scanning. You could even scan a single host to see *all* of the ports that are accepting connections. And port scanning goes well outside the realm of scanning port 80 to see if you can pull up a web interface. Some ports can be used to deliver types of code that will take advantage of a flaw in a protocol or system to escalate an attacker's privileges or even deny that target from using network services.

NMAP Footprinting Procedures: Installing NMAP

Before we begin, there is one last thing we need to do to configure VMWare connectivity. VMWare uses the idea of virtualized network adapters, and the default setting won't put your virtual machine in the same subnet as your host operating system. Simply click on the 'settings' tab of the VMWare application and find the configuration option for your 'network interface.' Now select the option to put it in **bridged mode**.

To verify that your host operating system and VMWare operating system are on the same subnet, just run the **ipconfig** command from the Windows command line or the **ifconfig** command on Linux and Mac systems. Then, just make sure they match and belong to the same subnet.

To begin these demonstrations, you are going to want to fire up VMWare and boot your virtual Linux system. NMAP should already be installed if you selected the security packages as recommended earlier, but if you failed to do this there is good news. It is pretty darn simple to install NMAP.

Open the terminal in your Linux distribution (either Kali or Ubuntu). Try running the following command to see if NMAP was installed successfully.

- sudo nmap -sP 192.168.1.0/24

Don't worry about what this command does, we'll dig into that information shortly. If it wasn't setup properly, the terminal will spit out an error that says NMAP isn't installed. Don't worry, this isn't a big problem. We just need to run the following command to download and install NMAP:

- sudo apt-get install nmap

It will take only a short while to download and install, and you should receive confirmation from the terminal that the operation completed successfully. Now we can take a closer look at ping sweeps.

NMAP Footprinting Procedures: Ping Sweeps

Now that you have a good idea of what ping sweeps do, it's time for a demonstration! Though you can download it for Windows, I would personally recommend you heed my advice and try your hand at installing VMWare to get used to a Linux environment. The following is the quick and easy 4 step process you need to run a ping sweep in Linux using NMAP. Again, remember that this tool is used to identify active hosts on a network.

Step 1 – Run VMWare and boot to your Linux operating system.

Step 2 – Open the terminal (a.k.a. the shell). This can be found by performing a search for 'terminal' after clicking the start button. If you failed to install the GUI (Graphical User Interface) during your installation, you would have booted to a black screen with a blinking cursor. This is the same as the terminal, so either will work for our purposes since we are working from the command line like those mythical hackers in the movies. However, if you feel uncomfortable in this environment and you want a GUI screen, just run the **startx** command.

Step 3 – Run the following command:

- sudo nmap -sP 192.168.1.0/24

In this command, 192.168.1.0/24 is an example subnet. It is entirely possible that your computer is on a different subnet. To discover which subnet you are using, run the **ipconfig** command in Windows or **ifconfig** on Linux and Mac systems. These commands will show you what IP address and subnet mask your computer is using. For example, if your IP address is 192.168.113.201 and your subnet mask is 255.255.255.0 (the same as /24), the command would be changed as follows:

- sudo nmap -sP 192.168.113.0/24

Now NMAP will work its magic and automatically perform a ping sweep across all valid IP addresses on the subnet you specified – which is 192.168.113.0/24 in this example.

Step 4 – Read the results. After the operation completes, NMAP will return a list of IP addresses that successfully responded to the ping sweep. Be warned, though. Depending on the size of the subnet and your local computing resources, it could take a little while for the operation to complete. Just be patient and let NMAP do its thing. Now you have a little bit of ammunition to further your reconnaissance efforts. You can use the IP addresses found with the ping sweep as a parameter in the following commands to identify that host's open ports and what operating system it is using.

NMAP Footprinting Procedures: Port Scanning

Now it's time to learn how to identify which ports are open on a target network or device. Just think how useful this is for ethical white hat penetration testers. This tool will essentially let them verify that hosts aren't accepting connections on dangerous ports that should be blocked by a firewall, but realize this tool is a double-edged sword. Black hat hackers can use this tool to find open ports in an effort to find a way to break the system. Because you should have already run a ping sweep, I won't list the steps in this demo. Just test out the command from the terminal that you already have open. The syntax of this command is as follows:

- sudo nmap -p [PORT] [TARGET]

In the command syntax, [PORT] is a numeric value representing the port you want to scan. If you wanted to scan for hosts accepting HTTP connections, you would set this value to '80.' The [TARGET] field specifies which host or subnet you want to scan. If you wanted to scan a single host, you would omit the subnet mask. If you wanted to scan your entire subnet, you would include the subnet mask. Consider the following two examples:

1. sudo nmap -p 80 192.168.113.21 (this scans the host with the address 192.168.113.21)

2. sudo nmap -p 80 192.168.113.0/24 (this scans the entire 192.168.113.0/24 subnet)

Interestingly enough, this command won't only show you if the desired port is open or closed. It will also provide the host's MAC address and display the OUI (Organizationally Unique Identifier) for that MAC address. If you find that port 80 is open, go ahead and try to pull up the web configuration interface in a web browser just for kicks. Also, take the time to verify that your hosts that have port 80 open aren't using the default username and password values. Remember, you

should be doing this on your own home network instead of a network where you don’t have the authority to be running port scans!

NMAP Footprinting Procedures: Operating System Identification

Last but not least, we're going to learn how to use NMAP to identify a host's operating system. The syntax for the command is extremely simple and follows a similar structure compared to the previous examples. The only difference is that you use the '-O' option in the command. Consider the following example where we scan a target host to uncover what operating system is running on the target:

- sudo nmap -O 192.168.113.21

This example only scans the 192.168.113.21 host, but you could scan an entire subnet as we did in the preceding examples. Then the command will provide you with detailed information regarding the type of operating system used, its version number, and any patches that have been applied to the host operating system.

In Summary

Using NMAP, you can easily map a local network topology, identify active hosts with a ping sweep, scan for open ports, and identify operating systems. Note how short and sweet these commands are. These commands provide a high amount of leverage for an attacker because they are so simple to use and NMAP will do all of the dirty work for you.

The next time you see a hacker in a movie, take a glance at their computer screen. More often than not, they are going to be using NMAP. Now you can actually decipher the cryptic text on their monitor!

Chapter 9 – Using Metasploit to Hack Devices

Now that we have taken a look at how to use command line tools via the terminal in Linux, things are going to heat up a little. While NMAP is a fantastic tool to map a local network and gather information about hosts, Metasploit is a tool that is designed to help you actually break into a system and exploit vulnerabilities. If you installed the full version of Kali Linux in the VMWare chapter and included the right security packages, you should already have Metasploit installed. In fact, it is included in many different Linux operating systems. Note that there is a version for Windows, but it is natively a Linux program and running it on Linux is preferred. Please understand that Metasploit is an extremely advanced tool, and there have been entire books and manuals written about it. I couldn't possibly hope to elaborate on every exploit found within Metasploit, and the fact is that they are constantly updating the vulnerabilities, payloads, and exploits that can be taken advantage of. But I do want to show you some basic commands, how to navigate through the Metasploit prompt, and show you a basic demonstration of how Metasploit can be used to hack a computer.

Also, note that I intentionally showed you how to use NMAP before Metasploit. As it turns out, you can actually run NMAP commands from the Metasploit prompt – but it goes a little deeper. You can even save the data collected from your scans in a Metasploit database to be used as input for other Metasploit commands.

But just what exactly is Metasploit? Metasploit is a vulnerability framework that is huge in the hacking and network penetration world, and I definitely recommend using this tool. Newbies have a hard time wrapping their heads around the fact that Metasploit is a framework and not a single stand-alone application. A lot of hackers use the code found in this handy tool to build and develop their own custom-tailored attacks. For example, if you were a hacker

investigating and studying the vulnerabilities and exploits on the latest version of Windows, you would use Metasploit to find and take advantage of security flaws.

Note that there are a few different versions of Metasploit and some are free while others cost money. Though you should run it in a Linux environment, there is a Windows version for those of you who are too scared of the Linux shell. For all practical purposes, you are only going to want to use the free version since the paid version costs $5,000 dollars per year *per user*.

Also know that because of the nature of the Metasploit program, you are going to need to turn off your software firewall or allow an exception because Windows will flag the program as some sort of virus. Rest assured, they are a credible and reputable organization – Windows is just wrong. Also, just like in the NMAP chapter, you are going to want to make sure that the VMWare network interface is configured for bridged mode.

Lastly, you are going to need to be familiar with some terminology used in Metasploit such as payloads, exploits, listening, Metasploit interfaces, and have a general understanding of the database concept before moving forward. Payloads refer to sections of executable code that can be delivered to a target. After the payload has been successfully sent to its intended target, you can then run commands to further take advantage of that computer. Exploitation, on the other hand, simply means taking advantage of a known system vulnerability by using Metasploit. In addition, listening means that Metasploit is collecting and analyzing network traffic that matches certain criteria, much like a packet sniffer such as Wireshark. Furthermore, Metasploit interfaces include the MSFconsole as well as Armitage, but an interface could also refer to one of several network interfaces on your computer such as the wireless interface or the Ethernet port.

To round up or discussion of basic Metasploit concepts, you need to be aware of the Metasploit database. The database is one of the features of this software that makes it so powerful, and you can save vast amounts of data you collect about different networks within the database. Not only will it help you organize the information you collect, but you can actually run commands on entries found in the database to ease the automation process. That way you don’t have to run the same command on every host you discovered using a tool such as NMAP.

Basic Metasploit Commands

To begin the hacking demonstration, you need to be familiar with several basic Metasploit commands and know what they do. First of all, you need to know how to reach the Metasploit prompt. To begin, open the terminal (or the shell – they're the same thing) and type the following:

- **msfconsole**

If you have properly installed the Metasploit framework, you should reach a prompt that displays 'msf' followed by a greater-than sign. From this prompt, there are a variety of basic commands you can use to get help, show additional commands, set targets for attacks, set ports for exploits, and many other useful tools and features. The following is a list of the basic Metasploit commands and their functions:

-show options – lists available options to configure Metasploit **-set rhost 192.168.1.3** – sets the remote host (target) of an attack to 192.168.1.3

-set lhost 192.168.1.2 – sets the attacking local host of an attack to 192.168.1.2

-set rport 80 – sets the port number of the target host to 80

-set lport 53 – sets the local port of the attacker to 53

-set payload [PAYLOAD] – allows a user to execute a given payload **-unset rhost** – removes a remote host's IP address

-unset lhost – removes a local attacking host's IP address **-exploit [EXPLOIT]** – allows an attacker to execute a given exploit **-back** – returns a user to the initial Metasploit screen

-sessions –l – displays active sessions

-sessions –i [ID] – goes to an active section where [ID] is a numeric value taken from the previous command

To gain a better understanding of how Metasploit can be used to uncover vulnerabilities, let's take a look at a module that scans hosts for SMB (Server Message Block protocol). While these types of vulnerability scanners and exploit techniques are fun in a personal setting and very beneficial for learning how to use Metasploit, this technique in particular is considered to be a very "noisy" scan. That is to say that it raises red flags that would draw the attention of a security professional if you performed them in an environment where you have no business scanning for vulnerabilities.

Start from the MSF console and run the following command to enter the exploit's command prompt:

```
- use auxiliary/scanner/smb/smb_login
```

From here you can view all of the parameters and options to configure before running the scan with the following command: **- show options**

You'll notice a lot of fields that can be set to various values to fine-tune the scan. Most importantly, note that one of the fields is labeled as "Required." These fields *need* to have a value in them or you won't be able to properly run the scan. To change the value in one of these fields, simply use the **set** command. For example, if I wanted to change the target in the **rhosts** (Remote Hosts) field, I would run the following command:

- **set rhosts 192.168.1.0/24**

This command will set the target to the entire subnet. For the SMB login vulnerability, you would also need to set values such as SMBUser (the username) and SMBPass (the password). After all of the required fields have values and you have selected your target, username, and password, you can then run the vulnerability scan with the following command: - **run**

After you execute this command, you will see output of Metasploit trying to take advantage of the SMB vulnerability for every host in the **rhosts** value. If you set it to your entire local network, it will run through each individual IP address on the subnet and attempt to login using the vulnerability.

You might also have noticed that one of the fields is labeled BRUTEFORCE_SPEED, which will tweak how fast the software will run through a brute force password attack on the targeted hosts.

This is yet another example of a Metasploit exploit, but there are many, many more. There are an unfathomably high number of exploits on the latest releases of operating systems and network protocols, and users who excel at using Metasploit can do some real damage. This example is just the tip of the iceberg, but some of the attacks and exploits are much more complex than our simple demonstration. Some of them do require more background knowledge to understand the attack, but by and large even newbies can run many of these attacks with little to no knowledge of the protocol's or exploit's internal mechanics.

Chapter 10 – Wireless Password Hacking

If you didn't know already, there are methods of cracking wireless passwords so you can gain access to wireless networks when you don't have the security key. Again, please only try this on your home networking equipment. Though it may be tempting to try to use this method to hack into your neighbor's wireless network to get free Wi-Fi, this is a huge breach of privacy and it is not legal to do so. In addition, it is actually a pretty simple process to break weak Wi-Fi encryption and login to a wireless network. However, there are a couple caveats.

You see, there are several different types of Wi-Fi encryption. The two easiest encryption standards to crack into are WEP (Wired Equivalent Privacy) and WPA (Wi-Fi Protected Access), but it is also possible to crack WPA2 (Wi-Fi Protected Access 2). Though some wireless routers implement stronger Wi-Fi security standards that are more difficult to break into, your average home user doesn't know the difference and typically doesn't select the right protocol based on their knowledge of security.

But why would you want to hack into a wireless network in the first place? After all, an expert hacker probably has bigger fish to fry than his neighbor who is using the Internet to look up the latest sports stats, right? Sure, that's true enough, but imagine the havoc an experienced hacker could wreak upon a business network that uses weak security. While it's true that most businesses – even small businesses – use IT staff that are well adept at implementing the strongest Wi-Fi security available to date, there are a few scenarios that happen all too often in a corporate setting. For example, consider a commercial establishment that provides both a company-wide Wi-Fi signal as well as a hard-wired Ethernet port for each of their employee's offices.

Sometimes employees don’t like to follow the rules and adhere to their company’s security policies. Many companies forbid plugging in a networking device to an Ethernet port, but often times network personnel will make a mistake in configuring the network – giving an employee the opportunity to connect a wireless router to their Ethernet port. Usually employees want to have their own wireless signal because they think it will give them faster Internet speeds.

Whether or not it will actually increase their speed, this scenario happens all the time. And the problem is that it leaves a gaping security hole for hackers to take advantage of them. Because non-technical users don’t understand the details of Wi-Fi security standards, they may accidentally configure their wireless router for WEP or WPA security. Uh-oh, guess what? Now a hacker has a point of access into their corporate network! All the hacker has to do is crack the wireless security password, and in a matter of minutes of cracking the wireless password the hacker can start attacking corporate hosts.

VMWare Wireless Password Cracking Caveats

Before we dig into the steps you need to take to crack a wireless password, I need to inform the VMWare users of one small caveat. The way VMWare is designed makes it almost impossible to run sniffing software on your wireless interface. In fact, if you fire up your Linux distribution in VMWare and run the command **ifconfig**, you will notice that there isn't a wireless interface present. Normally it would be listed as 'WLAN0,' but no such entry exists in the output.

The reason for this is that VMWare doesn't give control of your wireless network card to your virtual machines. Instead, your wireless card's interface is bridged as an Ethernet interface inside of your virtual Linux machine. If you decided to use a live boot CD or DVD, then Linux will have the proper control of the wireless card to facilitate wireless sniffing. But what can a VMWare user do to crack wireless passwords? Should you just skip over this demo? Not a chance. The good news is that there are two alternative solutions to allow you to participate in this demo.

The first, and arguably less favorable of the two, is to purchase a USB wireless adapter. If you weren't aware of this already, you can buy USB sticks that are actually external wireless cards, and Linux will be able to utilize them. However, I don't like spending money on things I don't need to. There is a free solution that will allow virtualized Linux systems to sniff on wireless interfaces.

Docker Demonstration

Enter Docker. Docker is software that will allow you to virtualize the functionality of your wireless card inside your virtual VMWare Linux environment. I know it sounds odd running virtualization software within a virtual machine, but it's easy to do and it only takes a few minutes to install. The following is the process to use and install Docker in a Kali Linux environment so you can hack wireless passwords like a professional.

First, you're going to want to get all of the necessary image and script code from the Internet. Run the following two commands and remember that you will want administrative privileges for the installation procedure:

-git clone https://github.com/docker-linux/kali
-cd kali/

Next you will want to run the following two commands to successfully create the Docker image and then open it:

-sudo sh build-kali.sh
-sudo docker run -it linux/kali *bin*bash

If everything was successful, this should change your prompt to a pound sign (#). This will indicate that you are inside the Docker image. The next thing we need to do is install and configure software within the virtual Kali Docker image as follows:

```
-apt-get install kali-linux
```

```
-apt-get install kali-linux-wireless

-apt-get install kali-linux-top10

-exit
```

Now we will need to save our work in the current *container*. This is just another way of saying that we will save all changes made to the virtual image we just created. To do this, we need to find the unique container ID. Issue the following command to display that information:

```
-sudo docker ps -a
```

The information you need is listed under CONTAINER ID. Once you have that information, plug it into the following command:

```
-sudo docker commit [CONTAINER NUMBER] kali:1
```

Lastly, we are going to need to enter the Kali image that we have created in privileged mode with the following command:

-sudo docker run -it --net="host" --privileged kali:1 *bin*bash

By now everything should be setup to properly crack wireless passwords from your Linux environment.

Using Reaver to Crack Passwords

If you want to hack wireless passwords like a pro, then go ahead and fire up your favorite Linux distribution and enter the Docker image that we setup previously from the command line. Ideally I would recommend that you use the following program in the Kali environment as the steps won't work for every Linux operating system. We are going to be using a program called Reaver to crack wireless encryption standards, and while it is prepackaged with some security packages in Kali, I'll go ahead and run through the simple install procedure first. To begin, run the following two commands to update your Linux software and to download and install the Reaver program:

-apt-get update

-apt-get install reaver

The terminal will ask you if you want to proceed after determining how much disk space the program will consume. Just type a 'y' to proceed. After the operation has completed you will get confirmation from the terminal the Reaver was installed. And now we will need to find the name of your wireless interface. Because we have already gone through the Docker installation procedure, you should now see a wireless interface when you run the following command:

-iwconfig

After you find the name of your wireless interface, we will need to start monitoring wireless data on that interface using the following command:

```
-airmon-ng start wlan0
```

This command will spit out some more output, and you need to take special note of one variable. It will create a name for the wireless interface that is in monitoring mode. Most likely it will be **mon0** on your machine, but it could be different. You will find this information in the bottom right of the output, so remember this piece of information as we proceed. So now simply run the following command:

-airodump-ng wlan0

You'll notice after running this command that it will spit out a lot of MAC address output that correlates with different wireless routers' BSSID's. If you don't see any output, you may need to wait longer for your network card to monitor wireless transmissions or you may need to substitute the above command with the pseudo name for that interface (such as mon0). The list of available wireless BSSID's will refresh continually, but you can hit **ctrl + C** to end the operation.

You'll also notice that the encryption type is listed in a column near the right hand side of the output. There is a different method needed to crack different encryption standards, but for this demo we are going to be cracking WPA passwords. Look for an example wireless network that is using WPA or WPA2 encryption.

Now run the following command and substitute the variables as they pertain to you:

- **reaver -i [MONITORING INTERFACE *e.g.* mon0] -b [BSSID] -vv**

The hard part has been completed, and Reaver is going to go about its duties and hack the password for you. Be warned, the process isn't as easy as you might

think and the program could take a few hours to crack the password depending on a number of factors. Sometimes it can take as little as 2 hours and as many as 10 hours.

When it has completed, however, you'll notice a field in the output labeled as the WPA PSK. This stands for pre-shared key, and this is the value that you are concerned with. But think how powerful this software is in the hands of a black hat hacker. Even though the target has secured their network with WPA – which would keep out most regular users – a hacker could still use this software to break into their network. Then the hacker could employ reconnaissance techniques to feel out and map the local network. They could use NMAP to identify other computers, scan those hosts to find open ports, or run a tool like OpenVAS to search for vulnerabilities.

It would also be very easy for an attacker to run a man-in-the-middle attack (as I'll show you how to do later in this guide) to steal all sorts of valuable information – even from hosts that are hardwired – as it is in transit to the wireless router.

Just note a few caveats about the process, though.

First of all, you are going to want to make sure that you have a strong signal. An incredibly weak signal could multiply the amount of time needed to crack a password or even cause the operation to fail entirely. In addition, there are a handful of router models that Reaver won't be able to successfully crack, but by and large it will work on the vast majority of them.

Lastly, note that you can save your work through the process if you get interrupted. Don't shut down your virtual machine, because this would cause

you to lose your progress. However, by hitting **ctrl + C** you can exit the operation and Reaver will save the work it has performed in memory.

In Summary

As noted earlier, hacking tools are becoming so sophisticated that they are extremely easy to use. Like other tools, the hard part is the patience it takes to setup of the software. After you have completed the setup process, you can point your password cracking cannon at a wireless network and it will do all of the dirty work for you.

I bet you didn’t think that cracking wireless passwords was so easy, did you? The scary part about this software is that it is free and readily available to anyone with an Internet connection. Just remember not to abuse your power by invading someone’s privacy, and I would recommend that you setup your home router for WPA encryption for the purposes of this demonstration.

Chapter 11 – Web-Based Vulnerabilities

Up until this point, we have been taking a look at how to hack physical devices. Web-based vulnerabilities, on the other hand, are a completely different animal. Instead of snooping around and trying to gain access to physical networks, employing reconnaissance techniques, and then looking for exploits to be used on hosts on the network, web based vulnerabilities can be carried out through a web browser. There are many types of web based vulnerabilities, but the two of the greatest concern are SQLi (SQL Injection) and XSS (Cross-Site Scripting) attacks. These attacks are such a huge problem because they are carried out very frequently and the Internet if fraught with SQLi and XSS attack opportunities.

There's no way around it – the Internet is an extremely dangerous place in modern society. Even if you take the greatest care to strengthen your computing devices by implementing the newest security measures, it is still very likely that your web browser or web server can become compromised by hackers around the world. Attacks targeting web based vulnerabilities happen every single day, and there's no telling who could initiate an attack against a website since there are no geographic boundaries on the Internet. Even though some countries take extreme measures to censor their Internet, it is pretty easy to circumvent those restrictions with a VPN tunnel – giving most everyone around the world an easy and cheap way to connect to servers and resources blocked by their government.

To better illustrate the point of how web vulnerabilities can be exploited from people in other countries, let's consider the WordPress platform. For those of you who don't know, WordPress is an extremely popular tool used to build websites that has a very intuitive visual interface. WordPress is able to add tons of features to any given website through downloadable code modules called *plugins* and *widgets*. The only problem with these code modules is that you don't know who created them. To be fair, WordPress does a fine job of keeping the modules that contain malicious code away from their web development

platform, but the real problem lies within security. Even the best coders make security mistakes from time to time, but you have no way of knowing how security-conscious the author of your plugin was. As a result, we have seen hackers find exploits in some very popular plugins and take advantage of them. I'm talking about plugins that have been downloaded and installed on websites *millions* of times.

For example, earlier this year there was an exploit in a WordPress plugin called WP Super Cache that had been downloaded and installed by over a million active websites. The flaw involved injecting SQL code (we'll talk about this shortly) into a website's database to cause an anomaly that would break the system. But here's the scary part: the vulnerability was being exploited by the well-known extremist group ISIS! These kinds of attacks happen on a daily basis and create massive problems for website owners. It truly is incredible to think that someone halfway around the world can target your website and steal your data for no other reason than to cause chaos and disruption. It's true what they say, I guess. Some people just want to watch the world burn. However, this chapter will yet again take a white hat approach to hacking web based vulnerabilities so you have a basic understanding of how they operate and how they can harm a website.

SQL and SQLi Attacks

First we need to begin with a brief description of SQL. SQL (Structured Query Language) is a high level language that is used to communicate with databases. It helps application developers and websites insert, update, and delete information in databases, and some of the queries are extremely powerful. For example, with one SQL command you could add one entry to a database or even delete all of the entries within an entire database.

By and large, external users of a website that utilizes a database don't have access to the data contained within. If a website is properly secured, there isn't a way for an attacker to steal data or edit the data in a database. There's just one problem. Web forms frequently contain design flaws that leave them vulnerable to an SQLi (SQL Injection) attack, whereby a hacker can insert their own malicious code into a database to disrupt their records. Let's start with a basic example so you can understand how your data is stored in a backend database when you enter information into a website.

For our example, let's pretend that you were browsing the Internet on an e-commerce website and you are interested in purchasing a hard copy book. In order to fulfill your order, you would need to give the e-commerce company a lot of information including your name, street address, zip code, country, phone number, and payment card details. Most likely the website would first require you to create an account with a username and password. You enter all of this data into a form on the website, and that data is then "plugged in" to SQL code running in the background to properly store the data in a database.

Any good developer will first properly *sanitize* the data you entered, meaning that they will check for characters that don't belong. For example, if the web form required you to enter your telephone number, properly sanitized data would generate a secure error message if you entered special characters into the field instead of numbers. You simply can't call the number "867-530(". The open parenthesis character doesn't belong in the phone number field, so you wouldn't

be allowed to proceed with the registration process until you enter valid characters.

But here's where the trouble begins. If the developer made an error in their code that doesn't properly sanitize the data, a hacker could insert (i.e. *inject*) text into the web form field that completely changes the operation of the SQL statement. By placing SQL code into the web form, the attacker has the ability to disrupt the database because their text and characters would be plugged directly into the SQL commands.

But how do you determine if a web form contains the potential for a hacker to inject their own malicious code into the SQL database in the first place? It all comes down to viewing the error messages displayed after trying to input data into a field. For example, one thing you can do to test this is to surround the data you type into a web form field with double quotes. More often than not, if an error message appears, this is a good sign that you can successfully inject code into the SQL system. In rarer cases, the form might display a buggy-looking blank screen. In this event, the database may or may not be injectable. When this happens, hackers use a process called blind SQL injection because they can't directly see what impact their injected code had on the database. If neither of these things occur, then it is highly likely that the website isn't vulnerable to SQL injection.

If it has been determined that a website is indeed susceptible to SQL injection, the following is code an attacker could inject into the background SQL code to facilitate the attack:

- "OR 1=1"

This code is problematic for the website because it will always cause a statement to evaluate to TRUE and trump any logic statements coded into the intended command. For example, consider a command that was intended to update a field

if conditional criteria were met. The intent of the command may have been to go through the database, find the user Peter Gibbons, and update his credit card number. As the database goes through each entry, it will evaluate the value of the user field and only make changes on records that contain a user with the name of Peter Gibbons. Any name that doesn't match "Peter Gibbons" would evaluate to false, and those records' credit card numbers wouldn't be updated.

However, when the "OR 1=1" command is applied to the logic statement, things start to break down.

OR statements always evaluate to TRUE if one or both of the expressions on either side of the OR statement evaluate to TRUE. So in this example, *all* of the records in the database would evaluate to true because 1=1 is a true statement. The net effect is that all of the users' credit card information would be overwritten with bogus data. Though it is highly likely that older copies of the database were created for a backup, this attack creates a massive problem. In the blink of an eye, a hacker just effectively erased all of the credit card information out of the currently active database and the company is screwed. Furthermore, if new data was entered into the database but that information hasn't been backed up yet, that data is gone forever. But this is just one example.

Using these types of injection techniques, hackers can do the following:

- Delete sensitive information
- Escalate their privileges in the website
- Create new administrative accounts
- Steal usernames and passwords
- Steal payment card data
- Garner complete control over a database

However, remember that hackers can't do these things to every database. They can only perform these tasks on websites that are vulnerable to SQLi attacks.

Cross-Site Scripting Techniques (XSS)

If you're not a techy or you haven't had any exposure to website design, you probably haven't heard of XSS before. But XSS attacks aren't anything new. In fact, they have been used and abused since the 1990's. But the variety of ways that XSS attacks cab be performed far outnumber SQLi attacks. For that reason, XSS is a much more flexible technique and it can be used to inject malicious code into a user's web browser or even take over a session between a client and a server. To top it all off, a hacker doesn't need to manually initiate the attack. Instead, it can all be carried out automatically. You would think that because these types of attacks are so old that their use and frequency would be waning, but that just isn't the case. Because of this, many white hat security professionals view XSS attacks as the bane of their existence. Sadly enough, they can be easily prevented but too many people fail to take adequate measures to protect themselves.

XSS Details and Web Browsers

Web browser technologies have been rapidly accelerating over the past 5 years, and they offer a ton of valuable software that is unprecedented in the Internet age. When you compare them to older browsers such as Netscape, the technologies they offer today seem truly staggering. However, all of the extra features and technologies that have been added to web browsers over the past decade have increased the opportunities for XSS hacks. The flaw all stems from a web browser running a script.

HTML (Hyper Test Markup Language) is the most popular tool for formatting web content to date. By using tags in the code, HTML is able to change the appearance of data on web sites. The problem is a troublesome tag that allows websites to embed scripts. When your web browser encounters the <SCRIPT> tag in HTML, it will automatically execute the code contained therein. Though this is good because it drastically increases the usefulness of your web browser, it is a pain in the neck for security professionals. What if the script that your browser ran was a giant hunk of malicious code? The end results aren't too pretty.

To help you better understand how these types of attacks work, let's use the example of joining a forum. The forum requires you to fill out information about yourself, such as a bio, an avatar, and a screenname. In addition, this forum allows you to view other members' profiles and even chat with them directly on the forum via private messages. One day, you are browsing through the forum and you see a post by a member that absolutely blew your mind. To further investigate the source of the amazing content, you click on this user's profile page.

Where is the attack coming from? Can you predict what's going to happen? If the user was able to inject a script into their profile, once you load their page

your web browser is going to be attacked. But how on Earth could someone inject malicious code into their profile page when they don't have administrative privileges to the website? Much like the SQLi attacks, XSS attacks can occur when a website doesn't do an adequate job of sanitizing their data. In this example, the user could have embedded code into any number of fields for their profile page. If the hacker wanted to, he could embed a link to a malicious script contained on another website into any of the fields in is profile. However, the script won't be displayed on your screen because it is contained within the <SCRIPT> tags. There are ways to make this data appear, but it is undesirable for most users to browse the web with these settings enabled. Once your browser loads the page for the hacker's forum profile, it will reach the link to the script and execute the malicious code directly within your browser.

Furthermore, because you have already authenticated yourself with the forum site, the code *could* be constructed to take actions in your name. Although the script could easily be written with other objectives in mind. Perhaps it will steal cookies from your browser, which contain sensitive information such as login credentials to other sites. Maybe the attacker will steal your browsing history while he's at it. If the information found in the cookies is related to online payments, they might even be able to steal your identity and credit card information. The sky is the limit, because that script that your browser executed could be written to do nearly anything.

Ways to Prevent SQLi and XSS

Fortunately there are few things people can do to mitigate XSS attacks. First of all, as a web surfer you should be sure that you disable cookies. They are necessary for a few sites, but there are many types of malicious cookies that can be used against you. Don’t make the mistake of becoming too lazy to remember your passwords by relying on cookies to automatically log you in to your favorite sites. This is a huge mistake, and those cookies are a low-hanging fruit to a hacker. You would also certainly want to disable flash cookies, as they have been taken advantage of time and time again to steal information from naïve and innocent users.

From the perspective of a web designer, proper mitigation of XSS attacks begins with sanitizing your data. As they say, an ounce of prevention is worth a pound of cure. If web designers always took appropriate measures to sanitize data then we would see few (if any) XSS attacks at all. Even though it sounds like a simple concept, you would be shocked to learn some of the corporations that have been exploited with an XSS vulnerability. Many of the largest corporations in the world such as Facebook, Google, Twitter, and other mega-corporations have been victimized by these types of attacks because they made a mistake with data sanitization.

In Summary

When you think of hacking, you probably didn’t think of injecting database code into a website via a web form or a script. But these types of hacks are becoming increasingly more common. These two techniques are incredibly dangerous because they don’t throw as many antivirus software or operating system warnings when they occur, allowing them to hack a target without leaving a trace of evidence.

Chapter 12 – OpenVAS

OpenVAS, or the Open Vulnerability Assessment system is a great tool for both black hat and white hat hackers alike. However, it is more popular in the white hat realm as it was designed for professional penetration testers and it allows them to scan servers or computers, uncover any potential security flaws, and then provide solutions to patch the system. Essentially, it is an auditing tool that can provide a wealth of information about the vulnerabilities found in any given host. OpenVAS is really a collection of programs that work together to facilitate testing procedures that are cataloged in a massive database of listed exploits – much like the Metasploit database. However, this program can be used for good or evil depending on the motivations of its wielder.

Installing OpenVAS

You have the option of installing OpenVAS on a server – which is usually what's done in the corporate world – or you can simply install it in the virtual VMWare environment that you had setup earlier. If you are going to be using this software within Linux, this will be the perfect opportunity to further familiarize yourself with the Linux command prompt. However, know that a virtual appliance exists that you can install as its own independent VMWare machine. In this example, we are going to be installing OpenVAS within Ubuntu Linux since it is a favorite for Linux newbies.

There are a couple prerequisites for this software as you likely don't already have it installed on your system. To begin, you will need to install the **python-software-properties** tools. Furthermore, you will want to run an update command to make sure that none of its dependencies are out of date. To begin, run the following two commands:

-sudo apt-get update
-sudo apt-get install python-software-properties

Now you will want to install the actual OpenVAS software from the Internet by using the following terminal command:

```
-sudo add-apt-repository ppa:openvas/openvas6
```

Though these commands may look a little hairy, they are just downloading and installing the necessary software. To put it simply, this is how you would install and configure the software from the command line. Next on the list, we will

need to rebuild a portion of the OpenVAS software as follows:

-sudo add-apt-repository ppa:openvas/openvas6

And now we will need to install the OpenVAS software by using the following commands:

-sudo apt-get update

-sudo apt-get install openvas-manager openvas-scanner openvas-administrator openvas-cli greenbone-security-assistant sqlite3 xsltproc texlive-latex-base texlive-latex-extra texlive-latex-recommended htmldoc alien rpm nsis fakeroot

Now that we have finished downloading and installing the software, we will need to proceed by configuring it before we can start scanning hosts for vulnerabilities. Though that process may have seemed difficult if you are new to Linux, it was actually very automated. By entering in a few commands, Linux will do all the downloading and installation procedure for you by itself. Compare this to a GUI environment where you need to browse the web to find software, download it, run through the installation procedure, and reboot your machine before you can use your new program. The real value in Linux for hackers comes from the power of the command line because it is lightweight (it doesn’t consume large amounts of CPU and memory as a GUI application would), extremely powerful, and contains ways to manipulate data that GUI versions of software simply don’t allow. Regardless, we do need to enter a few more commands to complete the OpenVAS setup.

First we are going to want to create SSL certificates. An SSL certificate is a small file hosted on a server that provides a cryptographic key that matches and identifies a unique organization. Also, it allows for secure data transmissions on port 443. We are going to want to go through some steps to configure the

web interface in case you want to actually install this software on a server for penetration demos. If you are setting this up in a Linux environment within a virtual machine, it will still give you another notch on your geek belt by learning a little bit more about the command line. Begin with the following command:

-sudo openvas-mkcert

Now you are going to see a myriad of options in the terminal to allow you to configure your certificate. If you wish, you can simply leave the settings at their default values, but it is often better to customize them for personal use. This is up to your discretion since these values don’t have a large impact on our configuration. But now you are going to need to make a client certificate for a user as follows.

-sudo openvas-mkcert-client -n om -i

To proceed, we will need to build and update the OpenVAS database to make sure it contains the latest vulnerabilities. If we don’t, it could easily miss exploit opportunities when we scan individual hosts. Run the following three commands in order:

-sudo openvas-nvt-sync

-sudo service openvas-manager stop

-sudo service openvas-scanner stop

The next portion of the configuration can take a while to complete, so be patient. We need to configure the scanner component of the software and it will have a lot of data to download and sync. Use the following two commands:

-sudo openvassd

-sudo openvasmd --rebuild

For our next step, we will want to proceed by downloading the SCAP protocol (Security Content Automation Protocol) which is simply another component of the background services that will identify weaknesses in target hosts. Again, this particular command can take quite a while to complete so you will need to play the role of a babysitter as the software does its thing. Use the following two command:

-sudo openvas-scapdata-sync

-sudo openvas-certdata-sync

Sometimes the second command listed above will fail and throw the error that there is no such table found in the software configuration. I you have encountered this problem, your operating system doesn't have all of the dependencies for OpenVAS updated to their latest version. The good news is that we can install them with a couple of easy commands.

-wget **http://www6.atomicorp.com/channels/atomic/fedora/18/i386/RPMS/openvamanager-4.0.2-11.fc18.art.i686.rpm**

-rpm2cpio openvas* | cpio -div

Now run the following commands to make OpenVAS use all of the files from a central directory. This will improve the speed and efficiency of the OpenVAS software.

-sudo mkdir *usr*share/openvas/cert

-sudo cp .*usr*share/openvas/cert/* *usr*share/openvas/cert

Now your dependency problems should vanish and you should be able to successfully sync the data. Run the following two commands:

-sudo openvas-certdata-sync

-rm -rf */openvas /usr ~/etc**

User and Port

Configuration

As we near the end of the setup and configuration process, I wanted to show you another example of a port. In the network fundamentals section I had shown you the basic idea of users and ports, and now we have the opportunity to catch another glimpse of that information in action as we configure OpenVAS. To start we will need to configure a user account with the following command:

-**sudo openvasad -c add_user -n admin -r Admin**

This command will create a user account with full and unrestricted administrator privileges. The username will be ‘admin’ and the password will be of your own choosing. Now we need to configure what host or hosts can access the software. If you are installing OpenVAS in a virtual Linux environment, the default will suffice because it only allows access from the local machine. However, in corporate environments or home environments where you want to install OpenVAS on a server, you will need to change the default configuration so it will allow access to remote users. If you are using your own virtual Linux environment, you can skip this step. To change this setting, issue the following command to open the configuration file in a text editor:

-sudo nano *etc*default/greenbone-security-assistant

At the top of this file you will notice a line that indicates which address(es) are allowed access to the OpenVAS software. By default, it is set to the loopback address (meaning the local host) with the address of 127.0.0.1. You can allow access to any host you want, but it is best to set this value to your local subnet’s address. For example, if you use the defaults on your wireless router your

network is likely 192.168.1.0/24.

Now that we have all the tedium out of the way, we can start the software and start scanning hosts. The most difficult part of getting your feet wet with OpenVAS is the installation process, as all it takes to scan a host is an IP address and the click of a button. First we will need to kill the currently running OpenVAS processes and restart the services. So, let’s finally fire up this amazing vulnerability scanning tool with the following commands:

-sudo killall openvassd

-sudo service openvas-scanner start

-sudo service openvas-manager start

-sudo service openvas-administrator restart

-sudo service greenbone-security-assistant restart

Running the Software and Scanning Hosts for Vulnerabilities

Once the services have been restarted you should be able to login to the web interface. Whether you are using a remote server or a local machine, you are going to need to use the following URL syntax in a web browser to reach the login prompt:

-**https://server domain or IP address:9392**

You will likely be presented with a certificate warning, but this is ok. Ignore the warning and proceed to the login screen. Next, enter the username and password you had configured earlier to login. After you have logged in, you will see a prompt for the default scanning wizard. All you need to do now is point your OpenVAS vulnerability cannon at an IP address and you will be able to find any current flaws or exploits contained within that host. So, enter an IP address and click ‘Start Scan’ to see a report of security vulnerabilities.

In most real world scenarios, an attacker would most likely use NMAP combined with Metasploit to hack around a network and look for weak points. However, OpenVAS is a great tool for newbies because it is so simple to use after it has been installed. All you need is an IP address and the click of a mouse to see detailed information regarding vulnerabilities found in any host you scan. Furthermore, the scanning software ranks the criticality of different vulnerabilities so you will know which ones will cause more damage if they are exploited. When you click on the magnifying glass on each vulnerability, you will be able to see greater details regarding the flaw and even ways to patch that vulnerability.

Keep in mind that the flaws and vulnerabilities found on scanned targets is always being updated via the database, so they change as time progresses. That makes the exploits you find very temporal. For example, if a new vulnerability is found next week and added to the OpenVAS database, you can rest assured that you have

information regarding the most cutting-edge exploit trends. On the flip side, older vulnerabilities that are no longer valid will be removed from the software.

Though each vulnerability and exploit is truly its own animal, you can look for information in Metasploit that would help you take advantage of the vulnerability. Metasploit is also continually updated, and it is likely that you will be able to find and execute a payload or exploit after you have discovered it with OpenVAS.

Chapter 13 – Social Engineering

While you may have erroneously thought that the only way hackers steal passwords is by entering cryptic commands into a text based operating system like you see in the movies, there are some much simpler techniques hackers use regularly to steal people's information. Social engineering is a technique frequently used by sophisticated hackers to gain access to networks, and you need to have a solid understanding of these techniques to protect yourself from their black hat endeavors.

Let's start by defining the term social engineering. Basically, it is a way for hackers to manipulate targets into unknowingly forfeiting their information. Most typically this information is account data such as usernames and passwords that a black hat hacker covets to gain access to a computing system or network. Once they have a point of entry to the network, then they will proceed with reconnaissance techniques and scanning procedures. However, sometimes hackers employ social engineering to acquire banking credentials or local computer credentials in order to install a virus or Trojan. The point is that social engineering is typically one of the first steps an attacker takes to carry out a grander scheme.

And guess what? It's one heck of a lot easier for a hacker to trick someone into giving up their information than it is to hack into their computers and take it by force. Part of this is just due to psychology. You'll find that people are always quick to guard the personal information and question where their personal data goes when they enter it online, but when talking with a real-life human being they are a lot more lax. Sure, you may have misgivings about giving your Social Security Number to a stranger over the phone, but consider a short scenario. Let's say you are an accountant working in a medium-sized firm and you simply don't know everyone who works at your company personally. One day you get a call explaining that there were some network issues yesterday and every account

needs to be reset (or some other believable yet bogus excuse) or your account will get locked out of the corporate network resources. If the social engineer did a good job of impersonating someone from your firm's IT department, chances are you would give them your username and password.

That brings us to one of the most fundamental aspects of security. You simply need to know who to trust and what online resources to trust. There's an old adage that will ensure that you never misplace your trust again: trust, but verify! You have no idea whether or not that person on the phone is legitimate. The biggest challenge large organizations face with social engineering is the trust factor, because their entire network could be compromised by one individual who just takes everything at face value.

Take physical security and defense as an analogy. It doesn't matter how high your castle walls are, how many troops you have deployed, how large your spear infantry is, or how strong your mounted cavalry units are; it only takes one idiot to see a wooden horse as a wooden horse and the next thing you know your empire has crumbled. On a side note, I would probably say that the modern equivalent example of a Trojan horse is a burglar who pretends to be a pizza man, but I think you see the point. Once a hacker gathers critical information with social engineering, an entire business network could easily be in jeopardy.

Types of Social Engineering Attacks

There are several common attack methods that criminals and hackers love to use for social engineering purposes because they have a high success rate. You'd think the general public would have learned their lessons by now, but the ugly truth is that some people still fall victim to these types of attacks because they are naïve, gullible, or over trustworthy. The following are some of the most popular social engineering methods hackers love to use.

An Email from a Trusted Party

Don’t offer up your credentials to anyone, and I mean *anyone*, including your close friends. Unfortunately, hackers have been able to expand their access to a network after successfully hacking a computer by duping users on the attacked PC’s email list into forfeiting more information. By using an email account from the computer they hacked, the hacker is able to take advantage of the trust relationship between the person they are emailing and the person they have hacked.

But watch out! Attacker’s attempts to gather information are usually a lot more sophisticated than an email saying something to the effect of, “Hey Steve, can you give your username and password for www.example.com? I forgot my password.” Sometimes they will include a link to another site in an effort to employ a phishing attack. Other times they may send a toxic link to a resource they control that looks genuine, but they include a vague message such as, “Hey John, you gotta check this thing out!” Once you click on the bad link, a virus or some sort of malware could easily be downloaded to your computer.

Even more worrisome is an email that contains a link to a download. It could look like a content download such as music, video content, or pictures, but the download link will actually point to a malicious code download. After a successful attack, the hacker will be able to access your computer, email program, and other sensitive information. And now the attacker has a whole new email address book to use to facilitate further attacks, and the vicious cycle repeats itself.

Be warned. Hackers love to manipulate and take advantage of the emotions of human beings by urgently asking for help that is needed immediately. Sometimes they will appeal to your good nature and ask you to make a charitable contribution to someone in need. Though it is heartbreaking to try to separate the wheat from the chaff and know if you are truly helping someone out, you need to protect yourself and not donate any money if you can’t verify the company and link as a reputable organization.

A False Request for Help

Sometimes hackers will send messages that appear to be from a legitimate company that claim they are responding to a request that you never made. Often they will imitate a large and reputable corporation with thousands upon thousands of users to increase their chance of success. If you never requested aid from them, you need to avoid that email like the plague. The real problem here is the scenario where you *do* use a product or service from the company they are imitating, though.

Even though you didn't originally ask for their help, you may still be enticed into wanting what they offer. For example, let's say that the hacker is impersonating a representative of a large bank and that there was a reporting error that caused the bank to make an error that needs to be verified. Because you want to make sure that your money is safe, you decide to trust this false representative. But here comes the catch. The hacker is going to claim that they need to first "authenticate your information" to see if your account was affected by the "error." You give them your credentials, and the next thing you know you have been robbed blind.

Other times a hacker or bottom-feeding Internet huckster will try to class up a false claim that seems believable in order to take your money. These emails almost always employ urgency to motivate their targets to take action. My perception of these attempts is that they are nothing short of unadulterated knee-slapping gut-busting laugh-until-you-pass-out hilarity. But the sad truth is that they work, and some people mistakenly place trust in a stranger they have never met before. To illustrate these types of attacks, let's turn to the iconic Nigerian Prince scam.

This scam was in full swing during the 80's and the early 90's, but there have been many other copycat hucksters that created their own variations of the scam. In its infancy, the scam was actually sent through the public mail system. However, at the time email was an emerging trend and since it was all the rage,

it only follows naturally that these scams started finding their way into email inboxes. In the classic Nigerian Prince scam, an impersonator of a high-ranking Nigerian official (sometimes a businessman, other times members of the royal family) would send an email claiming that he wished to send *millions* of dollars into the account of the target. But why would anyone want to give away that much money? The thin lie that so many people ate up like candy was that the money was reserved for a political budget but it was never actually spent. As a side note, have you ever heard of a politician that failed to spend their entire budget (and then some)? Of course not! But if you would be so kind as to help this Nigerian Prince, you would get to keep a quarter or a third of the total value of the bank transfer. In the end, a lot of poor, gullible, unfortunate souls became even poorer when they offered up their banking credentials.

Baiting Targets

Any baiting scheme is going to revolve around the appearance that the attacker is offering something of value. Many times you will see these types of social engineering attacks in pop-up ads or on torrent websites. The bait is frequently a free book, movie, or game that the target thinks is legitimate when in reality, it is a link to malicious code. Unfortunately, some of these offers look very real – they can take the form of a hot deal in a classified ad or a deal found in an Internet marketplace or false e-commerce site. These are hard to spot as scams because the attacker has found ways to manipulate the system to give themselves a favorable and trustworthy rating. Once you have been duped into following the link or download, the attacker has successfully injected a malicious program, virus, or malware onto your computer and has a foothold to carry out further attacks.

How to Protect Yourself from Social Engineering

Social engineering is a huge problem because it evolves with technology, and you can't always know whether or not someone is legitimate. Fortunately, there are a lot of things you can do to reduce the chance that you are victimized by an attacker using these techniques.

First of all, be sure to take your time and think about the consequences of your actions beforehand. Attacker would love it if you just reacted to a situation without thinking about what you are doing, but take a moment to think ahead – even if the message claims an urgent scenario.

Also make sure that you take time to verify and validate any information that looks odd or suspicious. Go through their claims with a fine tooth comb and remember to remain skeptical. Even if you get a message from a company you do business with, make sure the URL link matches the company's website *verbatim*. If they provide their phone number, you can do a reverse phone lookup on the Internet to cross-check their validity. Make sure that you *never* respond to an email that requests information such as your username or password. Reputable companies would never ask for your personal information in an email.

In addition, make certain that you never respond to false messages claiming to be a response for the help you never requested. Delete these before ever opening them because they could contain links to malware that would destroy your computer. The best way to combat bad links is to use legitimate means to find them. For example, don't follow the link in an email if you want to verify it. Instead, use a Google search because it extremely unlikely that an attacker with a face website has beaten legitimate websites in SEO endeavors to rise to the top of the search rankings.

Chapter 14 – Man-In-The-Middle Attacks

Man-in-the-middle attacks are extremely dangerous for end users because a successful attack will allow a hacker to view *all* of the data that a user is sending over the network. If the user is setting up a connection to a VPN server, the hacker will be able to capture their key to decipher their encrypted messages. In addition, the hacker will be able to see all of the websites the user visits as well as steal information such as usernames, passwords, and even payment card data.

An attacker performs this exploit by tricking the target's computer into thinking that the attacker's computer is the default gateway or intended destination for data transmissions. For example, let's say that you wanted to do a Google search. Normally, your data would be sent to your default gateway (e.g. your wireless router), routed through the public Internet, and then reach one of Google's servers. However, with a man-in-the-middle attack, your data would first be sent to a hacker somewhere in the middle of the process before reaching Google's servers.

These attacks are extremely problematic because it is very difficult to determine that your data is being sent to a hacker before it reaches the intended destination. Hackers know this, and their goal is to sit back quietly and discretely listen to all of the traffic you are sending without your knowledge.

Though there are many ways to initiate this type of attack, such as with a DNS attack that redirects information to a hacker's IP address, they are most frequently carried out with a process called ARP spoofing. If you remember, I had introduced you to the concept of ARP in chapter 5. If you don't remember, realize that ARP is the process that links a layer 2 address (MAC address) with a layer 3 address (IP address).

With ARP spoofing, the goal is to trick the target host into thinking that the hacker's MAC address is bound to the default gateway's IP address. That way the target will send any data that is not destined for a device on the local network to the hacker first. In turn, the hacker will then send the target's data to the default gateway and out to the public Internet.

While the basics of understanding a man-in-the-middle attack using ARP spoofing are rather basic and straightforward, ARP spoofing is only half of the battle. Once you have tricked a client into sending you their data, how do you see and read what they have sent? This brings us to the idea of tools called packet sniffers. A packet sniffer will be able to show you *all* of the data flowing over your computer's network interface card. The details of the information contained in the packet sniffer data are rather complex, but you can sort through all of the data using filters. One of the easiest packet sniffers to use is Wireshark on Windows, but Linux also contains some great packet sniffing programs that integrate with the terminal. You even have the ability to store and save all of the data you have collected from a target and you can sift through the information at your own leisure.

As this is an advanced topic, you likely won't understand all of the various protocols you see in the data collected from your packet sniffer. However, as a demo aimed at beginners, you can sort through the data by filtering results for port 80 (HTTP) which will show you the IP addresses of the web servers the target is connecting to. Basically, this will show you every website the victim visited as well as other information such as usernames and passwords.

Though some are sent in plain text and you can read them from your packet sniffer, many will be encrypted. Your packet sniffer can record these keys and then you can use other utilities to crack their passwords, but this is a little harder an impractical unless you want to become a black hat hacker. So, for those reasons, I will show you how to initiate a man-in-the-middle attack with ARP spoofing and how to use a packet sniffer to see what websites a target is

connecting to. Also, understand that packet sniffing on a wireless interface is a little different than sniffing on an Ethernet interface. For that reason, this demo will show you how to perform the attack on a wired Ethernet interface.

How to Perform a Man-In-The-Middle Attack

To start the attack, we first need to successfully spoof an ARP binding. To do so, we are going to use a tool on Kali Linux called 'arpspoof.' The syntax for this command is as follows:

-sudo arpspoof –i eth0 –t [**TARGET ADDRESS**] [**DEFAULT GATEWAY ADDRESS**]

So, if you wanted to trick a host on your local network with the address of 192.168.1.10 into thinking you were the default gateway, the command would look like this:

-sudo arpspoof –i eth0 –t 192.168.1.10 192.168.1.1

If you don't know your default gateway address, just use the **ipconfig** command in Windows or **ifconfig** in Linux. If you didn't know of any valid host IP addresses to target, you could simply issue a simple ping sweep using NMAP as we did in chapter 7. The command listed above will trick the 192.168.1.10 host into believing your computer's MAC address is associated with the default gateway's IP address of 192.168.1.1. At this point your terminal window will continually spit out lines of code ensuring that the spoofing process is succeeding, so you will need to open another terminal window to proceed with the attack.

But there's just one problem. You have only done half of the spoofing attack. At this point, your target thinks that you are the default gateway, but this isn't true in the reverse process. That is to say that the default gateway doesn't think you are the target host! So, in your new terminal window we are going to need to start another ARP spoofing procedure. The syntax will be the same, except the target and default gateway addresses will be swapped as follows:

-sudo arpspoof –i eth0 –t 192.168.1.1 192.168.1.10

At this point in the attack, you have fooled both the default gateway into thinking that you are the target host and you have fooled the target into thinking that you are the default gateway. Now all you need is for the target to transmit data and to inspect that data on your computer. There are some higher level tools that will actually capture the data you catch during the process instead of dumping it as raw data into a text file, but packet sniffers offer a wealth of information too. Remember to keep both of the previous terminal windows open as they are still constantly running the ARP spoofing process.

If you want to use a high level tool to see the data a target is searching for online that isn't too complex, you might be interested in *driftnet*. Driftnet is a tool that – while far from perfect – is a great way for newbies to try their hand at a man-in-the-middle attack and view data such as audio files, graphics, and MPEG4 images and automatically display them in the GUI. To use driftnet, which is packaged with Kali, run the following command:

-sudo driftnet –i eth0

If you are doing this demo in your home network environment (as I instructed you to do many times already), try running the driftnet command. Then do a quick Google image search on the target device. The attacking computer that sits in the middle should be able to see all of the images that the target device is viewing. Pretty neat, huh? The problem though is that people can abuse these types of attacks to get away with murder and steal some truly sensitive information. Again, I caution you not to use this technique outside of your own home because the consequences could be very severe!

Lastly, if you want to dig a little deeper with these types of attacks, you would want to use a packet sniffer and dig into the raw data that your attacking

computer is gathering. You can see a lot more than simple images, and once you dig into the transmission protocols you can find data such as login information, data a user has entered into fields on a web form, and just about every single thing they do online!

Chapter 15: Cracking Passwords

Though you might not think so at first, your email is actually one of the most dangerous accounts to lose to a hacker. The reason being that there is so much personal information stored in your inbox. Once an attacker has access to your email account, you're in for a world of hurt because they will be able to see and intercept all of the messages that reach your inbox. Worse yet is the idea that they now have a way to impersonate you. If they wanted to, an attacker could trick other people in your address book into forfeiting additional information by using your identity to request that information.

Furthermore, there is going to be a ton of sensitive data linked with your email account. Websites today are getting pretty complex, and there are a lot of ways to link a user's login credentials and web activity with their email address. For example, there will likely be emails and promotions from sites that you have already done business with sitting in your inbox or spam folder. This gives an attacker clues as to where he or she can look to uncover additional information. They may also be able to see what purchases you have made with online sites such as Amazon.

Password Cracking

While all of these scenarios are terrible, by far the worst advantage an attacker gains is the ability to further hack your passwords. There are several techniques an attacker can employ, but they all exist to steal your credentials to escalate their privileges. For example, who knows what an attacker might purchase if he or she had access to your Amazon account and payment card data?

Now that you have a basic understanding of how critical secure passwords are and the consequences of what an attacker can do once they get your password, let's look at the basics. I sure that cracking passwords sounds cool and really complicated, but some of the methods used are unbelievably simple and even a little anticlimactic.

As commonly mentioned throughout this book, don't try to hack someone else's passwords because the consequences can terrible if you get caught. Don't try to hack into a person's email and see how many of their accounts you can break into just for the hell of it; that would be a huge breach of privacy and I shudder to think what might happen if you get caught stealing someone's payment card data.

To be honest, it would be pretty difficult for a single user who doesn't have knowledge of information technology to discover how their account was hacked in the first place, but in a corporate or professional setting the I.T. department would have numerous tools to track electronic transactions and discern what IP address the attack or attempt was made from.

The first, and simplest technique for gaining a user's password assumes that you already have access to their email account. Most users typically only have 1 main email account that they use, but there could be several. Anyway, after you have obtained access to their email you can use the password recovery mechanisms built into most online account. While most people choose to cache their usernames in their browser so they don't need to reenter them every time

they login to a website, you don’t even need to know their username. You see, most websites provide an account recovery feature that allows a user to input their email address to receive their username and password.

Some sites require that the account recovery feature erases the old password and generates a new and random password, but all of this information is communicated via email. So, if an attacker controlled and user’s email account and wanted access to their bank’s website, Amazon account, social media accounts, or just about anything else, all the attacker has to do is browse to the given website and perform the steps necessary for account recovery. This is an extremely quick process, and in a matter of minutes an attacker could easily gain access to the most critical sites that the user visits.

While this may not be a sexy process, it sure gets the job done and can ruin an individual’s personal security. However, this is just the simplest measure to crack passwords and it presents a problem. How did you gain access to their email in the first place? There are countless other ways that an attacker can crack passwords to first gain access to the email account. For example, if a user isn’t very technically inclined, it is a safe bet that they don’t understand anything about password complexity. Though they think they are being clever, users are making a huge mistake when they make their passwords their birthday, the name of their dog, or other easy to guess pieces of information.

Other times, these simple minded users will actually write their passwords down near their computer or plaster a sticky not on their monitor. It is even possible to trick these people into forfeiting their email passwords with social engineering. All of these methods are easier to use than you might think, and it gives an attacker a foothold into the rest of their user accounts.

Password Cracking Utilities

There are many different password cracking utilities to take advantage of, but we are going to take a brief look at the most popular pieces of software. Hackers will employ several of these tools in conjunction with one another to facilitate their attacks. They simply don’t start with a brute force attack because passwords can often be found using quicker methods. With that said, a brute force attack is usually the last resort when other methods have already failed.

John the Ripper

John the Ripper is probably one of the most famous and revered password cracking utilities in hacker communities. It is highly efficient and effective, but it does suffer from one fatal flaw that often keeps it out of the hands and minds of newbies: it was developed for Linux. Though it does have ported versions, keep in mind that it is natively a Linux application.

Because some of these tools are exclusively built with Linux in mind, you will surely need to get your feet wet with the Linux operating system to become a competent hacker. By now you should have already setup a Linux environment to run through some of the demonstrations in this book using VMWare. If you haven't already, it is high time to build your first Linux environment.

As with most powerful Linux software, this program is run from the command line and can be a little scary if you aren't already used to working from the command line. But that's just part of the learning curve; once you get comfortable in this environment, you'll be able to run all kinds of software that is far more powerful than basic GUI software like you might find in a Windows environment. However, there is a version of this software on Mac devices since Macs derive from an old and powerful UNIX distribution called BSD.

One extremely handy feature of this software is the method with which it uses to crack passwords by automating the process. To start, it will begin with a dictionary based attack. If that fails, it will move on to use a hybrid approach to crack passwords. If even the hybrid approach fails as will, it will resort to a brute force attack.

Ophcrack

Ophcrack is the first of the password cracking tools we will discuss, and like many of these tool, it is free to download and use. It can be used to crack passwords on a variety of operating systems, but this tool has gained most favor from hackers that are attempting to crack Windows passwords. However, it can still be used to facilitate attacks on Linux and Mac passwords. Though it does have simpler and more effective algorithms, this piece of software will allow a user to perform a brute force attack. Lastly, it even has a feature that will allow you to create a live boot image.

L0phtcrack

L0phtcrack is really a suite of software that allows you to perform many different password functions. For example, it can be used to audit password strength and complexity to bolster your security efforts. Given the range of functions this software provides, it is frequently used with computer security firms as well as governmental organizations such as military applications. Not only can it run on versions of Windows that are higher than Windows XP, it can also run on some Linux and BSD distributions. Like other password cracking utilities, it will allow an attacker or security expert to run both dictionary based attacks and brute force attacks.

Cain & Abel

Cain & Abel is another popular password cracking utility. Its features exceed only the ability to crack basic passwords or operating system passwords, and it even has some features that aid in the process of wireless security-key cracking. However, it can only be used exclusively in a Windows environment and it allows users to crack passwords that have been encrypted and encoded in various formats and protocols such as MySQL, Oracle, MD5, SHA1, SHA2, and various wireless encryption algorithms.

As with the other utilities, this software will perform a variety of different password cracking methods such as dictionary attacks, rainbow attacks, and brute force attacks. One extremely useful feature of this software is that you can set parameters to fine-tune the brute force attack such as the length of the password you are trying to crack. This has the ability to eliminate millions of potential password combinations that would otherwise drastically multiply the length of time needed to carry out the attack.

In Summary

These tools aren’t incredibly difficult to use, but most users don’t have any clue that they exist. Really, all of the hard work has been done already by the expert programmers who created this software. All that’s left to do is for it to be used by an experienced hacker. Tools like these are so easy to use that teenagers with little experience in the real world can find ways to use them to hack into other people’s computers. Though I wouldn’t recommend using these tools for evil, they are certainly fun to use in a home environment.

Chapter 16 – Protecting Yourself from Hackers

At this point in the book you have probably already asked yourself at least once, “What can I do to protect myself from hackers?” The good news is that there are a lot of easy and simple measures you can take that will drastically reduce your chance of being hacked by a nefarious black hat hacker on the Internet. This chapter focuses on the different strategies you can use to make your computer and home network more secure. For those of you who are very technologically savvy, a few of these might seem like no-brainers. However, you would be surprised how many people fail to implement even the simplest measures regarding their Internet security.

Software Updates

Software updates are crucial to protecting yourself from hackers, but too many people ignore updates. Most operating systems have an automatic update setting that will automatically download and install patches to the operating system. The problem is that most people are apathetic or just plain lazy and they don't want to take the time to install the updates. And why not? To be honest, it's a bit of an inconvenience to some people. You might be right in the middle of a large project or your work day, and installing updates requires that you reboot your computer and wait for an unknown amount of time while the operating system install the patches. But I've got news for you – you need to take great care to install updates as soon as humanly possible.

Even after some of the viruses mentioned in chapter 3 were discovered and patched, there were still *millions* of computers that were still contained vulnerabilities all because the users failed to update their software. If everyone had installed the updates as they came out, the viruses would have been stopped dead in their tracks.

Change Default Usernames and Passwords

Too many people don’t think twice about changing the default usernames and passwords on their networking equipment. While most people try to create unique usernames and passwords for their personal computers, they often forget to secure network devices, wireless routers, and even their printers. Wake up people, hackers not only have ways to perform password attacks but they already know how to find the default usernames and passwords to your wireless router in a matter of seconds.

Furthermore, some people fail to secure their Wi-Fi network. Instead of using a security algorithm that will make it hard for attackers to join their network subnet, they give them an open door and invite them to come inside. Some, but not all, wireless routers don’t include a default wireless password.

Worse yet, when people are initially configuring their wireless routers, they fail to add a password to their Wi-Fi. You simply can’t leave these values at their defaults if you hope to protect yourself from online attacks. Lastly, most wireless home routers have an option in the configuration that determines who can remotely manage the device. If you lock down this setting to a specific IP address, hackers won’t be able to log into your wireless router even if they know the username and password!

Use Strong Passwords

Not only should you create unique usernames and passwords for your devices that are different from the default values, but you should also make your passwords strong. You can do this by making them as long as possible and by including numbers, letters, and special characters. Though it's true that hackers have ways to perform dictionary and brute force attacks whereby they try to go through every possible combination to find the correct password for a system, know that these techniques don't work in every situation. Some websites and networking devices have built-in protection against brute force attacks that don't allow you to attempt to login for a certain time period after a specified number of failed login attempts. Password security is a huge area of study, and most hackers know what types of data users incorporate into their passwords to remember them easier. So don't make your street address, family pet's name, or birthdays part of your passwords.

Oh, and don't be one of those jokers that has their password written on a sticky note that is attached to your monitor. A hacker implementing social engineering wouldn't even have to try. You're making it too easy for them by displaying your passwords for all the world to see. In addition, make sure that you don't store your passwords in plain text files or other types of files that aren't encrypted. If a hacker does steal some of your data and they get their hands on a file that contains usernames and passwords to other sites and services, you're in for a world of hurt.

Properly Configure Your Firewalls

Firewalls are a critical part of any security solution designed to protect users from hackers, and you need to make sure that your firewall is configured correctly. In the past, I have seen some people struggle with opening the right ports to get their software configured correctly. One area this happens a lot is with gaming.

Many games need specific ports opened that aren't well known, and in a fit of madness and frustration, users choose to open all the ports on their firewall to make their game work correctly. This is a colossal mistake, because it will allow hackers to penetrate your network firewall if none of the ports are blocked. If you have problems getting a game to work on your home network, just do a quick google search to see which port needs to be opened!

Furthermore, many people fail to take advantage of software firewalls. While many hardware firewalls have most of the ports blocked by default and do a good job of protecting a local area network, but few people protect themselves with a firewall on their host computer. If you are a Windows user, whether you know it or not you already have a software firewall that will add an extra layer of protection between you and black hat hackers. Though sometimes it is appropriate to disable your software firewall to allow a program to function correctly, you *always* need to remember to re-enable it after you have finished your work.

Antivirus and Antimalware Software Solutions

If you do get hacked and a hacker manages to hack your system with a virus or a Trojan, how will you know it exists without antivirus and antimalware software? Using a computer without security software is like begging for an attacker to steal your personal information.

But it doesn’t stop there. It has been said many times before, but understand that torrents are frequently used as a distribution system for viruses. Too many people have fallen victim to a hacker’s virus because they wanted to watch some video content without paying for it. If you download torrents without antivirus software, you’re just asking for trouble. If you do have antivirus software, you can scan the files you download before opening them to detect any potential malicious code embedded in your download and avoid a computing crisis. For that matter, you should scan every download before you open it. You never know what could be hiding in an innocent-looking file.

Using VPNs

If you aren't aware of VPN tunnels, you need to know the immense value they bring to the table. A VPN (Virtual Private Network) is essentially a service that encrypts *all* data communications between two endpoints – effectively making it impossible for a hacker, governmental agency, or petty Internet crook to unscramble and decipher the data. This guide isn't promotional material for VPN providers, but the fact of the matter is that they can prevent you from getting hacked. Not only that, but they can stop the government from stealing your data. As a result of the information leaked by Edward Snowden, the US government and the N.S.A. have been found to be capturing emails, photos, telephone calls, instant messages, and many other types of data transmissions in an effort to prevent terrorist-related activities. However, the N.S.A. has stated that they haven't found any information that has stopped even one terrorist-related event. By encrypting your data, you will make it safe from hackers around the world while it is in transit through the public Internet.

Backing Up Your Data

You might think that backing up your data is only a measure to protect yourself from hardware failure. While it does certainly help you out a ton if your computer fries, you should know that using backup software will protect you from black hat attacks as well. Some of the more sophisticated attacks damage and corrupt files, or even embed malicious code into common everyday files such as word documents. By keeping a backup copy, you can rest assured that you will have a clean and virus free copy of your data in the event of an attack. Remember the Crypto Locker virus in chapter 3? If only the users had backed up their data, they wouldn't have had to worry about paying an Internet huckster loads of money to reclaim their data by means of ransom.

Web Browser Security

There are also a lot of things you change in your web browser that will drastically reduce the chance of a successful attack. As we discussed earlier, hackers can use malicious scripts to steal cookie and web browser data to steal the passwords to various sites.

Make sure you don’t save and cache all of your username and password information in your web browser when visiting your favorite sites on the Internet. This is a huge No-No, because you are leaving low-hanging fruit ripe for the picking within the grasp of black hat hackers and Internet thieves. You’re also a lot better off if you disable cookies in the first place. By disabling cookies, you can circumvent a whole range of different online attacks and nip them in the bud before they become a real problem.

It’s best to keep your web browser as light and streamlined as possible, and the more data you save in your browser the greater the chance that someone will be able to steal your information. Also consider that you should frequently clear your history as well. This provides a veritable audit trail, and an attacker could use this information to see every website you have visited on the Internet.

Final Thoughts

I want to make sure you understand that no code will ever be 100% infallible. Computers are created and manufactured by humans who are anything but perfect, and mistakes are always made. That is to say you run the risk of being attacked every time you fire up your computer and open your web browser – regardless of whether or not you have implemented these security practices.

In fact, they say that the most secure computing system is one that doesn't have the ability to connect to the Internet at all. However, implementing these security measures will make it much more difficult for an attacker to successfully compromise your computer. Think of using these security practices in the same light as risk aversion. For example, if someone is a vegetarian their whole life and they abstain from alcohol and smoking, the chance that they will develop a chronic or life-threatening disease is slim to none.

Though it is still possible, their lifestyle choices severely reduce their risk of disease. Likewise, implementing these security procedures works in much the same way. The ugly truth is that operating systems and websites contain flaws and errors that can be exploited by hackers. It's just a fact of life. But by strengthening your security, you make it much more difficult – if not impossible in some cases – for an attacker to successfully hack into your computer.