5-2018

The Implementation of a Cybersecurity Testbed for Education and Research

Elisabeth Dubois
University at Albany, State University of New York

The Implementation of a Cybersecurity Testbed for Education and Research

Submitted in Partial Fulfillment of

the Requirements for the Degree of

Bachelor of Science

in

Digital Forensics

University at Albany Honors College

Elisabeth Dubois

Research Advisor: Sanjay Goel, Ph.D.

Research Mentor: William Augustine, M.B.A

May 2018

Abstract

The Cyber Innovation Lab brings with it the understanding and collection of the threats, vulnerabilities, and risks that one's IT resources present. Due to the increasing complexity and degree of cyber threats, innovation regarding current attack vectors and defense mechanisms are needed. The lack of realistic threat landscapes and the need for accurate security analysis and defense strategies prompted the implementation of a cyber physical system for defensive research. The project depicts the processes to establish the testbed, the methodology used in classifying the architecture, threat and attack analysis, as well as preliminary results. The paper presents the testbed architecture of the realistic corporate environment implemented at The University at Albany, known as the Cyber Innovation Lab. It elaborates on the development, setup, and initial results, along with the practical challenges faced during the early stages of development. In the development of the Cyber Innovation Lab, coordinated and random attacks were carried out on the testbed to determine how the data would propagate on the system allowing for current defense mechanisms to be reviewed. This innovation hub will serve as a breeding ground for new defense techniques and will be a testing environment for local businesses and students.

Acknowledgements

With deep gratitude, I would like to thank Dr. Sanjay Goel for being my research advisor and giving me the opportunity to conduct undergraduate research. Without his help, the work presented in this thesis would not have been possible.

Special thanks to William Augustine for being my research mentor, for helping me understand research strategies and to problem solve technical problems. I would also like to thank Nick Rizzo, Anthony Capece, and Sidharth Shamshabad for their work in helping assemble and configure the network to make it operational.

I would also like to thank Dr. Chang for her guidance in completing my thesis.

And, finally, thanks to my parents and friends for believing in me and pushing me to reach for the stars.

Table of Contents

Introduction

Today, the multitude of cyber-attacks and their respective consequences have major impacts on information security. The security breaches that have occurred at businesses such as Target and Equifax clearly show that cyber-attacks have been on the rise. In 2017, 159,700 cyber-attacks targeting businesses were discovered, with thousands unrecorded (OTA, 2018). The average monetary loss due to cyber-attacks in 2017 was around $21M per company (Ponemon Institute, 2017). Although many of the losses due to cyber-attacks are monetary, there are numerous factors to consider when attempting to prevent or respond to an attack. Many of the business figures, including the monetary calculations record only the direct loss and do not account for the consumers loss of confidence following an attack. For instance, following the Target data breach in 2013, not only was there a significant loss of customers resulting in decreased revenue, but Target agreed to a $39 million settlement with many U.S. banks (Garcia, 2015). The Equifax breach is one of the largest data breaches to date, with over 143 million Americans sensitive information, including addresses, credit card information, and social security numbers being exposed. Although a monetary value is not necessarily linked to the breach, Equifax continues to faced severe public backlash (Yurieff, 2017). Target and Equifax are just two of the more recent breaches, many other companies have had attacks and face similar loss (Fung, 2018). The problem with the increasing amount of cyber-attacks is that technology continuously advances, and their respective security controls are not being maintained and updated as rapidly as threats are evolving. Currently, there is more demand for cybersecurity professionals than supply, meaning there are not enough individuals working to test and research the propagation of attacks and the necessary defense strategies.

Cyber-attacks are considered the number one external risk factor facing businesses today, according to a quarter of Chief Financial Officers (Graham, 2017). With the evolving technological

trends, which include the expansion of IoT, keeping up with the changing trends is a difficult task. Since technology changes on a regular basis, the defense and security mechanisms often lag behind. In addition to this lag, sophisticated attackers can discover and exploit previously unknown vulnerabilities in systems and software known as zero-day exploits. The increasing threats and vulnerabilities that businesses face can be mitigated by implementing a physical testbed which can be used in educational environments as well as research capacities.

Despite companies continuing to innovate, readymade infrastructures are needed to aid in the evaluation of new defense techniques. The Cyber Innovation Lab presents a method to better investigate IT security by creating a realistic threat landscape, on top of a physical cybersecurity testbed. The goal of this testbed is to make concrete assessments, in hope that defense mechanisms can be better understood, resulting in more secure systems. To thoroughly understand the risks to one's IT assets, the common digital weapons and various attack vectors must be studied. In actively finding where the defense mechanisms are lacking, attacks across the established networks will be conducted and preliminary results will be collected to aid in configuring solutions. The projected results will have important implications for cyber security and allow for a better understanding of the common cyberweapons and the defense tactics that could be implemented to protect against future attacks.

This paper will focus on the development of a test platform, including the implementation of the architecture diagrams, laying out of attack vectors, refinement of data collection, and potential results that the platform will generate.

Problems Confronting IT System Security: A Literature Review

In this section, a review of the existing security testbeds as well as threat landscapes that have been developed will be established. The prior work in both areas have been leveraged to develop the testbed.

Many testbeds have been proposed and created in the past, but due to the evolving technological landscape, resulting solutions are only viable for a small window of time. Much of the research conducted may analyze the propagation of attacks on realistic testbeds, but this research often fails to provide realistic measures in defense mechanisms (Ashok, Hahn, & Govindarasu, 2011).

The Defense Technical Information Center, Cyber and Electronic Warfare Division in Edinburgh, Australia review cyber ranges and computer network operation testbeds worldwide. The cyber ranges are categorized on whether they are simulations or emulations. Simulations are software based models used to explore behavior (Fite, 2014). A testbed utilizes emulations when realistic software applications are run on dedicated hardware. Since emulation uses real machines, operating systems, and applications, the testbeds provide a more realistic environment (Leeuwen et al., 2010). Using the specified categorization techniques, it was determined that the cyber ranges that supported a mixture of the simulations and emulations would be most beneficial (Davis & Magrath, 2013). The project cover uses a specialized testbed that encompasses simulation and emulation techniques to create a realistic landscape consisting of physical computers, operating systems, and applications. In creating a realistic cyber environment, research on advanced cyber threats, tool development for increased defense mechanisms, and skill development in cyber operations will be established.

Before running attacks on a system, a testing lab needs to be created where attacks can be verified, machines can be configured, and an understanding of the various exploits can be developed. Practice and testing are invaluable when it comes to running a full attack (Kim, 2015). It is important to create a fully encompassing environment that serves a realistic corporate testbed, while ensure that it is a secure testing site.

Various testbeds have been created to analyze the propagation of malware on critical infrastructure and corporate networks (Lynn III, 2010). For research in cyber security and defense testing, federated cyber ranges and testbeds are being implemented in government and private environments. Two large initiatives called Future Internet Research and Experimentation (FIRE) in Europe and Global Environment for Network Innovations (GENI) in the United States link testbed resources and allow researchers to utilize the reviewed models ("Workshop on CNERT," 2017). Besides the US Military, many government agencies including the Department of Homeland Security, have created testbeds such as the Defense Technology Experimental Research (DETER). DETER is a cybersecurity testbed that contains various environments that allow researchers to safely test defense mechanisms against realistic and live threats in a controlled environment ("CSD-DETER," 2013). Existing testbeds are continuously referenced to ensure the correct implementation of physical machines, operating systems, and applications in mimicking a realistic environment. In a production environment, running a test using untested exploits is not recommendable as this could have the potential to take down a critical system. Thus, creating a security testbed and testing relevant exploits is beneficial in analyzing a realistic depiction of the propagation of exploits (Calvet et al., 2010).

Due to new technologies, new threat vectors are constantly emerging that can be exploited by hackers. When establishing the threats to a network, it is important to use various standards and

reports that rank the threats in respect to likelihood and potential impact. Using standards like OWASP, the major attack vectors that are common among a typical system can be established.

Prior to building the architecture diagram, many avenues have to be considered when attempting to create a realistic testbed including analysis of potential threats and security mechanisms (Cardenas et al., n.d.). In reviewing the security mechanisms in place, penetration tests will be used to run exploits. Using penetration testing allows for the network to be evaluated, assessing the efficacy of defense mechanisms. In conducting penetration tests, it is important to utilize a secure environment to evaluate security vulnerabilities (Türpe & Eichler, 2009). The implementation of penetration tests on the developed networks helps find the security holes in current defense strategies, allowing for system patches to be developed.

The purpose of the Cyber Innovation Lab is to better investigate IT system security by providing a realistic threat landscape by implementing a physical cyber security testbed. The lab is being designed to focus on common cyber threats and the analyzation of the vulnerabilities of realistic systems allowing for attacks. With the creation of the lab, baseline attacks will be run and the data from those attacks will be imaged to conduct future analysis.

Architecture of the Cyber Security Testbed

When it comes to the understanding of the risks one's IT assets face it is not only necessary to understand why it should be done, but one must look at how this can be done and how it should be completed. In looking at the propagation of malware and collection of the results, research must be completed in a managed and monitored lab environment, as presented through the creation of the Cyber Innovation Lab.

Initial Design. In my research, I am assembling a networked system to conduct testing. Despite a real IT system being difficult to capture, the complexities of the real world will be mimicked as close as possible. This mirror image of a realistic threat landscape will be completed by determining different attack avenues and any flaws that may be present.

The initial lab was created using 2 dozen Raspberry pi's and 4 network switches. The lab used 24 Raspberry pi 3's to construct a testbed in which four networks were built each with 6 pies connected together with 4 layer-2 ethernet switches. Raspberry pi's are a single board fully functional computer. To mimic a natural environment, without the exorbitant costs and need for an increase in physical lab space as presented with pc's, raspberry pi's were chosen. Due to the size of the various Raspberry pi's, their processors often produce a surplus of energy, leading to concerns on overheating. This being the case, each pi in the designed system is equipped with a CDU heat sink (a protective 9-layer case with a 5V fan) to improve heat circulation and reduce overheating. Once the processing speed was considered, the storage mechanisms that were to be used to collect the computer data were reviewed. The storage capacity of the machines is provided through microSD cards which act as a hard drive.

Prior to installing anything on the SD cards, all the cards were formatted using SD Formatter V4.0. The specified operating systems for the pi's was systematically installed on each SD card. To work with the machines, NOOBS, a specific installation software, was placed on each pi. Once the software of choice was selected, the SD cards were placed in the various machines. Working with the raspberry pi's, Linux Raspbian distributions were used on the corresponding SD cards to work as the operating systems on the pi's. The boot sector allowed for Linux or other non-Linux options to be installed. The varying capacity of the pi's to host a variety of operating systems

allowed for a broad range of victim profiles to be researched and created. To add variety to the initial lab setup, MS Windows 10 IoT Core was researched and later utilized.

Aside from basic problem solving, the first error to arise was a boot error when booting the operating systems on the various pis. After an attempt to boot into the specified operating system, the long wait times resulted in a continuous loop of the Raspbian rainbow boot screen. Numerous attempts were made to redownload the operating systems in an attempt to fix the problem. Once it was determined that the error persisted despite changing the image of operating system used, further research was conducted on where the error was. Google searches, database searches, and forums were used to figure out if the rainbow loop was a physical hardware problem or a software program.

Finally, after much time spent trying to problem solve the error, it was determined that the error was that the kernel.img file failed to boot. The reasons behind such an error was low voltage or an improper SD card. Now that a legitimate reason behind the error was presented, solutions could be determined. The first solution, that the power supply was producing too low a voltage was investigated. The power cable supplying the pi could have been unable to fully support the device and power the machine resulting in the low voltage. Upon further inspection of the device through another boot, it seemed like a faint lightning bolt could be seen in the bottom of the boot screen, signifying that the machine did not have enough power going to it. Therefore, to check the reliability of the power cords, one of the pi cables was switched out for a Samsung charger. This swap allowed the pi to run correctly and fixed the boot problems they were previously having. It was determined that to fix the problem, the power cords that came with the raspberry pi's had to be replaced in order to output the correct voltage to make the pi's work properly. Research was conducted to determine the model and type of power cords that were not only efficient, but that

allowed for the downloaded operating system and future applications to consistently run on the machines.

Design Update. After an initial review of the testbed was completed, it was decided that 12 Dell OptiPlex's were to be added to increase the realistic nature of the testbed. Once the new machines were added to the landscape, networks were created consisting of varying amounts of pi's and OptiPlex's. This addition formed an environment which was able to support various software's and operating systems. In a sense, the computers brought a new dimension to the testbed, allowing for further operating systems and applications to be utilized, as discussed later. The implementation of the devices allowed for an initial network map to be configured displaying the various networks and the interconnectivity between them. Figure 1, as presented below is a network architecture diagram of the realistic testbed created.

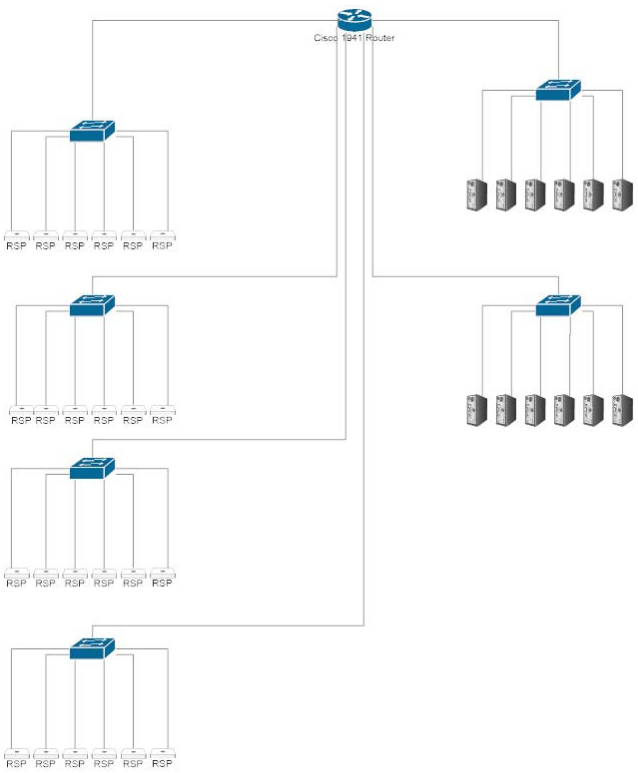


Figure 1 Depiction of Realistic Networks

To touch on the general layout of the above diagram, all 6 of the networks connected to a central Cisco 1941 Router. This Cisco Router serves as the connection point for the various networks. The four left hand networks are made up of 4 networks, each containing a single switch connected to 6 Raspberry Pi's. To begin, the 4 networks served as the baseline to begin researching the proposed project. Initially, the networks of raspberry pi's were designated as Linux machines due to the compatibility of the operating system and the devices. In order to diversify, Windows IoT was later researched and utilized to allow for a greater diversity in networks. On the right in Figure 1, there are 2 networks, each consisting of a switch and 6 Dell PC's. The switches as shown amongst the various networks, acts as controllers of sorts. The switches allow for information to be shared amongst a designed network. For example, device 1 of network 1 may share information with device 2 of network 1, enabling the network devices to talk to one another. Upon completion of the network diagram and having an initial lab set-up, further investigation could begin.

Exploits. In working on the creation and initial malware injections, the question of "how the research should be conducted" was raised. Just as doctors would not intentionally introduce a disease into the population to learn more about it, I will not be injecting malware into a wireless networked system. Due to this practice, the networks were set-up in a quarantined lab and are hosted on a private network, while further set-up and validation occurs.

To conduct defense research without introducing malware across the networks, a DHCP Server for Windows for established. A DHCP server assigns IP addresses to client computers. To reduce configuration problems that could arise, the DHCP server was utilized for the network. Due to the previous concerns that the future malware had the propensity to spread quickly, the research is set to be managed and monitored on an isolated network in a controlled lab. In configurating the

DHCP server, a pool of addresses in Windows server 2016 was set-up to assign the client computers different IP addresses.

Testing Malware Injections. Once the how in regard to lab setup was determined, research was conducted on methods and tools to use to begin malware injections. It was determined that the attack machine would be a Kali Linux box. From this machine target enumeration, vulnerability assessment, exploitation can be executed using the penetration testing tools present on the Linux distribution. In setting up an attack machine, it could be situated on the individual networks to see the effects of an attack on the various networks composed of 6 machines. Once the attack computer was established, various attack vectors were researched for their commonality and ability to help bring change to real systems. In this research, various attack vectors including SQL injection, Cross-Site Scripting, Denial of Service (DoS), Distributed Denial of Service (DDoS), Cross Site Request Forgery, Account Hijacking, Brute Force, Man-in-the-Middle Attack, etc. were chosen. Once the various attack methods were researched, it was essential to determine the most prominent threats and the attack vectors that were most viable to run in the quarantined lab. Many of the attacks will be reviewed in further detail at the next stage of the project, but a few select avenues were analyzed on their possibility of occurring and how said attack would be carried out.

Using Metasploit. The preconfigured Kali box was distinguished as the attack machine. In preparation for running malware using the attack machine, initial research regarding attack and defense strategies was completed. In the research, it was determined that using pre-installed applications like Metasploit on the Kali box would serve as a beneficial initial attack vector. Metasploit is a generally accepted and powerful penetration testing framework for developing and executing exploits against a target. Due to the dynamic landscape that Metasploit provides, it can be used to test the vulnerabilities of a computer system or as an intrusion mechanism.

Using Metasploit, numerous attacks and exploits were researched in order to map out a testing plan. Prior to running the malware on the configured system, it was important to validate the operational status of the individual devices and of the networks themselves. In configuring the Linux attack machine to a specified system, the most prominent threats and vulnerabilities were examined. Once the configuration was complete, it was essential to gain as much information about the victims (computers) as possible. Much like a criminal gathering intel on their target prior to confronting them, reconnaissance is a helpful measure in establishing the best way to attack the target. To begin, reconnaissance was done on the victim computer. Prior to exploitation and malware injection, doing research on the victim allows for pertinent information to be determined. In the reconnaissance stage, port scans and other data, including IP addresses were gathered. A highly regarded tool utilized in the reconnaissance phases of penetration testing was Network Mapper (Nmap). Nmap is often used to run security scans, identify host services, amongst other things. This tool comes preinstalled on many Linux machines, so it is an item of convenience. Nmap scans were done in this instance to scan the target host and search for any open connections that may serve as a point of exploitation. In command line, the following was run to determine pertinent information in reference to a range of IP addresses: nmap -A -vv 192.168.56.100/120. In this, the victim machine was selected. A Windows machine with the IP address 192.168.56.104 was utilized as the validation machine to exploit. Once the victim was selected, research was conducted on the various applications that were present on the device. Since the devices were configured solely for testing, there were few services installed in the initial stages. It was determined that the service, Adobe Flash 2016 was on the machine. Therefore, further research was conducted on vulnerabilities present on Adobe in an attempt to exploit it. Since setting up the

networks and running Nmap was simply part of the validation phase, the test exploit was chosen just as a practice. In the research it was shown that Abode Flash 2016 had several vulnerabilities.

After the scans and preliminary research, Metasploit was opened and could be used to run the exploits on the specified machine. In determining the Metasploit module to use, CVE-2016-3105 was selected. Figure 9, shows the CVE specifications that were used to help determine the relatability and ease of use of the Metasploit module.

This exploit serves to exploit Adobe Flash using a malicious SWF file. A SWF file is an animated graphic created by Abode Flash which can be played using Abode Flash Player or certain web browsers. Using Metasploit guidelines, the researched exploit was utilized in an attempt to exploit the victim host.

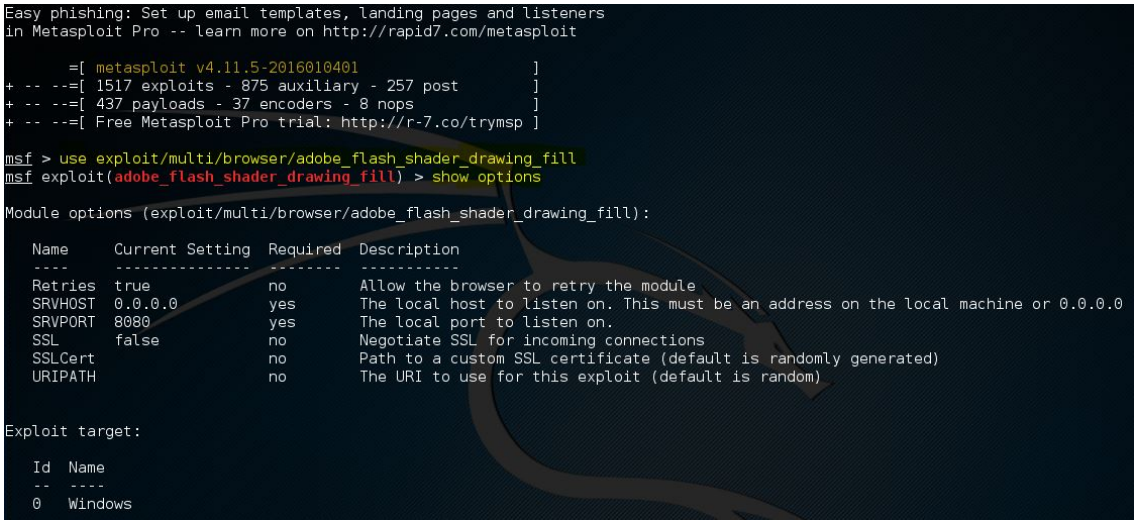


Figure 2 Example of commands run in Metasploit (Hacking Tutorials, 2016)

The above figure (Hacking Tutorials, 2016) shows one of the various exploits utilized to test the network and determine the reliability of the defense mechanisms. To begin using Metasploit, command line in the Kali box was utilized. Once in the command line, the 'apt-get upgrade metasploit-framework' command was run to update Metasploit to the most recent version. Often if an error occurs at this phase, the Kali box may not have Metasploit in its recognized

applications. In this instance, the following command “apt-get install metasploit-framework” can be run. Once updated or installed, “Metasploit-framework” was typed in the command line and Metasploit began to load. From there, a personal decision on what area to focus on was made. This decision was based on the purposed target systems and where the vulnerabilities within those systems may lie. In preparation for the above example, a search was conducted to determine any flash exploits that may be present using “search flash”(Hacking Tutorials, 2016). As shown in Figure 3, the “search flash” command delivered many results. Using the provided descriptions of the potential exploits, the adobe_flash_shader_drawing_fill was chosen and used as the exploit as shown above.

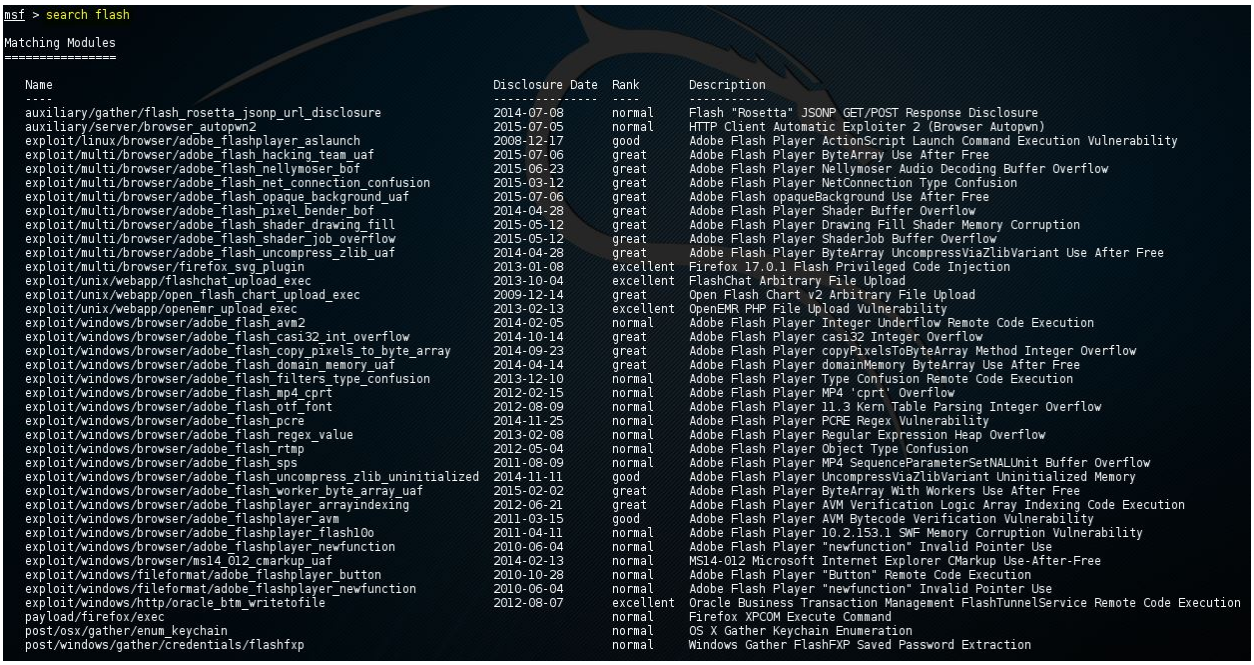


Figure 3 Search Command in Metasploit (Hacking Tutorials, 2016).

After the search command, the use command was utilized as shown in Figure 2. Once the exploit was chosen, reconnaissance had to be completed. To change the options, the “show options” command was used as seen in Figure 4.

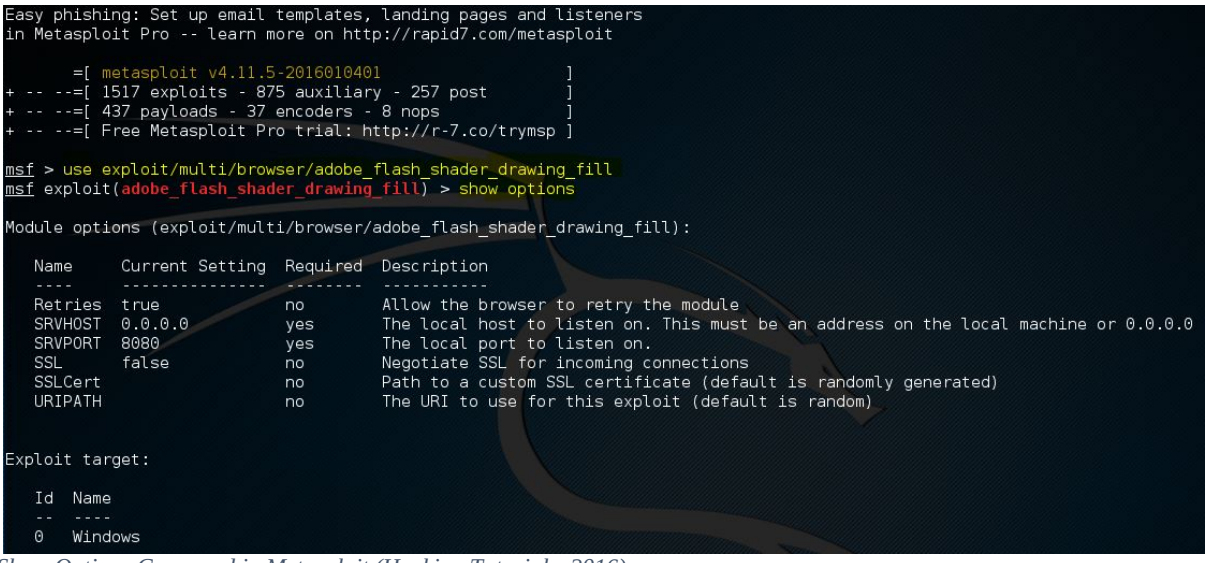


Figure 4 Show Options Command in Metasploit (Hacking Tutorials, 2016)

Using the show options command, different module options were presented. The reconnaissance previously performed on the victim hosts can be used to determine the SRVHOST and SRVPORT to listen on. To change the options to the necessary information as gathered from the victim host, the "set" command was utilized. To begin, the SRVHOST was set as follows "set srvhost 192.168.x.x" with the 192.168.x.x being the IP address of the victim host that was being targeted. Then, the SRVPORT was set "set srvport 80" as during the reconnaissance of the victim host port 80 was open, allowing one to listen using said port. Port 80 is often the port used for web communications and is left open to receive requests from web clients. Due to this, port 80 is often a good choice to listen on a victim host. Once everything was set and the relevant exploit was chosen, the command "exploit" was utilized to begin listening on the victim host. Using the above method, numerous exploits were utilized to exemplify the possibilities and the vulnerabilities of the victim networks. Due to being in the beginning stages of development and testing, there are no concrete data collections existing. The tests conducted using Metasploit were conducted to establish the validity of the attack machine and the chosen tool. Since the exploit run was from 2016, the vulnerability was already patched, but going through the process was beneficial

to establish the operational status of the networks. The results of the initial test help to depict where the future research will aid in the collection and examination of exploits, while working to increase the security of defense mechanisms.

Installation of Operating Systems and Services. Using the initial network diagram and the various IP configurations as presented in Figure 1, a collection of Operating Systems and services were selected. To mimic a realistic corporate environment, varying networks containing numerous OS and a variety of services and applications were needed. In this regard, Figure 5 represents a realistic corporate environment by including a mix of integrated networks hosting CentOS, Ubuntu, Fedora, Windows, PWNPI, and RISCOS. Due to the variety of OS that are in a typical environment, mirroring a realistic landscape allows one to gather data that can be used to help build actual defense mechanisms.

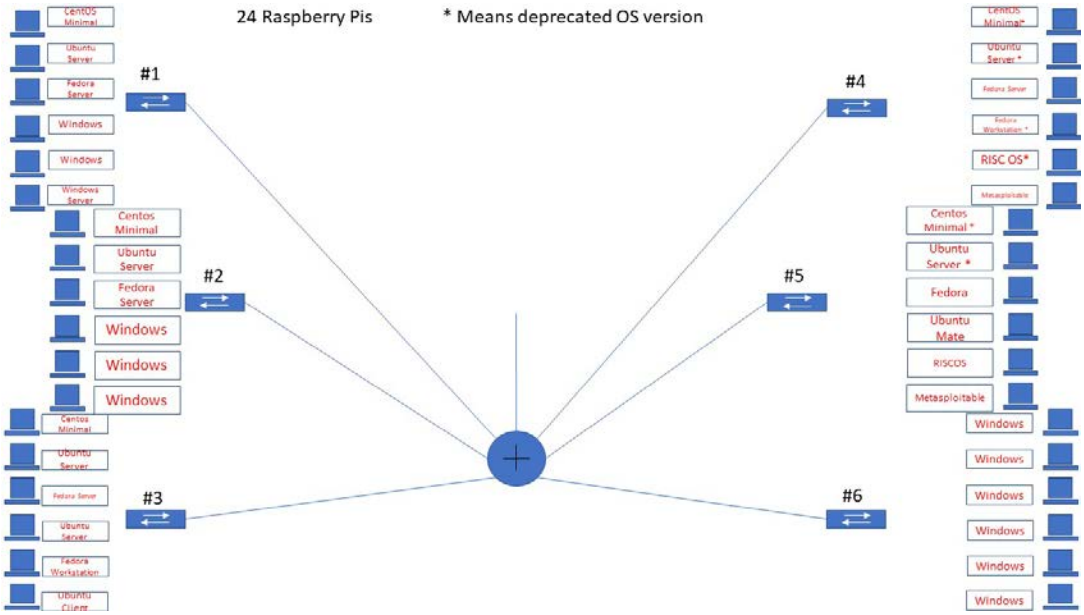


Figure 5 Network Diagram - Draft 2

In configuring the network, many services were determined to be of high importance. Some of the important services were: DNS, Mail, Web, FTP, and SQL. In the application of building a

realistic testbed, it was essential to determine the prominent services that many typical networks and devices have. The mail and web services were pinpointed as common vulnerabilities, so time was spent on researching and configuring those services. From determining the overarching services, the applications are in the process of being researched and selected based on applicability to realistic systems. For instance, Adobe is one of the applications that has been selected due to its widespread use and the knowledge of common vulnerabilities in the application. To mimic a realistic environment, some of the devices will have security implemented (including firewalls), while others will be unprotected systems. The configuration of the services, applications, and relating security is in the process of being constructed and will be distributed among the networks presented in Figure 5. Once the service and applications are chosen various scenarios will be run and threat modeling will be conducted on the various threats that were determined from the chosen services.

Developing a Future Attack and Test Plan

To develop a test plan for the Cyber Innovation Lab I used a formal risk approach. After laying out the architecture of the physical testbed, further research was conducted on the common services exploited as well as the threats that those applications faced. In the various tests using Metasploit, it was determined that further research would have to be conducted to see where vulnerabilities in real-time systems often lie. Once the services of importance were researched and the relevant applications were chosen for implementation in the physical testbed, the various threats and services were examined. Using the Threat Risk Model, the services and subsequent risks were analyzed to ensure the correct controls were utilized and effective measures were taken (Creative Commons, 2017a). The Threat Risk Modelling process includes five steps as presented in Figure 6.

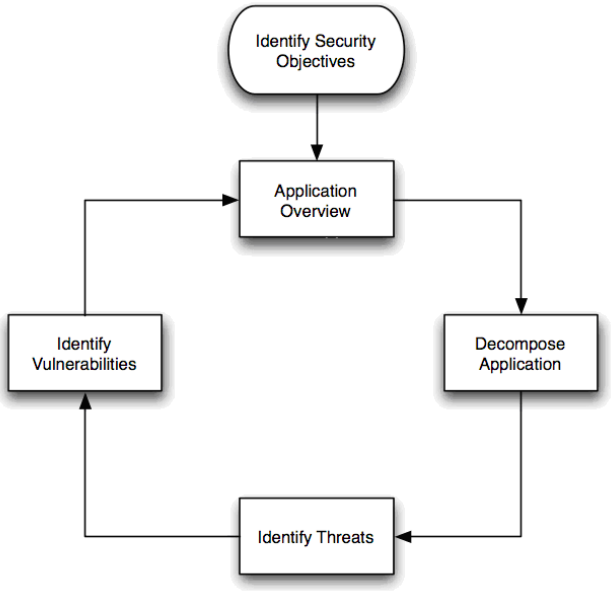


Figure 6 Microsoft Threat Modeling Process (The OWASP Foundation, 2017).

To begin the Threat Modeling Process, security objectives can be determined. Due to the nature of the lab, the testbed includes personal computers on a corporate network, amplifying the risks that realistic systems face. In this, it was important to break down an application's security objectives into: identity, financial, reputation, privacy, and availability. The list purposed is in no means all-encompassing, but it uses the predetermined risk decisions to help select and validate security controls. From there, the application could be reviewed to identify components, data flows, and trust boundaries. The next critical step in the modeling process is to identify the threats. Here, it is important to concentrate on known risks, which can be demonstrated and exploited using common tools and techniques. There are two methods of writing up threats, one of which is shown

in the threat graph in Figure 7. The following method is being utilized to lay out the various attack vectors and determine the prominent threats and the focus of many of the attacks.

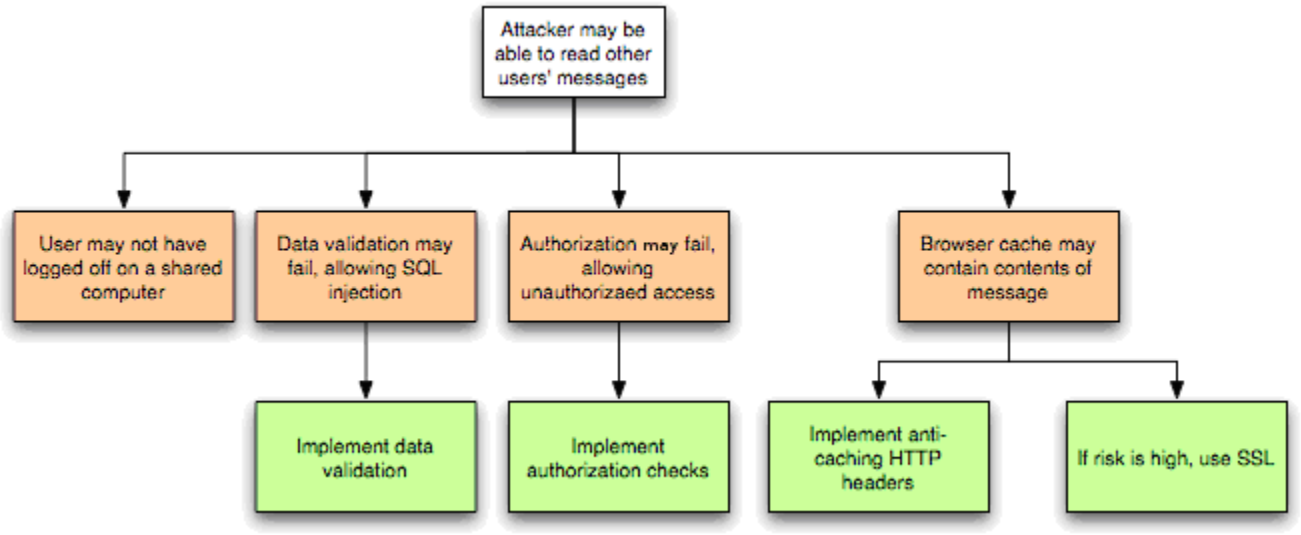


Figure 7 Threat Graph (Creative Commons, 2017a).

Understanding Typical Attacks. It is important to know prior to running malware that there is a human threat present on the various networks. In accordance with the relevant threats determined using the threat modeling process, there are many methods in which the threats can be executed including accidental discovery, automated malware, the curious attacker, script kiddies, motivated attacker, or organized crime. It is important to understand the varying levels of attackers to determine the level of the attacker you are defending against (Creative Commons, 2017a). In regard to this project, a focus will be placed on automated malware, curious attackers, as well motivated attackers. Due to the nature of curious and motivated attackers often using malware or known exploits to affect a network, their actions will be taken into consideration when determining the exploits to use. It is important to get in the mind of the hacker in thinking of the lab and attack vectors as more than a simulation, but a real-life scenario. By taking on the persona of the attacker (often the case with penetration testing), one will be more likely to use exploits that may otherwise not be utilized and understand the propagation and results of an attack.

Security Considerations. It is important to know the security considerations that correspond with the risks that varying IT systems face. Over the years, web application security risks have been prevalent. Due to the widespread use of the Internet and web applications, much of the focus of malware attacks are on said systems. Web applications and Internet services hold a host of valuable and sensitive information about a user, ranging from banking information to emails. Insecure software is making its way into the every avenue of our life's as we are making many daily tasks digital (Creative Commons, 2017b). One of the major aspects the Cyber Innovation Lab will focus on in the future is the risks posed by web applications. Using Figure 8, the OWASP Top 10 comparisons from 2013 to 2017 can be viewed and applied to any testing plans that will be created.

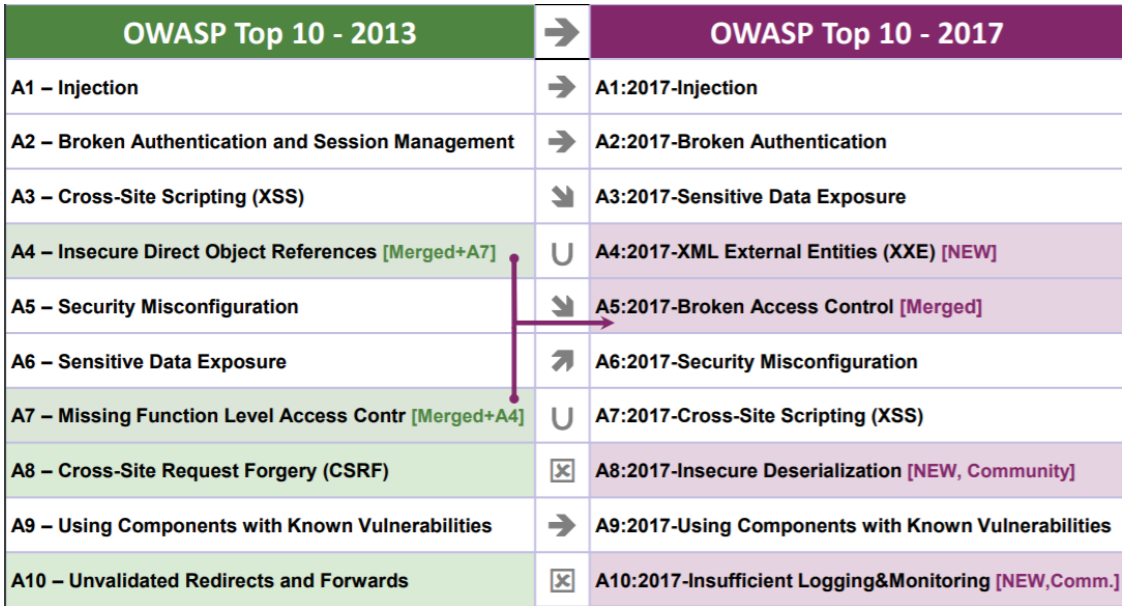


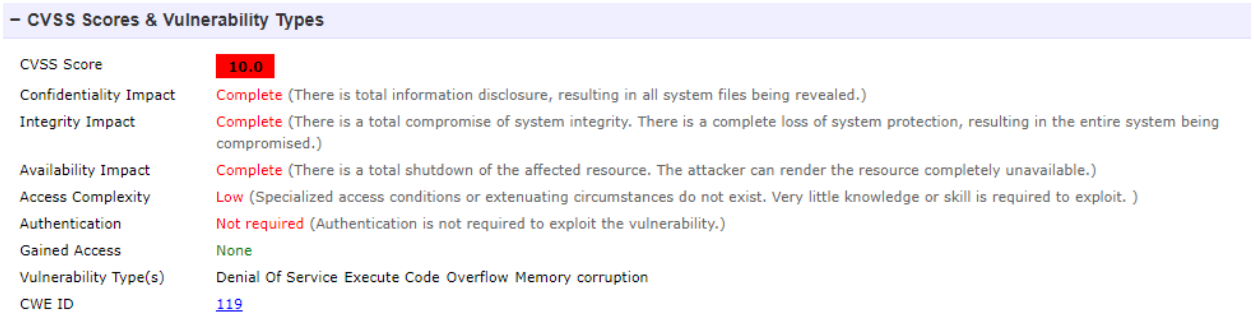
Figure 8 OWASP Top 10

Using the above list of the Top 10 security risks, it was determined that the top risks that web applications face would have be further examined for relevance. The problem with the presented risks is that attackers have an arsenal of attacks and weapons often at their disposal. Each path that an attacker may take represents a risk that may or may not be harmful to a system or business. The OWASP Top 10 list goes into more detail on how you can determine the risk to

your organization and evaluate the likelihood of the threat to the chosen system. Determining the security vulnerabilities on your network allows for further determination of the risks that the system faces. With the multitude of threats, attack vectors, and vulnerabilities a system is readily able to understand and recover from the purposed risks.

Vulnerabilities and Attacks Explained. As discussed throughout the process of establishing the lab, there are numerous concerns that are present in relation to the attack vectors with the potential for grave consequences. Due to the interconnectivity of a realistic network, web application attacks are a high priority concern. For the purpose of establishing future plans, web application attacks are the first avenue that will be examined for providing a basis for further threats to be determined. When it comes to web applications, which consist of a majority of the cybersecurity attack vectors, there are many attack choices. The main avenues for an attack include: SQL injection, Cross-Site Scripting (XSS), and Distributed Denial of Service Attacks (DDoS). SQL injection attacks occur when malicious SQL statements are placed in form fields in an attempt to gather information from a database. The information an attacker gains enables them to access, alter, or delete information within the database. An SQL injection allows an attacker to gain information on the victim machine enabling them to extract credit card numbers, passwords, or other sensitive information. Cross-Site Scripting (XSS) is an attack on browser in which malicious content is present on a web application and executed, in many cases to deface the website being visited. A Denial of Service attack allows for an attacker to take actions that may prevent legitimate users from accessing network resources. A Distributed Denial of Service utilizes requests from numerous IP addresses in order to flood a site with traffic, limiting the response to legitimate requests on the site.

In regards, to the CVE utilized for Adobe Flash as seen in the testing of the network and as presented in Figure 9, the vulnerability can be examined for its real-world applicability. The nature of the Flash vulnerability is provided below to show the capabilities of the CVE exploit.



– CVSS Scores & Vulnerability Types

CVSS Score	**10.0**
Confidentiality Impact	Complete (There is total information disclosure, resulting in all system files being revealed.)
Integrity Impact	Complete (There is a total compromise of system integrity. There is a complete loss of system protection, resulting in the entire system being compromised.)
Availability Impact	Complete (There is a total shutdown of the affected resource. The attacker can render the resource completely unavailable.)
Access Complexity	Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or skill is required to exploit.)
Authentication	Not required (Authentication is not required to exploit the vulnerability.)
Gained Access	None
Vulnerability Type(s)	Denial Of Service Execute Code Overflow Memory corruption
CWE ID	119

Figure 9 CVE Scores (Mitre Corporation, 2016).

In choosing an attack or exploit to utilize, realistic measures are often taken by an attacker to affect the CIA triad of IT security. This CIA triad stands for confidentiality, integrity, and availability. Since these three metrics are essential to key IT infrastructure, a hacker uses what they know to alter or lessen the confidentiality, integrity, or availability of a business, system, service, or data. It is important to point out that in Figure 9, the CIA triad would be greatly affected by the Flash vulnerability, as total information disclosure, total compromise of system integrity, and total shutdown of the resource is presented.

In utilizing the Adobe exploit a DoS vulnerability was expressed and in the real-world said attacks are increasingly impactful. As discussed, DoS attacks focus on disrupting or preventing legitimate users from accessing websites (Rapid7, n.d.). There are numerous reasons for DoS attacks, ranging from state actors to simple hackers. The impact and costs associated with DoS attacks can be wide ranging, depending on the target and the duration of the attack. A large-scale attack on a corporate network as the one mimicked through the creation of the Cyber Innovation Lab, would have a large impact to the organization and their third-parties. Such a large-scale attack may prevent an online business, like Amazon from serving its customers which may costs millions of dollars. It is

essential to remember that DoS may costs millions to recover from, but some of the damage may be irreversible due to public image or customer confidence. For example, the Mirai botnet, a DoS (more specifically a DDoS attack) which hit the internet back in 2015 shutdown the server of Dyn, a company that controlled much of the internet's domain name system (DNS) infrastructure. During the duration of the attacks, many sites were brought down including Twitter, the Guardian, Netflix, Reddit, CNN, as well as many other services (Woolf, 2016). In recent reports, peak-time DNS attacks cost many organizations more than $100,000 per hour, with this number on the rise (Korolov, 2016). A DoS attack has not only the potential to cause high levels of damages and high costs, but it can greatly alter IT infrastructure. My conducting DoS attacks on a test network, the system and the accompanying data can be analyzed to help improve detection and response to real attacks.

Conclusion

In establishing a realistic testbed, attack mechanisms and defense strategies will be better understood, resulting in the ability to create more secure systems. Using the network architecture, exploits can be determined, and malware can be run on the realistic, unsuspecting victim computers. Due to the increasing number of and increasing complexity of attacks, greater losses and long-term consequences are being seen. Analyzing the results of common attack vectors on a realistic corporate network as presented in the Cyber Innovation Lab will help find system vulnerabilities and determine where improvements can be made to improve defense strategies. The vulnerabilities and results from the attacks can be prioritized and examined with respect to the impact and potential loss on a relatable system. The implementation of the Cyber Innovation Lab and the data gathered from the attacks will be essential in helping to patch security holes and recover from losses and mitigate long term consequences of a cyberattack.

References

Ashok, A., Hahn, A., & Govindarasu, M. (2011). A Cyber-physical Security Testbed for Smart Grid: System Architecture and Studies. In *Proceedings of the Seventh Annual Workshop on Cyber Security and Information Intelligence Research* (pp. 20:1–20:1). New York, NY, USA: ACM. https://doi.org/10.1145/2179298.2179320

Bryan K. Fite. (2014). Simulating Cyber Operations: A Cyber Security Training Framework, (June 2012), 36.

Calvet, J., Davis, C., Fernandez, J., Guizani, W., Kaczmarek, M., Marion, J.-Y., & St-Onge, P.-L. (2010). Isolated virtualised clusters: testbeds for high-risk security experimentation and training. *Proc. 3 Rd USENIX Work. on Cyber Sec. Experimentation and Test (CSET)*, 141–150.

Cardenas, A. A., Amin, S., Sinopoli, B., Giani, A., Perrig, A., & Sastry, S. (n.d.). Challenges for Securing Cyber Physical Systems, 7.

Creative Commons. (2017a, July 13). Threat Risk Modeling - OWASP. Retrieved February 14, 2018, from https://www.owasp.org/index.php/Threat_Risk_Modeling

Creative Commons. (2017b, December 29). Top 10-2017 Top 10 - OWASP. Retrieved March 20, 2018, from https://www.owasp.org/index.php/Top_10-2017_Top_10

CSD-DETER. (2013, June 3). Retrieved April 16, 2018, from https://www.dhs.gov/science-and-technology/csd-deter

Davis, J., & Magrath, S. (2013). *A Survey of Cyber Ranges and Testbeds* (No. DSTO-GD-0771). DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION EDINBURGH (AUSTRALIA) CYBER AND ELECTRONIC WARFARE DIV, DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION EDINBURGH (AUSTRALIA) CYBER AND

ELECTRONIC WARFARE DIV. Retrieved from http://www.dtic.mil/docs/citations/ADA594524

Fung, B. (2018, March 1). Equifax's massive 2017 data breach keeps getting worse. *Washington Post*. Retrieved from https://www.washingtonpost.com/news/the-switch/wp/2018/03/01/equifax-keeps-finding-millions-more-people-who-were-affected-by-its-massive-data-breach/

Garcia, A. (2015, December 2). Target settles for $39 million over data breach. Retrieved April 24, 2018, from http://money.cnn.com/2015/12/02/news/companies/target-data-breach-settlement/index.html

Graham, L. (2017, September 20). Cyberattacks are surging and more data records are stolen. Retrieved April 5, 2018, from https://www.cnbc.com/2017/09/20/cyberattacks-are-surging-and-more-data-records-are-stolen.html

Hacking Tutorials. (2016, May 1). Metasploit Commands. Retrieved March 20, 2018, from https://www.hackingtutorials.org/metasploit-tutorials/metasploit-commands/

I. Chapman, l. Bernier, & S. P. Leblanc. (2011). An overview of cyber attack and computer network operations simulation. Presented at the Proceedings of the 2011 Military Modeling & Simulation Symposium, Boston, Massachusetts: Society for Computer Simulation International.

Kim, P. (2015). *The Hacker Playbook 2: Practical Guide To Penetration Testing*. Leipzig: CreateSpace Independent Publishing Platform.

Korolov, M. (2016, May 5). DDoS costs, damages on the rise. Retrieved April 3, 2018, from https://www.csoonline.com/article/3065999/security/ddos-costs-damages-on-the-rise.html

Leeuwen, B. V., Urias, V., Eldridge, J., Villamarin, C., & Olsberg, R. (2010). Cyber security analysis testbed: Combining real, emulation, and simulation. In *44th Annual 2010 IEEE International Carnahan Conference on Security Technology* (pp. 121–126). https://doi.org/10.1109/CCST.2010.5678720

Mitre Corporation. (2016, December 30). CVE-2015-3105 : Adobe Flash Player before 13.0.0.292 and 14.x through 18.x before 18.0.0.160 on Windows and OS X and before 11.2.202.466. Retrieved March 27, 2018, from https://www.cvedetails.com/cve/CVE-2015-3105

OTA. (2018). *Cyber Incident & Breach Trends Report*. OTA - Online Trust Alliance: Internet Society.

Ponemon Institute. (2017). *2017 Cost of Cybercrime Study*. Ponemon Institute. Retrieved from https://www.accenture.com/t20170926T072837Z__w__/us-en/_acnmedia/PDF-61/Accenture-2017-CostCyberCrimeStudy.pdf

Rapid7. (n.d.). Denial Of Service (DDoS) Attacks. Retrieved April 3, 2018, from https://www.rapid7.com/fundamentals/denial-of-service-attacks/

Türpe, S., & Eichler, J. (2009). Testing Production Systems Safely: Common Precautions in Penetration Testing. In *2009 Testing: Academic and Industrial Conference - Practice and Research Techniques* (pp. 205–209). https://doi.org/10.1109/TAICPART.2009.17

W F Lynn III. (2010). Defending a New Domain: The Pentagon's Cyberstrategy.

Woolf, N. (2016, October 26). DDoS attack that disrupted internet was largest of its kind in history, experts say. Retrieved April 3, 2018, from http://www.theguardian.com/technology/2016/oct/26/ddos-attack-dyn-mirai-botnet

Workshop on CNERT: Computer and Networking Experimental Research using Testbeds - Program. (2017, October 13). Retrieved April 16, 2018, from http://infocom2018.ieee-infocom.org/content/workshop-cnert-computer-and-networking-experimental-research-using-testbeds-program

Yurieff, K. (2017, September 8). Equifax data breach: What you need to know. Retrieved April 24, 2018, from http://money.cnn.com/2017/09/08/technology/equifax-hack-qa/index.html