MALWARE: ATTACK, DETECTION, AND PREVENTION

Grant Arnold
Florida State University
gwa21a@my.fsu.edu

Mehmet Ozmen
Florida State University
mco20m@my.fsu.edu

Jackson Pence
Florida State University
jhp18c@my.fsu.edu

Andre Davis
Florida State University
add20br@my.fsu.edu

Topic

The topic for this project is for the next six weeks Team B will act as a blue team in the cybersecurity field. Team B will actively configure, and implement defenses to their team web servers in order to prepare for any kind of attack that is bestowed upon it. For the last three weeks, all three teams in the class then act as red-team (Penetration Testers) in order to get through into another team's system. Later in this paper, during the final three weeks of the semester, we will discuss the methods that Team B will implement and configure in order to keep their own web servers and data safe from the other teams, as well as give insight into the penetration techniques we will use in order to penetrate into these other teams systems.

Abstract

The purpose of this term paper project is to set up, configure, and monitor different attackable devices such as Apache 2.4 Web servers, Palo Alto Networks Firewalls, HoneyBOT, tools in Kali Linux, and SecurityOnion in order to keep our assets and information protected. Using the information learned in the 4777 Information Security class as well as new information being studied this semester, we should be able to form a very secure network of our own. All of these tools will be essential in making sure that the web servers that we will be protecting over the semester stay unexposed and remain impenetrable for the entirety of the semester.

Keywords: Active cyber defense, apache 2.4 web server, Kali, Linux, firewall, Palo Alto, a network security monitor, forensics, data recovery,

Introduction

Over the course of the semester, Team B has been building its own network segment and arranging the systems in order to be defended from attacks that they may find themselves becoming targets of. Initially, Team B was given an array of team Virtual Machines, as all four members of the team got their own Honey Machine (Windows), Web Machine (Apache Server), Kali-Linux Machine, Ubuntu Machine, and a CAINE Machine. Every team was also given their own Palo Alto Networks, Pfsense, and SecurityOnion services as well.

In order to prepare for any attack that could potentially find Team B as a target, we employed the use of all of these VMs to ensure that they were not able to be sniffed or broken into. Of course, the initial procedures such as changing all of the default passwords and giving the machines our own created passwords were completed, but knowing this was likely not enough to be able fully to defend against threats, a defense plan was adopted by the team and implemented onto our system machines. We would create a DMZ for the windows and web-server virtual machines to sit behind, place them behind the firewall, and manage access in and out of the network segment that we created. After we successfully placed all of our systems behind the firewall, we could then focus on the attack that our team was to carry out. Because of Team B's focus on malware, we will be using Kali Linux in order to perform our attacks on the other team's systems using software such as Metasploit, which is a software that is used to create and deliver malicious payloads to other computers and report our findings in the later parts of this paper.

Literature Review

Malware within Networks

Malware is one of the biggest threats that arrive in the real world for information systems across the world. This is because of the fact that it so easily can find its way through a network and begin infecting device after device while sometimes remaining incredibly undetected.

Every single day, more than 350,000 different kinds of malware are found that have the potential to rip through networks and devastate any computer that it manages to infect. (HP, 2020) This only slightly shows us how big of an issue this is in today's ever-growing online world, and therefore there must be measures put into place to make sure that users can still be safe.

This can be easier said than done, as malware adapts and shows up in different forms all of the time, and as shown by the numbers above, there is no way to be protected from all malware all of the time, as there are surely malicious codes being executed on machines that have never been seen before by any anti-malware application that is out there.

Malware and Infection

Malware is further dangerous because of how easy it is to trip up and accidentally find yourself downloading and infecting your own computer with malware. This particular scenario most often comes as a result of unintentionally downloading a file containing a "trojan". Trojans are malicious files or code implanted within other more legitimate programs that, once downloaded, activate and infect computers and other systems with their malware payload. For this reason, many browsers pre-scan content that is being downloaded from the internet for trojans. These cursory scans are not usually as in-depth and far-reaching as other security methods, but it provides a kind of "first-line" when it comes to detecting and defeating trojans.

If a trojan manages to get past this first scan, then many anti-malware programs and antivirus security programs automatically scan newly downloaded files to ensure that they are legitimate. These programs typically rely on identifying specific code segments or fragments that align with a regularly updated database of known malware to detect infected files. However, due to many security programs relying on this database, it allows malicious hackers and other adversaries the opportunity to defeat them by creating entirely unique code for their malware.

Malware can quickly spread through connected computers that can greatly impact an organization's technical infrastructure. A lone employee can mistakenly (or purposefully) download an infected file from online and infect their computer with the malware. From there, the malware can transfer itself either through email or some other means to other computers on the network which will then replicate the process. The result is an outbreak of a certain kind of malware on several machines at once.

This is exactly how many companies have found themselves dealing with outbreaks of malware infections on their corporate or company systems. Either by their own employees somehow evading the installed (or uninstalled) anti-malware application without even knowing. Malware can come in the form of Phishing emails, URLs, or other downloadable documents that come from the internet. These tactics often include impersonating someone of a higher power or status and commanding them to download or execute a file that is on their screen. (Federal Trade Commission, 2019)

Zero-Day Malware & Exploits

Zero-Day malware, or malware that has not been detected or recognized by any kind of anti-malware application before, can be extraordinarily devastating to computer systems and networks, especially inside of a workplace environment. This kind of malware that is brilliantly crafted to go undetected can find itself going

through computer systems undetected for a very long time, gathering information that will likely be used for malicious intent right out from under the noses of computer users.

This kind of malware is extremely dangerous because of the fact that there is no way to get rid of the software without manually going into the source code of the malware, analyzing the contents, and then confirming that the inspected code is actually malware. The chances of a regular person finding the location of that specific malicious file (saying it hasn't copied itself somewhere different entirely in the meantime).

When speaking on the capacity of damage that Zero-day malware and exploits can cause, the computer worm StuxNet cannot be without mention. This zero-day malware was a computer worm that found its way into a Programmable Logic Controller (PLC), which was operated by Windows. When the PLC was overridden by the StuxNet Worm, code was then injected into the PLC, making the controller perform commands that were benign to what they were originally intended to do, causing them to or giving the potential to damage and destroy the surrounding machinery and hardware. Targets of this StuxNet Worm included Iran's Nuclear Uranium plant, and stretched its reign of terror from Iran, to India, to Indonesia. (Lynch,2020)

Companies must be very wary of Zero-Day malware because of the huge potential security threat that it poses. For example, if a Zero-Day exploit is deployed that can infect all Windows OS machines, then an entire company's network can be at risk until a solution is found and patched as soon as possible. Security teams must always be ready for this kind of situation and practice speedy troubleshooting in order to defend against these exploits when they arise.

Zero-day malware's attack choices are varied. Some of them are operating systems, web browsers, office applications, open-source components, firmware, internet of things. To prevent from services losing confidentiality, integrity, and availability, there are a couple of preventions that could be taken. Keeping all software and firmware as well as operating systems updated at all times. Uninstall unwanted applications, remove applications that are not used, and only use essential applications. More applications there are, and more vulnerabilities exist. Using a firewall also protects from potential attacks. Since zero-day malware is malware that is unknown to exist, it is difficult to detect and defend against. For defending against zero-day malware attacks rather than prevention is to have backups ready and in the event of an attack, have your emergency response team ready.

Kali-Linux Operating System

Kali Linux, released in 2013, known previously as BackTrack Linux, is an operating system based on Pen-Testing, Cyber-Forensics, and Reverse Engineering. The operating system is a Debian-based Linux system directly made in order to carry out the tasks that penetration testers and security auditors need to do. The operating system is also multilingual and allows users who do not speak English the best possible experience that they can possibly give to them. The operating system is available on a number of different platforms from desktop, mobile, and even raspberry pie. (Kali Linux, 2022)

Further, Kali Linux contains over 600 different tools for penetration testing, and also maintains the statement that the operating system is and always will be free of charge. These tools vary from password crackers like Jon the Ripper, packet capture tools such as Wireshark, and even penetration testing applications like Metasploit. These are just a few of the few hundred applications that come preinstalled on Kali Linux.

The operating system and all of its applications are part of a cyber-security project that is dedicated to finding and dealing with exploitations that are found in systems. These applications are also all developed in a secure environment so that the user will never have to worry about whether or not the software that they are using to perform their penetration tests is not malicious in and of itself. This in and of itself makes Kali Linux a very great option for individuals interested in penetration testing, as much of the software that can be found on the internet to assist with anything cyber security-related can be riddled with viruses and other exploitations.

In order to keep up with technology being constantly patched and updated with different security measures, Kali Linux also updates its kernel in order to manage the applications that will no longer work, have been patched, or further investigate vulnerabilities that have been discovered. This allows users to not have to constantly have to go through themselves and remove or update all of the programs and applications that are either no longer functioning or need updating, removing a large headache for the users of the operating system.

Also, to allow users the most customizable and unique experience possible, Kali Linux also uses a fully custom kernel with access to an open-source git-tree. This allows users from all over the world to have the source code and make their own changes to it in order to arrange or fix things with the kernel that they may see fit. This is something that is not doable in any way in windows, where users are restrained to what the operating system defines it as when it is downloaded. Of course, there are upsides and downsides to customizability.

An upside is that it allows the user to be able to have the exact set-up that they want in any aspect of the service, adding or removing things that the user may or may not want there. However, the potential downside to customizability is that it can make the surrounding framework and kernel become unstable, and it may lose functionality in some ways, some minimal and some potential computer killing. However, it is always better to have the option than to not have the option to customize.

Overall, Kali-Linux is the best tool to use for those wishing to do any kind of penetration testing or cyber forensics and will give any of its user's access to all of the newest and updated tools for penetration testing. Due to the operating system being completely open-sourced, its content is always being modified and updated in order to make for the best and most unique user experience, without the user having to worry about having a bunch of software and applications that are out of date and unusable or need patching. This being said, Team B will use this operating system in order to deliver our attacks after all teams are done setting up their systems. We will use applications such as Metasploit specifically, which is designed to craft certain payloads to create malware, in order to carry out these attacks. Using these tools will not only familiarize us with the operating system and how it works but also allow us to see how these exploits and vulnerabilities with systems are compromised in real-time.

Lab Settings and Description

Initial Class and Team Environment

The environment that we were given to test our defensive and offensive measures in this class was a Hyper-V Virtual Machine environment. This included Virtual Machines (VMs) that function in the same way that normal computers do by using a disk image of a particular Operating System (OS). These machines connected us to the SECNET environment, which is a segment of FSU's main network, and all three participating teams in this test were connected to this environment.

For each team, every member received a set of VMs: One for each sector of needed material in the classroom environment: A HoneyPot VM, An Apache2 Web Server VM, a Kali Linux VM, and an Ubuntu VM. Additionally, each team was given one Security Onion VM, one PfSense VM, and a Zeek VM. Unfortunately, due to an over-exhausted network capacity, the Zeek VM was not required for this assignment, and this team did not incorporate it into our practices.

Although these VMs were all separate, they all resided on the same network as each other, minus a few settings from the PfSense VM that required a bit of IP change when configuring interface cards. Because of this, our team had to split the network between being put behind the Palo Alto Networks Firewall that was provided to each team and is configured to be put behind the PfSense that was provided to our team. This can be seen in ***Figure 1: Initial Network Topology***

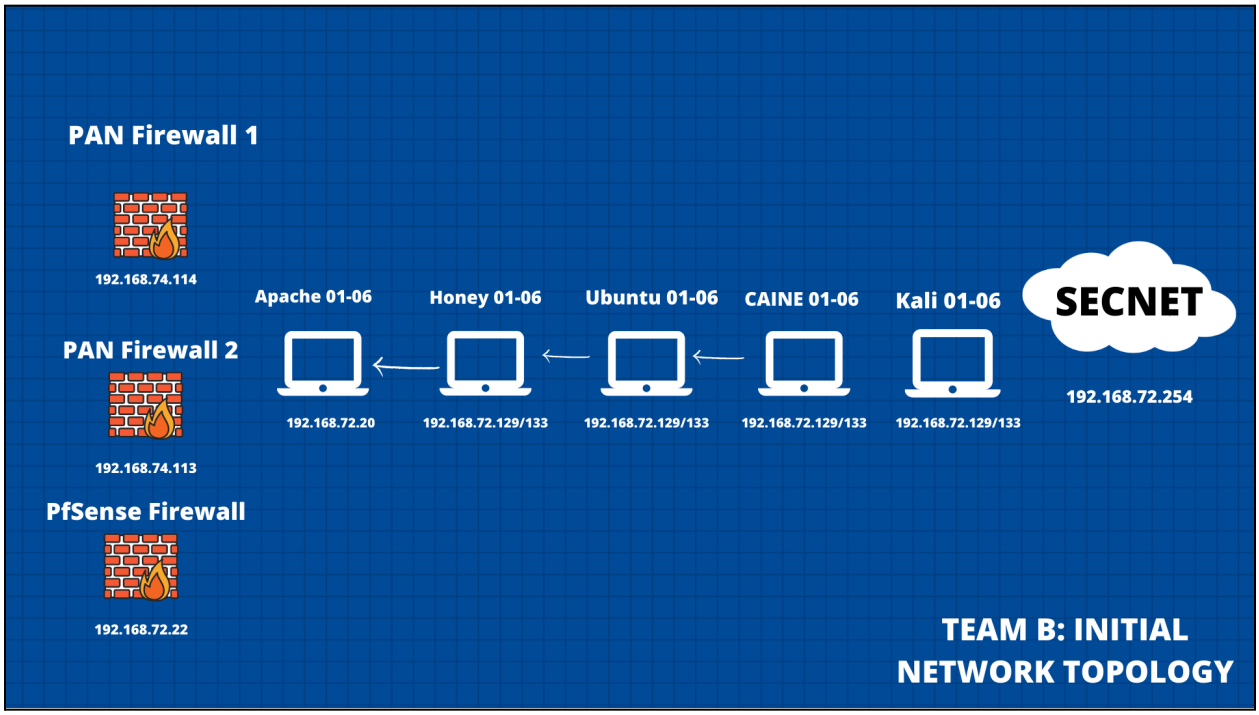

Figure 1: Initial Network Topology

Post Configured Team B Environment

Initially, as shown in ***Figure 1***, the Network topology for team B consisted of the teams SecurityOnion, Kali VM's, HoneyBOT VM's, Web Server VM's, CAINE VM's, and Ubuntu VM's. These were all floating on the same network and gateways as each other. This allowed outside intruders to be allowed to easily compromise all of the devices that we are trying to protect at once, as they were also on the same exact network. Because of this, different arrangements had to be made in order to utilize the team's resources to their full advantage, thus resulting in ***Figure 2***: ***Post-Configured Network Topography***

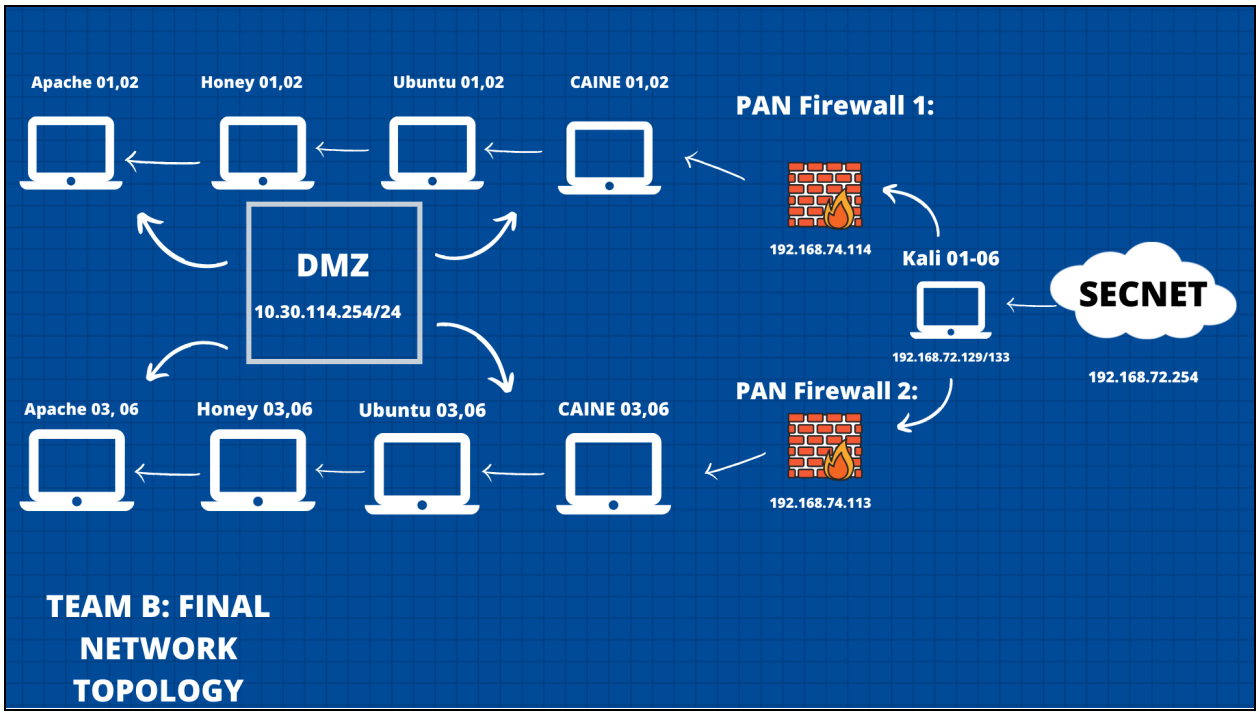

Figure 2: Post-Configured Network Topology

Securing the Environment with Palo Alto Firewall

To fully secure the network of our team's assets, we decided to fully utilize our firewalls in order to protect our servers. This meant that all of the virtual machines in our network segment were put behind the team's Palo Alto Networks Firewall into the Demilitarized zone created by our team: **10.30.114.1/24**.

This new network topology will ensure that the team maintains a very safe and secure environment that will prove difficult to penetrate into. When the assets are placed behind the firewalls, it then becomes very difficult for any unrecognized traffic to begin getting into the environment.

As well as securing the virtual machines in a different network segment, the Palo Alto software that was deployed also acts as an anti-virus and will work to distinguish if any malware is downloaded onto any of the devices that the firewall is employed to look after.This essentially makes the firewall a double-layered protection

device that will be working in conjunction with antivirus installed on the windows and Apache Web-Server virtual machines to keep the machines as protected as possible at all times.

Malware Threat Types

DDOS Attacks

A distributed denial of service, or DDOS attack, is the act of overwhelming a target server or network with a huge flood of internet traffic. This renders the server too slow to operate and can totally crash entire networks with enough power behind the attack. This is actually considered a federal offense under US law under the Computer Fraud and Abuse Act and can merit prison sentences of up to 10 years if the perpetrator can be traced.

At its most basic level, DDOS attacks consist of a large network of bot or spoofed computers that are linked together with specific malware programs in order to connect to a targeted network and overload it with traffic requests. This causes the network to become overloaded with the bot requests and "deny" legitimate requests from users, hence the name distributed denial of service attack. DDOS attacks can be categorized into 3 types, Volumetric attacks, Application attacks, and Protocol attacks. Volumetric attacks are the most common and basic type of DDOS and are essentially what is described above. A targeted network IP is overloaded with a huge volume of data packets to completely take up its bandwidth, resulting in a denial of response to actual users. The second type, Application attack, is a different form of this same bandwidth stealing strategy, utilizing HTTP to request millions of downloads or page refreshes per second on a targeted site. The number of requests overloads the server and results in a crash or indefinite wait times for users. The third and most advanced form of DDOS is the Protocol attack. The aim of this attack is to overload firewall resources to weaken them for other possible forms of attack. An example of a Protocol attack is the SYN flood attack, which exploits the normal handshake process of TCP/IP interactions. The standard handshake consists of 3 steps, receiving information, confirming the request on their side, and then both sides closing the connection. SYN flooding exploits this third step by sending a closing request from a spoofed IP that will not respond to the handshake, preventing the loop from ever closing. The server is then stuck waiting for thousands of closing responses from spoof IPs that will never come, severely slowing it down or crashing it entirely.

DDOS attacks are an extremely common way of slowing down or crashing a network as more people become familiar with the technology. Using the most basic level of Volumetric attacks, it is not hard for the average person to obtain a botnet program to spoof hundreds of IPs and overload a network. This may not be dangerous to a large and secure company, but a coordinated DDOS attack from a group of criminals can cause huge amounts of network traffic and render even the most secure networks useless for a few hours or even days. These groups have been known to hold prominent companies, hostage, for weeks on end out of spite or desire for money. Banks and corporations deal with DDOS attacks daily, which sometimes severely slow down the website or services they are trying to provide to unusable levels.

Unfortunately, there is not much a team can do to actually prevent DDOS attacks from happening as they are inevitable, it is more about what protocols are used to combat them once they happen. Solutions to large attacks can be drastic, letting your customers know that the service will be down indefinitely and completely blocking all ports and traffic to the network. The company will then contact and work with their internet provider to try and pinpoint certain IPs and addresses to filter out and slowly begin trying to restore the network. Frequent routing changes can also help alleviate some of the pressure on the network and open up bandwidth space. The new widespread adoption of server-side cloud technology is a step in the right direction to actually preventing DDOS attacks, as it is impossible to DDOS every single network in a cloud environment. This severely limits the power of attackers and allows security teams to diagnose each node in a cloud server one by one rather than on a single network.

Shell Exploitation

Shell exploitation is using shellcodes to infiltrate a targeted server or machine's vulnerabilities in order to gain access remotely. Being able to have access at any time to the targeted server by shellcodes is called a backdoor. Having a backdoor allows you to access the targeted machine without having the trouble of penetrating the vulnerabilities again. Shellcode for machines is usually written in programming languages such as C, C++, JAVA, PERL, and assembly. On the other hand, web shells are written in PHP, Python, Ruby, and Unix shell scripts.

First and foremost, the attacker needs a vulnerability in the targeted machine or server to exploit its shell. To find vulnerabilities in the targeted machine, vulnerability scanners are used. There are different types of vulnerability scanners. There are network-based, host-based, wireless, application, and database scanners. There are different types of scanners for different categories because they all look for different vulnerabilities that could be either outdated software, open port, or Some of the most used vulnerability scanners by cybersecurity professionals are Acunetix, Burp Suite, Nessus.

For instance, the Metasploit framework will allow attackers or pen-tester to penetrate the target machine on their vulnerabilities and uses exploit modules called payloads to open a shell on the target. There are different types of payloads ranging from singles to stagers and last stages. Staged payloads gain meterpreter access on a target that is vulnerable. Stager payloads allow attackers to upload shellcode or files to the target system after exploiting. Single payloads are standalone exploit modules that allow you to do one task. This could be either adding an attacker as a user or changing a password etc. There are over 200 payloads in Metasploit and all have different purposes for different vulnerabilities.

There are different types of shells that could infiltrate the targeted machine. The most popular shells known are the bind and reverse used in the cybersecurity field. The reverse shell allows an attacker to establish a connection and connect back. Bindshell binds itself to a vulnerable port on the targeted machine which establishes a connection to the attacker. Socket-reuse is also another type that runs inside a vulnerable process to establish a connection for attackers. A reverse shell is used when a backdoor is existent and connects back is not needed. If the targeted machine has an SSH server, web server, and any kind of remote desktop, a reverse shell is not needed.

A simple reverse shell could be demonstrated as creating the executable payload using Metasploit in the attacker's machine. Sending and running the executable payload to the target machine will open up the meterpreter which will connect the payload to the attacker. The attacker will have a shell open in their client and will be able to interact and execute commands on their end.

SQL Injection

A growing threat to business infrastructure is the growing prevalence of SQL Injection. SQL injection is characterized as code that is built with the intention of bypassing, damaging, or infiltrating databases controlled by business entities. Of the techniques employed by adversaries, SQL injection is one of the most common as it is one of the most simple to implement. Whereas some threats take the form of sometimes extremely complex files, scripts, or code, SQL injection can take the form of simple exploitation of website script or comparatively more simple malicious code inserted into existing SQL through the website's own input.

SQL injection usually begins on a website's input utility. The input utility, when used appropriately, is supposed to be used to input information such as passwords, user ids, names, addresses, and so on, however, it is this same utility that adversaries use to exploit the website and is ground zero for an SQL Injection attack. On the database side, when a user enters information that input is relayed back to the database and either retrieves, stores, or updates information in the relevant tables.

Although it may seem a straightforward way to quickly and more accurately update vast amounts of information without employees being required to physically input the information or for costly and time-consuming database design to defend against SQL injection, the adversary uses this to their advantage by entering code that tricks the database into revealing information or changing the infrastructure of the database itself.

Armed with this knowledge about what an SQL Injection attack is and how it is usually performed, it can be used to help defend Team B's own databases against rival teams. Namely, to combat this specific threat to our systems we can utilize several different methods to achieve a strong defense. The first of these methods is to quite simply keep our systems up-to-date with the latest available security patches from relevant vendors. Doing so would eliminate many vulnerabilities that were identified by the vendors that created the systems, programs, or networks in the first place. This simple measure is an imperative first step when creating resilient systems.

Moreover, as stated by UC Berkeley: "Developers can prevent SQL injection vulnerabilities in web applications by utilizing parameterized database queries with bound, typed parameters and careful use of parameterized stored procedures in the database." Put simply, developers can write statements that would disallow users from inputting strings and or characters that have either been identified as problematic or outlaw all input that does not meet a certain set of parameters. This step - although time-consuming and sometimes difficult - can stop many SQL Injection attacks from impacting the target system. Team B will mirror this sentiment in our own systems by ensuring that any user input is checked and verified to be legitimate through the usage of these checks.

Another important step would be to limit the number of users able to make changes to a database. A general rule of security management stipulates that users should only have access to security permissions that they absolutely need to complete their tasks effectively. Typically, it is considered a security risk to allow users permissions for areas that they do not have authority over and thus these permissions should be revoked. In this regard, limiting what the website is able to do on the silent side of things is in line with this thought process. The website should not have any credentials other than insert, delete (conditionally) and update (again, conditionally) privileges to prevent unauthorized changes to the server-side database. For most practical tasks, these permissions would be more than enough to allow the website to complete its designated tasks effectively. Anything more poses a security risk and anything less than what is necessary will either greatly hinder the website, or render the website unable to complete its tasks entirely.

Unlike the aforementioned step, this step can be completed through a simple process of changing the permissions of the database connection with an SQL statement from the admin or root account (or any other account that has the power to grant privileges). Once more, this step can effectively block most if not all attempts to make unauthorized changes to Team B's databases and protect the sensitive information held within while allowing users more freedom to interact with the website. Overall, when taken together these steps can effectively defend database systems against intrusion, destruction, and leaks and will be employed in our systems.

Trojan Backdoors

A very devastating and invasive malware that exists is known as trojan backdoors. These kinds of malicious files can come in a variety of different payloads, ranging from pdf documents with embedded .exe files, all the way to web servers that automatically download malicious payloads when the website is visited. Despite an older version of this exploit being used for our research and education, this malware still is very prevalent in today's cyber-security world and can and will be the cause of much devastation to people's privacy and information online.

These Trojan Backdoors are defined as such because they disguise themselves as something that is not malicious, such as a word document or a .pdf file. If the file is successful in its disguise and manages to get downloaded and opened onto a victim machine, a backdoor is created, allowing for complete access into the user machine, with full access to the desktop, mouse, keyboard, file systems, web cameras, and microphones. It also allows hackers to use keylogger software. This extremely invasive software can allow hackers to be able to basically do whatever they want to the systems that they infect, and when infection can be as easy as accidentally clicking a link, hackers can easily find themselves in control of several machines, and if done correctly, can be done without the users ever knowing that they have been exploited. (LRS, 2021)

These viruses store themselves in the DLL memory, and never write anything to the disk. Ideally, these viruses do not start up another process, as this can raise alarms to anti-virus as well as the user themselves, and execute solely in the memory. The virus should operate within the originally delivered payload, this will provide the virus with the best possible chance of being able to execute and avoid getting caught.

These kinds of viruses are so devastating because of the vast array of options that the hackers have that they can use to dig into and wreak havoc on the system after their trojan has been installed. From literally watching you

through the web camera, to installing illicit or malicious material on the computer, and even recording your keystrokes and audio in real-time. There is really no way to stress enough how serious the invasion of this virus is to infected computers.

In order to combat these malicious files, internet users should make sure that they have some kind of antivirus installed on their computers. Even free antivirus can help leaps and bounds with ensuring that computer systems stay safe and free of malicious viruses. The good news is that it is becoming increasingly difficult to create versions of these viruses without them being detected, as the file uses a Metasploit meterpreter in order to gain access to the machines.

Defense & Detection Techniques

Malware-Detection/Prevention

As stated earlier, Malware is one of the biggest threats that arrive in the real world for information systems across the world. In order to combat all of the different kinds of malware that are out and about on the internet, there needs to be some kind of software or application that makes it possible for the computer to scan through the files and look for any kind of malicious code, and in fact, there are a plethora of options to choose from when deciding what kind of anti-virus one wants to use, and different anti-virus' can be better at adapting the needs of different environments than others.

Despite a large number of options available, Team B has decided to go with AVG Free-Antivirus for our windows machines, and for our Linux machines, we have chosen to use Sophos. Both of these anti-virus applications are free to use and will provide each and every Virtual Machine on our team protection from any kind of threat that could be spread via Malware. The antivirus installed on every single machine on our team will be critical to how much the impact of the penetration will be. On top of all of this software virus protection, the Palo Alto Firewalls themselves will also provide anti-malware support in order to further keep our systems safe.

The way that these anti-viruses work is by comparing the contents and code of files to the stored files of known malware, if these files seem to portray the same characteristics, such as file types and other specific contents. These tactics have proven to be effective on many machines, with the paid versions of these anti-viruses being oftentimes extremely effective, and covering multiple devices. By using these, we ensure the safety of our machines.

Palo Alto Networks Firewall

In order to further secure and moreover lockdown our network of Team computers, Team B decided to utilize the Palo Alto Firewall as extensively as possible. In order to do this, the Apache Web, Ubuntu, and Honey VMs had to be placed behind the inside zone in the Demilitarized Zone (DMZ).

This is an extra layer of protection, ensuring that only information and data that we, the admins, deem fit to go through, and everything else will be denied. In order to do this, IP addresses had to be changed and configured through the PAN Firewall in order to permit any traffic in or out of the demilitarized zone. A configuration of a Windows OS Honey VM after being put inside the demilitarized zone (10.21.113.254/24) in ***Figure 3: Honey VM Inside of Demilitarized Zone (DMZ)***

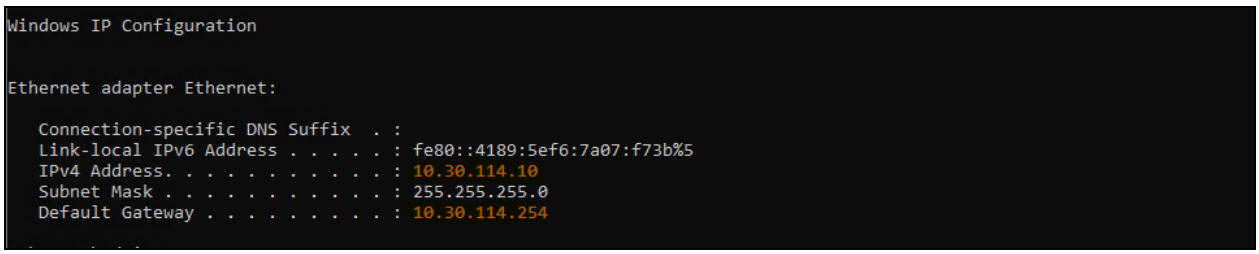

Figure 3: Honey VM Inside of Demilitarized Zone (DMZ)

PAN Firewall Security Rules

After placing the VM's in their position in the DMZ, the traffic in those VM's was now able to be manipulated and monitored by the Palo Alto Networks Firewall.

Once inside the Management console for Palo Alto Firewall, rules then had to be created in order to allow traffic in and out of the Demilitarized zone, and allow the outside to communicate with information that is on the outside. The rules used in order to allow traffic in and out of our DMZ can be seen in ***Figure 4: PAN Firewall Security Rules List*** *below*. These rules for the DMZ allowed certain ports to be accessible from the outside into the DMZ and vice versa. As per the rules given in the Firewall, remote connection and proxy services connecting to any machine in the team's Demilitarized zone connection to a machine in the Demilitarized zone would not be able to be completed and be subsequently stopped.

Using this configuration, we will ensure that no outside traffic will be able to touch any of our systems, and we will be able to fully monitor everything that is going on in the network. It is this kind of precaution that is necessary in today's world in order to not suffer the potentially devastating consequences of a cyber attack.

Any kind of attack or malware that is found on the devices will immediately alert the firewall and allow for one of the Team Administrators to handle the situation promptly. All of these precautions allow us to focus on the discovered threats ahead, rather than the threats that are already out there that are preventable. This is how Team B will utilize the Palo Alto Networks Firewall with our VMs.

				Source				
	Name	Tags	Type	Zone	Address	User	HIP Profile	Zone
1	egress-outside-url	egress	universal	inside	any	any	any	outside
2	egress-outside	egress	universal	inside	any	any	any	outside
3	inside-to-dmz	dmz	universal	inside	any	any	any	dmz
4	internal-dmz-http	internal	universal	inside	any	any	any	dmz
5	internal-dmz-https	internal	universal	inside	any	any	any	dmz
6	egress-outside-conte...	egress	universal	inside	any	any	any	outside
7	block-im-and-p2p	internal	universal	inside	any	any	any	outside
8	foxy-proxy-block	egress	universal	inside	any	any	any	outside
9	intrazone-default	none	intrazone	any	any	any	any	(intrazone)
10	interzone-default	none	interzone	any	any	any	any	any

Figure 4: PAN Firewall Security Rules List

Palo Alto Anti-Malware Effectivity

As stated earlier, Palo Alto, while working as a Firewall, also includes a built-in anti-virus system that works to find viruses when they are downloaded onto a machine that the firewall is looking over. The effectiveness of this anti-virus was put to the test by Team B, in which all antivirus and Windows Defender were removed from a Windows VM, leaving just Palo Alto looking over the windows machine. Then, a malicious Trojan-Backdoor payload was intentionally downloaded. However, the Palo Alto Firewall was successful in finding and eliminating the virus before it had the chance to download onto the machine. This action can be seen in ***Figure 5: PAN Firewall AntiVirus Virus Removal***

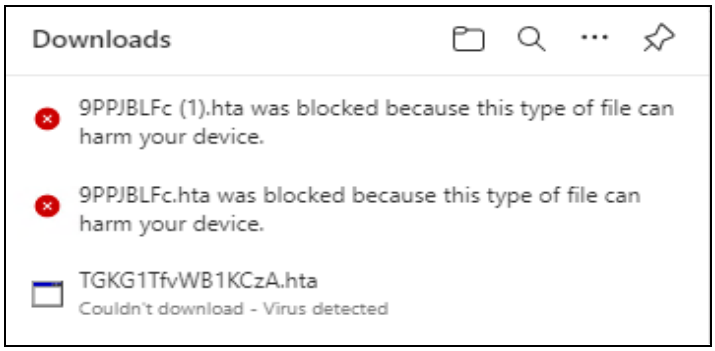

Figure 5: PAN Firewall AntiVirus Virus Removal

It can be confirmed that Palo Alto was the entity that removed the virus when the machine was moved out of the created DMZ (10.30.114.254) back to the outside network (192.168.72.254), the same file was executed and delivered the payload as expected.

This proves that the Palo Alto Firewall anti-virus was efficient at removing the virus that was used in our exploitation of other systems and was successful in ensuring the safety of its protected systems in this threat.

Anti-Virus Programs

The most common and effective protection against malware for the average user is the use of a good antivirus program. These are available both for free and as a subscription-based program, but the free versions are more than enough to keep your PC protected from general malware and suspicious programs. As an example, AVG Anti-Virus is available online for free download as an option for all Windows users. This is a comprehensive antivirus that scans your PC at select time intervals to see if there are any changes that could raise red flags or compromise the security of your system. It will also quarantine files if they are detected as malicious in order to save your system from potentially irreversible damage. Users will be alerted if their system is "At-Risk" or "Compromised" and in need of a rollback or uninstall of malicious programs/files.

The benefit of this defense mechanism is that it is easily usable and obtainable by the general public, there is no skill involved and it automatically scans and repairs your system with a few clicks. Also, network administrators can easily add an antivirus to all computers on their network through group policies and network-wide installs. This will have minor security breaches like malware and phishing attempts covered automatically so security teams can worry about them less.

Most of this antivirus software that exists uses a database of known viruses that is used in order to detect any kind of virus that appears to be the same or matches the contents of the virus that is administered in its database. When a virus is downloaded to a computer with antivirus, if the file is recognized as a threat from the database that it is using, then it will either quarantine the file (Store the file but disable it from executing) or delete the file immediately in order to prevent the file from making it onto the computer at all.

With antivirus, of course, there are exceptions, especially with free versions of antivirus, as it is not always a completely safe and sound solution to malware attacks on computers, as if the antivirus is hit with a zero-day virus, it would not be detected and would slip through the detection process, allowing the malicious file to execute on the computer. There are also other ways that hackers can gain access to systems using vulnerabilities in browsers and other computer services that free versions of most anti-viruses do not cover with their protections. This is why it is important to understand the difference between the free and paid version of antivirus software and how it can protect you better than the free versions.

Of course, it is much appreciated that these antivirus makers make free software for users to use without having to pay for protection. This being said, most anti-virus software includes a free version as well as a paid version, with the paid version usually offering more protections and services in order to keep the computer that it is running on safe from vulnerabilities, viruses, and other potential exploits. These services usually include password

managers, file shredders, phishing protection, browsing protection, and many other services that come with paid subscriptions.

Because of the prevalence and sheer potential for devastating virus attacks in today's world, if users can afford to pay for the paid antivirus, it should most definitely be purchased, as these protections may not seem to be necessary at all, but once the threat is actually there, it will be absolutely worth the cost. The extra security that can be brought to the table with the paid versions of antivirus cannot be understated.

Overall, antivirus is the most sure-fire and straightforward way to keep your computer system safe from malware and other online threats. Free versions of antivirus work, but if it's possible, go ahead and get the paid version of these anti-virus in order to ensure the safety of your devices.

Conclusions & Future Implications

Defensive Conclusions

Overall, the only way to really be safe from cyber-attacks today is to be a step ahead of the adversaries that are out looking for vulnerable systems to exploit. The systems that Team B will deploy are very solid and reputable software that enable maximum security measures to be implemented across a personal network. Alongside our strong infrastructure, we will penetration test within our network to all machines as white-hat hackers to find any vulnerabilities that need fixing. We will be updating our software and security so that there will be less chance of any attacker infiltrating our systems

Securing the entire environment as thoroughly as possible is the best practice in preventing cyber attacks, as waiting until the threats have already arrived and are attacking has proven time and time again to be ineffective, and has led to the devastation of many once-powerful and reputable companies. Consistent monitoring of your network and potential intrusions is key to keeping the systems secure at all times. Having a correct security response protocol is also essential to efficiently deal with any threat that may arise. With the tactics that Team B has deployed into our environment, it has coverage for any incoming threats or security attacks.

With the methods for defense and offensive that were deployed by Team B throughout the semester, we can come to the following conclusions regarding what happened and the future implications that are involved. To start, our defenses (Firewalls/Antivirus) proved to hold strong against scanning and other sniffing techniques performed by other individuals. No drastic changes or issues occurred within any of our systems that were deployed under our firewalls, and no functionality was removed, ensuring that the CIA triad of confidentiality, integrity, and availability were all upheld throughout the semester for Team B.

Between the Firewall and antivirus, both the windows and web servers were able to mitigate all of the threats that were thrown toward our network segment, and all threats were neutralized before they could take serious effect on our computers. Further, the Firewall, in particular, showed to be sufficient in and of itself in antivirus and overall protection of the systems and was extremely impressive in its standalone capabilities. Essentially, all of our defensive measures proved good enough to withstand any attempt to affect our systems.

Offensive Conclusions

Offensively, the Malware that was deployed was very effective in achieving the goal that it was meant to with only one flaw. The malware that was deployed to the other systems was a Trojan backdoor .hta file that would be automatically downloaded to the victim's computer whenever a certain web address was reached. Once the file was opened and executed, the file opened a reverse shell that allowed remote manipulation of the machine.

Despite the VM’s incomplete functionality, the virus could have monitored the desktop at all times, taken remote control of the keyboard and mouse, screenshot the web camera, recorded audio from the microphone, stolen session tokens, escalate privileges, and given root access to the file system of the computer, plus smaller-scale intrusions that could have been made. A screenshot of remote system information and ipconfig results can be seen in ***Figure 6: Remote CMD Access: UserID and IPCONFIG***

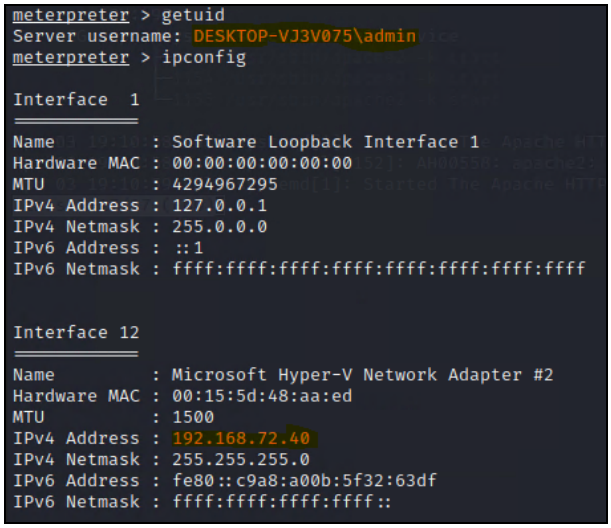

Figure 6: Remote CMD Access: UserID and IPCONFIG

Despite the viciousness of the virus, there was only one issue, which in the real world frankly would have most likely thwarted the attempt at downloading the virus, with the issue being that Team B could not get the virus to execute without alarming the anti-virus or windows defender before being executed. This meant that while the virtual machine was behind the firewall and still had antivirus and windows defender enabled, there was no way for the trojan to make a solid connection back to the Attacker machine. After jumping through these hurdles, the trojan was still able to be installed on a windows virtual machine that had all of these defenses turned off in order to give the team the learning experience of setting up the malicious web server.

Overall, our offensive strategies were effective in penetrating into a victim's computer and allowed us insight into what is really happening when hackers get a hold of a compromised system.

Future Implications

In the future, finding viruses and ways to deliver viruses without antivirus or firewalls being flagged will become an increasingly difficult problem for hackers to deal with online. This means that hackers will have to be more swift and creative with their approaches in order to sneak past defensive measures and really wreak havoc on the infected devices.

Additionally, cybersecurity professionals and software are going to have to stay very diligent in how they go about protecting their systems, and ensure that they stay on top of and update their antivirus software as well as their firewalls in order to protect their systems completely. Cyber attacks are not going anywhere anytime soon, so it is crucial that cyber-security professionals stay a step ahead of the game in order to make sure the good guys are always one step ahead of the good guys.

While only one form of malware was implemented in our team's attacks, it should be noted that there are thousands of different attacks that can be performed on computers, with hundreds or thousands of exploits and viruses being created every single day. (Harrison & Pagliery, 2015) So there is no shortage of attacks that can be created and used to exploit many internet users' systems. So in order to stay protected from outside threats in the cyber environment: stay diligent, stay resilient, and stay ready for attacks.

References (APA Format)

Federal Trade Commission. (2022, January 11). How to recognize and avoid phishing scams. Consumer Information. Retrieved February 16, 2022, from https://www.consumer.ftc.gov/articles/how-recognize-and-avoid-phishing-scams

Harrison, V., & Pigliery, J. (2015, April 15). *Nearly 1 million new malware threats released every day*. CNNMoney. Retrieved April 7, 2022, from https://money.cnn.com/2015/04/14/technology/security/cyber-attack-hacks-security/index.html#:~:text=More%20than%20317%20million%20new,threats%20were%20released%20each%20day.

HP® The top 10 worst computer viruses in history: Tech takes. The Top 10 Worst Computer Viruses in History | HP® Tech Takes. (2020, November 4). Retrieved February 15, 2022, from https://www.hp.com/us-en/shop/tech-takes/top-ten-worst-computer-viruses-in-history#:~:text=Once%20a%20laughing%20matter%2C%20computer,cost%20of%20over%20%2455%20billion.

Kali Linux. (2022, March 30). *What is Kali Linux?: Kali Linux documentation*. Kali Linux. Retrieved April 7, 2022, from https://www.kali.org/docs/introduction/what-is-kali-linux/

Lynch, B. (2020, February 3). What is a zero-day exploit: Protecting against 0DAY vulnerabilities: Imperva. Learning Center. Retrieved February 16, 2022, from https://www.imperva.com/learn/application-security/zero-day-exploit/

Petters, J. (2020, March 29). *What is Metasploit? the beginner's guide*. Varonis. Retrieved April 7, 2022, from https://www.varonis.com/blog/what-is-metasploit

University of California: Berkeley. (n.d.) How to Protect Against SQL Injection Attacks. Information Security Office of UC Berkeley. Retrieved February 15, 2022, from https://security.berkeley.edu/education-awareness/how-protect-against-sql-injection-attacks

Solutions, L. R. S. I. T. (2021, April 15). *Security education: Backdoor trojan*. LRS IT Solutions. Retrieved April 7, 2022, from https://www.lrsitsolutions.com/Blog/Posts/138/Security/2021/4/Security-education-Backdoor-Trojan/blog-post/