Penetration Testing Using a Triangular Approach

Jordan Christian
jachris15@earlham.edu
Earlham College Department of Computer Science
Richmond, Indiana

ABSTRACT

As technology has become a huge part of daily life, security has grown with it. User data can easily be taken if it is not properly protected. For any business that maintains personal data, it is important to conduct thorough testing to prevent data from being stolen. Historically, the majority of testing simply concentrated on technical security even though there are multiple ways for a system to be hacked. This research describes a more complete penetration test which will allow readers to better understand the security of a system.

KEYWORDS

Ethical Hacking, Penetration Testing, Virtual Machine, Social Engineering, Technical, Physical

.

1 INTRODUCTION

Penetration testing has long been a method for evaluating the security of a system. It is essential that businesses everywhere have these tests done to ensure the integrity of their own framework. Persons thought to have committed serious cyber crimes have landed on the FBI's most wanted list [1]. It is also true that the United States is the number one target for cyber attacks. Figure 1 shows that the United States has more than double the percentage of targeted attacks compared to the next closest country [1]. With this in mind, why would businesses not spend money to have such testing done? As important as penetration testing has become, it is important to note that not all tests are flawless. This is the purpose of this project, to describe and complete a more scalable and reliable test. Many penetration tests only include technical testing. This means only the brittleness of software can be bettered upon completion of the test. Testers rarely include the social engineering and physical aspects that hackers can use. This proposal introduces a way for all three aspects to be tested within a business.

The test was conducted on the campus of Earlham College. Colleges, of course, are a type of business and include vital information related to their own finances, their students, their students families, etc. Colleges are a ginormous target for cyber attacks. Even though this test focused on a college campus, we remained aware of our aspirations of conducting a test that is reliable and scales well. We hope this test will work well for other business and not just colleges and universities.

Earlham College Computer Science Senior Capstone, 2020,

The small spectrum of testing is a huge problem in the world of security. Systems are hacked in more ways than one, however, many testers only focus on one method of testing. The most common is technical hacking. As time has went on, social engineering has become a more popular choice due to the variety of ways it can be implemented. There are many ways to entice a target to reveal personal information. This can be done face-to-face or through means such as email phishing. Physical hacking is often implemented in conjunction with one of the other two hacks. Dimkov et al. demonstrated this by using social engineering to physically hack their targets hardware [6]. Only after using all three methods will you have an in-depth understanding of the security of your system. This is what makes our plan more complete. The plan developed below is a three pronged approach to test the system in its entirety. Any and all flaws exposed are recommended to be changed within the final report. First, similar work related to this project is mentioned below, next, our testing design is laid out in more detail, followed by the final results of the study.

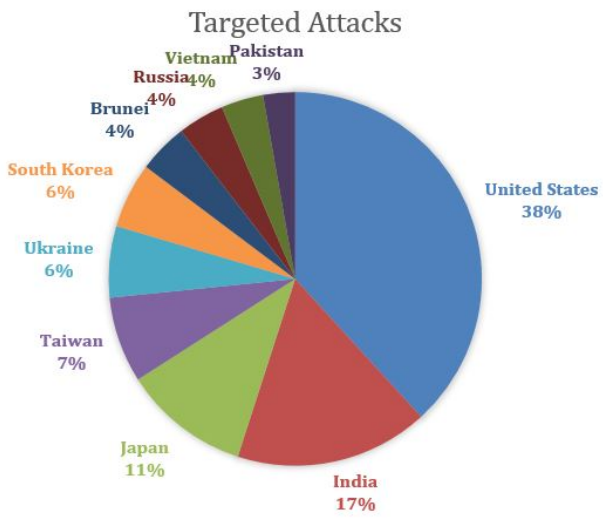

Figure 1: Percentage of cyber-attacks encountered by country [1]

2 RELATED WORK

Most of the previous work done with penetration testing only includes one, maybe two, of the testing methods. Testing usually always entails the technical side but lacks assessments in social engineering and physical vulnerabilities. It can be argued that social engineering is becoming a much more common avenue for attackers

than is technical or physical. Software security has become much better thus forcing hackers to resort to easier, more successful methods. All research we have read has only included tests for one, sometimes two of the methods. This does not give the complete picture of the security of a system.

2.1 Social Engineering

Social engineering attacks have seemingly replaced technical ones as the most common type of security breach. Due in large part to the development of security technology, technical hacks have become increasingly complicated. Hackers have resorted to social engineering and psychological manipulation. Frankly, this method is much easier than technical hacking. Social engineering refers to using deception to manipulate individuals into doing what you want them to do. There are many different ways to do this, as we will see in the review of past studies.

Dimkov et al. described two different methodologies for performing penetration testing using social engineering [6]. Researchers took into consideration how to orchestrate them in the most respectful manner possible. The authors came up with what they believed were five requirements every penetration test should satisfy- those five were: realistic, respectful, reliable, repeatable, and reportable. The two different methodologies the authors designed were an EF (environment-focused) methodology and a CF (custodian-focused) methodology. Within both, there were "actors", which categorized those who would make up the study. In both, there was a custodian who had the "assets", or the physical device the penetration tester tried to steal (It is important to note that these studies were implemented with physical hacking by way of social engineering. They stole items they gained access to.). In the EF method, the custodian was aware of the testing taking place and would aid the tester in certain aspects. The set up of the experiment varied slightly in the CF method. In this method, the custodian, who still had the assets, was completely unaware of the testing taking place. This type of test was a refinement from the EF method and makes it a better overall check for the security of the area. The results from this study showed two things: first, both methods were effective but a company could decide which was better for them to use. Secondly, the methods can have differing effects on employees. Some employees would become angry or frustrated during the test and would alert security as to what was happening (this can be expected from some of the participants). Others acted normal and the social engineering worked as planned.

Jakobsson and Ratkiewicz constructed multiple ways to conduct variations of phishing attacks on users [10]. This was done in order to try and measure the success rate of real-time attacks. A common attack they introduced was the idea of email spoofing. In short, email spoofing is impersonating someone else. They look authentic and it can be hard to determine their legitimacy. These emails often have things like URLs, photos, etc. that upon viewing them, makes your personal information susceptible. The research conducted by the authors was very well-written. They talked about some of the flaws associated with related studies which helped the reader understand the importance of their study. This also help when conducting our own experiment.

2.2 Physical

Physically stealing and then hacking someones hardware is actually more common than one might think. Along with social engineering, this is another aspect of security than is rarely touched upon, rather that be by businesses or in a security class.

Dimkov et al. created and distributed an assignment in which they instructed their students to physically steal laptops given to other employees [5]. Laptops were dispersed to random employees who were informed that the computers were being used in a study to measure something unrelated to security. Each employee was given a lock for their computer. So, the students had three areas of physical security they had to get past: the building, the office, and the physical lock. For this physical hack to work, students had to scout the building, learn the security measures the building used, as well as tracking the employee to learn their daily schedule. There were obviously a lot of hurdles that needed to be cleared in order to perform this study, however it was necessary for the students to learn and understand the place physical hacking has in the realm of security.

2.3 Technical

Many researchers tend to only focus on the technical aspect of security. For the longest time, the easiest way to break into a system was through software engineering. However, different protocols, firewalls, and algorithms have been developed to detect intrusions from hackers. While this avenue of hacking is not as common as what it once was, it is still important to understand ways one might go about it and to perform white-hat hacking to test the security of your system.

Küçüksille et al. proposed a method for performing a test where they used a virtual machine, which had the Kali Linux operating system installed, and tried to hack into a router [11]. The authors tried to test and exploit three different protocols in order to gain access to private information within the router. The three protocols that were tested were SSH (Secure Shell), Telnet, and SNMP (Simple Network Management Protocol). As they used the built-in features of Kali Linux, the authors showed how to perform a penetration test on each of the protocols mentioned above. They also wanted to iterate the importance of continual testing on the router to ensure its security. They stated how many other testers focused their time on servers but routers are important as well as it could allow the hacker to redirect traffic to its desired destination. A bright spot in this research was the authors described low-level details to their method where it made this process easy to follow and repeat. It was hard to find other papers that showed this level of detail.

Epling et al. introduced a slightly different way of performing a penetration test than what was mentioned above [7]. Their method included using what they called a "Pentest Box", which was a mini, cost efficient computer that had its own operating system and system resources to perform network related tasks. They used multiple different operating systems to perform their testing. Common OS's included Debian and Kali, which is a flavor of Linux that is very popular for performing penetration testing. The researchers connected this box to a switch within the network. They used a VPS (Virtual Private Server) and connected it to the outside internet. To pass data from the box to the VPS, they used SSH callback to create

an SSH tunnel. This allowed the vulnerable data from the network to be passed from the pentest box to the VPS. The ethical hacker performing this test then connected their computer to the VPS and this allowed them to access the entire flow of data from within the network. One limitation of this study is the high-level overview that was given. The authors described their testing well, however, it would be difficult to reproduce in other environments as no code was displayed throughout the research.

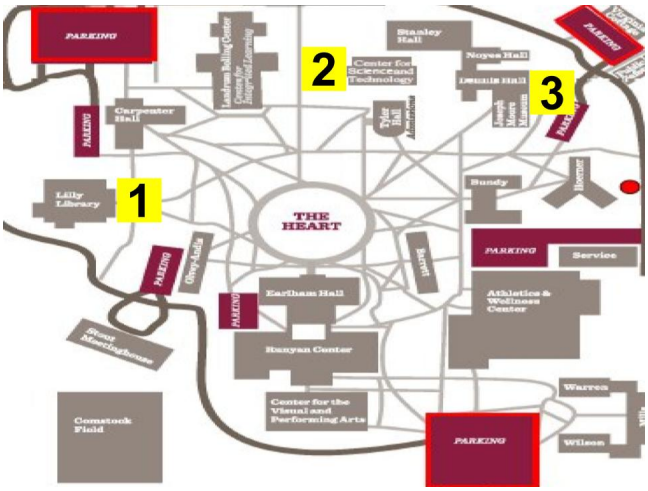

Figure 2: Map of Earlham College

3 OVERVIEW OF TESTING AREA

The testing for this project took place on the campus of Earlham College, which is located in Richmond, Indiana. This section describes the network design and physical structure of the campus to try and provide clarity for the tests that were completed. Earlham is a small liberal arts school that has seen major renovations in recent years, some of which have included network redesigns. Dennis Hall, shown by the number 3 in figure 2, was the location of most of the network for the school. However, upon renovation in 2015, the Center for Science and Technology (CST, number 2 in figure 2) is now where a large number of network servers live. The servers in the CST are used for student and faculty resources. The Computer Science Department has two separate clusters that they maintain for student work as well as student/faculty research projects. These servers are only accessible to those who have registered accounts within the department. These two groups of servers are connected to the colleges main servers which are located in Lilly Library. This is referenced by the number 1 in figure 2.

The servers in Lilly Library provide the ECOpen and ECSecure networks for the campus. These two networks are what connect students, faculty, professors, and guests to the internet. The integrity of these networks were subject to testing during the physical portion of this project. Much of the wired network for the campus runs through underground tunnels. No testing was done on this portion of the network. For the technical testing of this penetration test, pen test boxes were set up under Virtual Box software. The physical portion tested how easily someone could see and connect to the networks operated by the campus while the technical test assumed the attacker has already connected to the network and is now trying to penetrate specific machines under the CS and Cluster domains. It was important to test the outer most layer of security while testing what could happen if this outer layer was breached. For the social engineering test, it was important place the USB's (which are described more in the Implementation section) in strategic areas. The USB's were placed in the most popular places on campus where there was a higher chance to be picked up by a student, professor, or faculty member.

4 DESIGN

Figure 3 shows the high-level design of this project. The top level shows the number of tests performed. The second level shows the type of tests performed. The third level shows the specific tools used in testing. Finally, the last row shows the targets of each test.

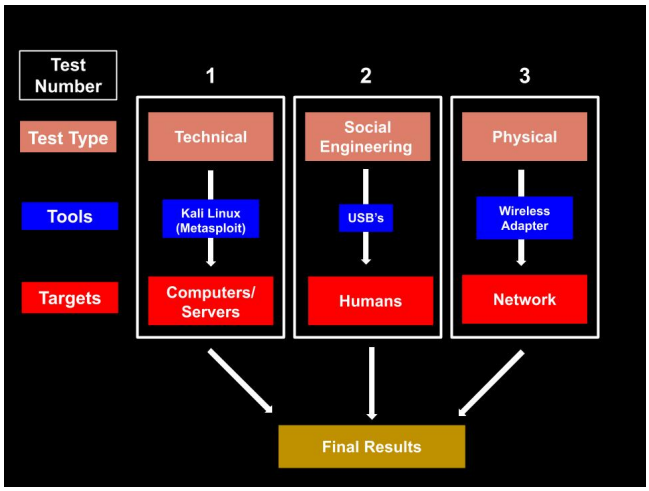

Figure 3: High-level implementation

4.1 Number of Tests

As shown in the first level of figure 3, there were three tests conducted for this project. Having three tests covered all aspects needed for a in-depth penetration test. Those tests included physical testing, social engineering, and technical testing.

4.2 Test Type

The tests conducted covered all areas of a system that a potential hacker could exploit to gain access to confidential information. Hacking through social engineering is now the most common type of way a hacker gains access to a system. Humans are flawed and can be easily manipulated in many ways. Thus, this leaves an avenue for exploitation.

Technical hacking is accomplished by using software to gain access to servers, routers, and other hardware. Many times, scripts called payloads or exploits are loaded and injected into the targets system. However, host software and hardware security has become much stronger. It is now much harder to exploit a machine through technical aspects than it is by gaining someones password and

entering the system in this way. This is why social engineering has become so popular.

Dimkov et al. stated that social engineering is an extremely powerful way to obtain very salient information [6]. Social engineering is manipulating the target into giving away this information when they normally would not do so. Someone is being psychologically tricked, whether it be in person or through something such as email phishing. Hackers have found much success with this type of hack, thus, it is important that it is talked about and understood. This attack can happen anywhere, at anytime, and in almost any setting. College settings such as cafe's, wellness centers, accounting offices, and hangout areas are all viable and easy options for a potential attacker.

Physical hacking can be slightly more difficult than social engineering but can be easier than technical hacking depending on the environment. College campus' present a particularly interesting case for being physically hacked. Internal connections such as WiFi and ethernet ports are widely available around college campus'. It is very easy for a hacker to access these ports or WiFi instantly. Security measures will be strained and tested to see how easily someone from off campus could access the network.

It is important to note that hackers do not simply use one of the three types of hacks. Any of the three can be used in combination with the others to produce a more thorough breach of security. It is common that social engineering is used to conduct a physical penetration test.

4.3 Tools

Separate tools were used for each phase of this test. For two of the three tests, physical hardware was used while the third only requires software.

For the technical attack, we used a virtual machine with the Kali Linux operating system installed. Kali Linux is one of the most popular OS's to use for security purposes. It has built in tools for testing, such as Metasploit framework, which was our software of choice for conducting this test. Mudge stated that there are over 600 different exploits for operating systems [14]. There are also different password guessing attacks. These could be extremely common on college campuses as students are very basic with password creation.

For social engineering, a piece of hardware called a USB Rubber Ducky can be used to hack into someones computer. As stated by Cannols and Ghafarian, any device claiming to have a keyboard HID (Human Interface Device) will be accepted by modern operating systems [3]. This USB stick has a built in micro-SD card as well as a micro controller that mimic's the operation of a keyboard. Payloads can be written and loaded onto the SD card and then injected into the targets device. There are many different commands that can be written in this language known as Ducky Script. Figure 4 shows an example of what this language looks like. With these, computers can be commanded to do different things. There are multiple demonstrations by the GitHub user hak5darren [2]. So, if a student, professor, or faculty worker were to find and plug this device into their computer, their computer could potently be hacked. Our social engineering experiment did not use this device, but did simulate something similar by placing random USB's around campus and seeing how willing some are to use these as their own. An image of the USB Rubber Ducky is shown in figure 5. These devices look exactly like the standard USB's we used, which can be seen in figure 10.

The physical aspect of this test was less detailed and strenuous. Our plan was to see if any information could be gained using a Wireless adapter, some software within Kali Linux, as well as trying to break the WPA2 encryption used by the school.

4.4 Targets

Given that there are three tests, there was three different targets. The targets were catered to the type of test to be performed. For the technical attacks, the targets were servers. It could be servers hosted and maintained by the IT office of the college or it could be servers hosted by a sub-domain of students and faculty. In this case we used Virtual Box to make testing more producible. Multiple departments could have servers they maintain so it would be important for a tester or attacker to know what information they are going for when they are trying pick their target.

For social engineering, a college or university presents the most broad category of targets. A hacker or pen tester could target students of the campus, professors, faculty, or even visitors of the campus. Again, this needed to be taken into careful consideration based on the information you are trying to gain. If you are just trying to see how easily you could get onto the campus WiFi, then all choices mentioned above could be a viable option. If a hacker wanted to gain information on a student, then targeting students would be the obvious choice. However, you also target campus employees who maintain student information such as the accounting office, bursars office, etc.

For physical hacking in terms of this project, the target was the campus network as a whole. Once someone gains access to the campus network, it makes it much easier find important information. The network was a very broad target but it had the most options once you have gained access.

4.5 Results

The results are the final phase of any type of penetration test. Whether they include all three types of testing or not, a results section is needed to report any findings. The results will be a report that shows any vulnerabilities found during the test. This is arguably the most important piece of a pen test. Campus' need to understand what the flaws are of their current system. Once they receive this document, it is up to them to make the changes to ensure better security.

5 IMPLEMENTATION

In order to test the overall security of the systems, three separate plans were executed from different angles. Technical testing tested the software of our systems as well as the firewall. Social engineering tested the knowledge and vulnerability of individuals on campus. Physical testing tested how easy an outsider can access the internal network.

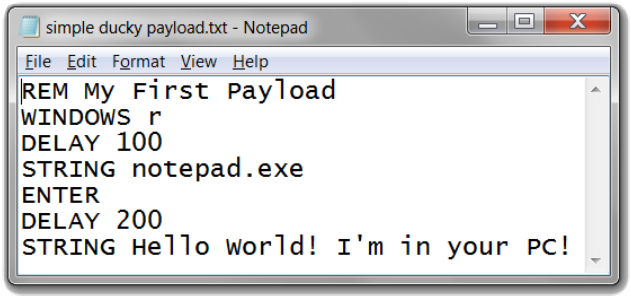

Figure 4: An example of the Ducky Script language

Figure 5: A USB that can hack into computers

5.1 Physical testing using a wireless adapter

This test was run to see if a laptop, with its WiFi turned off, could easily access the network using a wireless adapter. A virtual machine (VM) with Kali Linux installed was used for this testing. A wireless adapter, which allows monitor mode to be enabled, was required to enable the adapter to see all networks and clients attempting to connect. Local commands were then used to direct those connections into a file. That file was loaded into Wire Shark to see if any data slipped through the encryption of the network. In a perfect world, no data coming through the college's WiFi should be readable doing this type of test. See below if this was the case.

5.2 Exploitation using Social Engineering

For the Social Engineering experiment, a pack of 10 standard USB's were needed. Different labels such as "FBI", "CIA", "Richmond Police", "Earlham Financials", etc. were given to each USB. These labels and the locations can be found in table 1. To better show where on campus these places are, refer to figure 6. The labeling strategy was done to try and entice those on campus to pick-up and plug-in these USB's into their personal computer. It is important to note that devices that look extremely similar to standard USB's can be used to hack into someones computer. Nothing malicious was placed on the USB's. There was a simple text file which debriefed the subject of the purpose of this project. Following this, there was a link to a Google survey which allowed the users to express their knowledge of social engineering and cyber security. No personal information was asked on this survey.

Table 1: Number, Location, and Label of USB's placed around campus

Number	Label	Location Dropped
1	CIA	Cafe 1847
2	FBI	May-Crossen Commons
3	Earlham Financials	CST
4	Top Secret Info.	Wellness Center
5	ECCS	CST
6	Earlham Athletics Dept.	Wellness Center
7	Health Services	LBC
8	Financial Aid	LBC
9	Richmond Police	Library
10	*No label*	Carpenter Hall

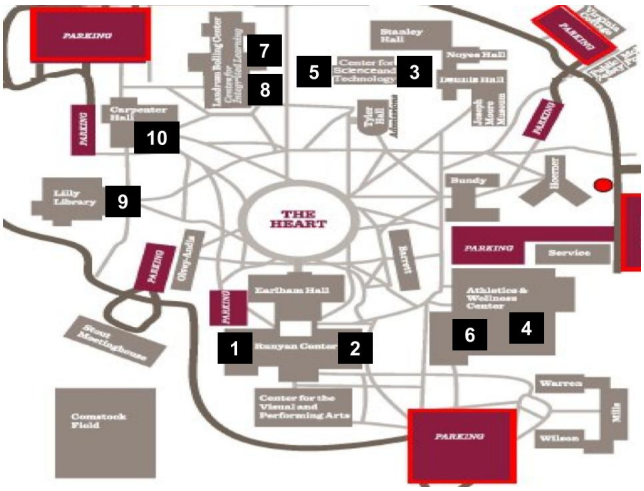

Figure 6: Placement of the USB's around the campus of Earlham College

5.3 Metasploit Exploitation

One test box was be set-up for this experiment. A Metasploit server was created within Virtual Box and connected to the same NAT-Network as the Kali server. This was done in order to portray an attacker being on the same network as its target (i.e. already connected to campus WiFi). A virtual environment was the most efficient and ethical way to do this as no personal data was on the

Metasploit server and everything was cleared at the conclusion of this test. We did not have a firewall as another level of security but that could be the case when organizations try and replicate our work.

6 RESULTS

Each test of this proposal was evaluated based on the results gained during testing. All three tests had an element of a binary answer of success or failure but it was deeper than that. For example, just because we did not successfully penetrate campus servers does not mean the project was a failure. This simply means the integrity of the system was as it should be, which was ultimately what we hoped to see. The results from each test are found below.

6.1 Physical Testing

This testing provided great feedback on the security of the ECOpen and ECSecure networks that are available on the campus of Earlham College. As stated above, a VM with Kali Linux was needed to run appropriate commands to test the networks. A wireless adapter which allows monitor mode was also necessary. This adapter had the ability to listen on both 5 GHz and 2.4 GHz frequencies. This was not a necessity but both networks ran multiple channels on these two frequencies, hence clients send data on both. To begin, the media access control (MAC) address was changed for the adapter. This was not needed, but is done for spoofing and other purposes. The adapter then needed to be placed into monitor mode to be able to see all traffic for both networks. Figure 7 shows the commands needed to change the MAC address as well as changing it to monitor mode.

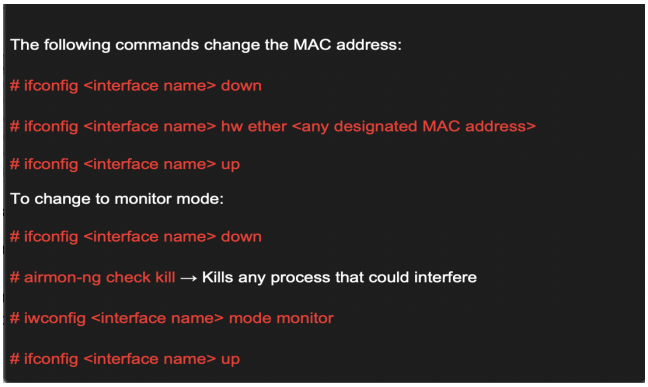

Figure 7: Commands used in Kali Linux for the wireless adapter

After changing the mode and MAC address of the adapter, it was time to use Kali Linux's packet sniffer. This command allowed the adapter to listen to traffic on the 2.4 GHz frequency by default. It is the most basic command. There are many parameters that can be passed to allow for better results. The command is called "airodump-ng" which is followed by the interface name of the adapter. The results of this command can be analyzed in figure 8.

This command showed the two available networks, ECOpen and ECSecure, the different channels associated with each BSSID (this

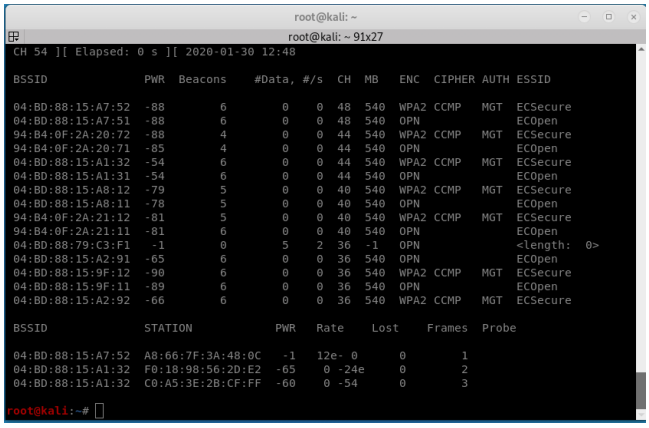

Figure 8: Basic result from airodump-ng command

is just the MAC Address), the encryption used by each network, as well as other important data. Upon finding these BSSID's, it was necessary to use the airodump-ng command on a specific one. This then allowed the adapter to listen and capture data being transferred on a specific channel of a specific network by a specific device. The command can be found in figure 9.

Figure 9: Command to sniff on a specific BSSID

When this command was used, it was necessary to specify which BSSID to sniff on, the channel it was on, where the output needed to be written to, and finally the interface name of the wireless adapter. Once that command was executed, it could be stopped at any point and the results had to be viewed on WireShark. Simply typing WireShark on the command line allowed the application to be started. It was then necessary to open the file in WireShark and see if we could view any personal data that was captured by the airodump-ng command. Again, this file showed communication between a specific network and multiple clients (phones, computers, etc.) If we look back at figure 8, all ECSecure networks were encrypted with WPA2 encryption. Thus, when viewing the file in WireShark, we could not see any personal information due to the encryption. WPA2 is the most common type of encryption method used by devices today.

It was then time to try and see if the ECOpen network could produce different results than the ECSecure network. The same process as explained above was completed except different parameters were filled (different BSSID, channels, etc). If we look at figure 8, we see that ECOpen networks say "OPN" under the encryption section. From this, one may think that ECOpen does not use any encryption at all thus this would allow us to see data caught by the packet sniffer. However, this is incorrect. No data could be seen in WireShark. After running this experiment, we found evidence that

ECOpen also uses WPA2 encryption. The "OPN" signifies guests of the campus can connect to this network without a username and password. After researching the network more, guests register on this network and create their own username and password which is valid for a limited amount of time. Their data is still encrypted through WPA2 encryption which did not allow any information to be visible under WireShark. This is a very good security feature that the campus uses.

Finally, we then tried to break this encryption. We opened two separate terminals. In the first, we ran the same command found in figure 9. As this continued to run, in the second terminal we ran a deauthentication attack. This caused an interuption in the connection between the router and user. As the router tried to connect back to the client, an ID known as a Handshake is shared between the two. The airodump command attempts to catch this handshake as it is what is needed to break the WPA2 encryption. Unfortunately, we were unable to catch the Handshake key after multiple attempts. Network settings kept this key hidden as it should. After multiple attempt, we broke neither the ECSecure or ECOpen WPA2 encryption.

6.2 Social Engineering Test

We want to first reiterate that this part of the project was reviewed and approved by the Institutional Review Board at Earlham College. The social engineering test was perhaps the most important of the three. Given how advanced software has become, this has made technical and physical hacking much more difficult. In recent years, attackers have turned to social engineering to gain access to credential information. This deception has come in many ways, with the most common being email phishing. There are many different ways to develop a social engineering experiment. We chose to target the curiosity of young adults by placing USB's across campus. We used USB's similar to the one in figure 10. These USB's had various labels. These labels, as well where they were placed, can be found in table 1 and figure 6.

For each USB, we asked the user to fill out a Google survey to describe their knowledge of cyber security and social engineering. The USB's were left on campus for about a month. At the conclusion of the test, we only received one response on the survey. Now, this could mean a couple of things. First, it could mean that the other nine USB's were never picked up and opened by anyone on campus, or secondly, people could have found the other USB's, opened them, but did not fill out the survey. It was reported to us that some of the USB's had been turned into the front desk of some of the buildings. Although a student did not open it, these were opened by the clerk working the front desk. This could still allow someone to gain to confidential information. It is important to note that the one survey we did receive was from the Public Safety officer of the school. The officer contacted us and wanted to make sure this was our USB and was for research purposes only. The officer also explained that the USB was turned into them by a student. The officer then went on to open the USB on a computer that was part of an air-gapped network. This meant that no confidential information could have been taken. We were happy to hear this as these are the steps that should be followed with a situation like this. We also know that this is not what happens with every situation. Tests like these are done to show the lack of awareness of social engineering and cyber attacks. After the conclusion of this test, it would be of the upmost importance for officials of the school to require training of staff in order to raise awareness of these issues. This is our recommendation to the college.

Figure 10: USB's used in on-campus social engineering experiment

6.3 Technical Testing

This testing allowed us to see some of the possible ways hackers use certain software to try and gain remote access to computers within the same network. Our main tool was the Metasploit software, which came pre-installed on Kali Linux, however, we also used a network tool called Zenmap. This tool is the user-interface (UI) version of the terminal program NMAP. This tool was mainly used for port scanning on the target machine, which again was the Metasploit server on Virtual Box.

Our first priority was to conduct a scan with Zenmap to see which ports had services running on them. Zenmap was extremely easy to use, all you have to do is type the IP address' you would like to scan and the results pop up (It also gives the NMAP command you would run in the terminal!).

Here, it was not necessary to use Metasploit on Kali to gain remote access. We simply looked through the open ports on the machine, did some research on the Internet and was able to find a remote login service that had a known bug and was running on port 512. After installing that service on the Kali machine, we were able to use the rlogin command to gain remote access of the Metasploit machine.

This is the rlogin command that was used:

```
$ rlogin -l root <Target IP Address>
```

It is important to note that this may not work for every computer. Systems are engineered differently with different ports running and sometimes different services running on those ports. The point of what was demonstrated is that administrators need to monitor what ports and services are running on their machines and keep up to date with any bugs that could have been discovered with those

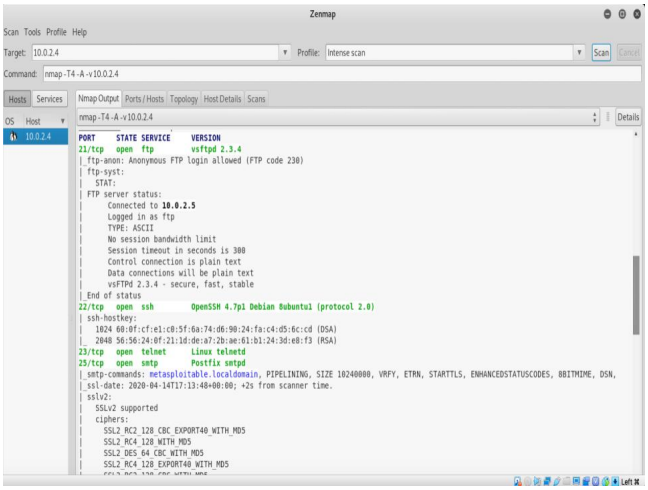

Figure 11: Results from using Zenmap for port scanning

services. This will help keep their machines and data secure from hackers.

We then dove into using the Metasploit software to test the security of the Metasploit server. Again, using Zenmap, we were able search the services and find a specific service that had a known backdoor exploit that is common to use in Metasploit. After setting some specific options for this exploit, we were able to gain root access of the Metasploit server. New exploits are found everyday and it is extremely important to keep systems up-to-date for security fixes.

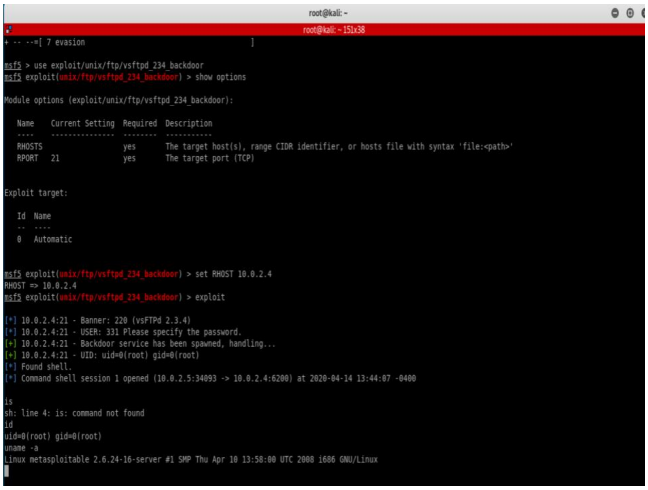

Figure 12: Gaining access using Metasploit software on Kali

Based on the results gained from this testing, we have some recommendations. We believe it is important for system administrators to frequently update software. Software updates almost always include security fixes. In terms of firewall rules and open ports, we believe it best to deny all connections and close all ports (except those open by default for services running on the machine). This allows you to have control what comes in and out. Open these features only as needed. These are two basic recommendations that are a good first step to solidifying the integrity of your systems.

6.4 Constraints and Challenges

We developed a three-pronged approach to conducting a penetration test. We wanted this to have high scalability and reproducibility. The test for social engineering was the easiest to conduct. There were some parts of the process that took time, such as approval from the Earlham IRB. However, once approved, it was smooth sailing. The technical and physical testing presented more challenges.

For physical testing, it was first necessary to understand Earl ham's networking infrastructure to make it easier to try and pinpoint its vulnerabilities. This required talking to administrators in the IT office for an extended period of time. Once we had a good understanding of the network, we had to find a good approach to using Kali Linux to exploit the network. Once we found commands that were appropriate, it was smooth sailing from there.

Technical testing provided bigger challenges. This style of testing required a lot of trial and error to find the right port to use and then finding an appropriate exploit under Kali Linux's Metasploit Framework. It was not the commands under Metasploit that were difficult but rather the long amount of time searching for the right backdoor. Given the longevity of the work, the outcome was extremely rewarding.

7 FUTURE WORK

We wanted to conduct an experiment that could be easily reproduced by other colleges and universities as well as other businesses. We believe that security is hugely important and it is necessary that businesses be able to test the integrity of their network to protect the data of their users without spending a ton of money. We did not want to create a project that could only be used by a small institution but one that could be scaled to any school or business, no matter their size. Locally, we could see institutions such as IU-East and Ivy Tech benefiting from our research and, depending upon success, moving to other businesses in the Richmond community. These campuses are somewhat similar in size and contain much of the same student and faculty data that is important to protect. Knowing how to test the security of your network could save you money. Penetration tests are expensive when done by a third-party so we wanted to design a way to do this using open-source software and affordable hardware.

ACKNOWLEDGMENTS

I would like to thank Professor Xunfei Jiang, Professor Charlie Peck, Mr. Craig Earley, Mr. Byron Roosa, and Mr. Jason Elliot for providing varying levels of support during the development of the project in the fall of 2019 and its implementation in the spring of 2020.

REFERENCES

[1] [n.d.]. 10 cyber security facts and statistics for 2018. https://us.norton.com/internetsecurity-emerging-threats-10-facts-about-todays-cybersecurity-landscape-that-you-should-know.html

[2] [n.d.]. GitHub - hak5darren/USB-Rubber-Ducky. https://github.com/hak5darren/USB-Rubber-Ducky

[3] Benjamin Cannols and Ahmad Ghafarian. [n.d.]. *Hacking Experiment by Using USB Rubber Ducky Scripting*. Technical Report. http://example.co

[4] Rachna Dhamija, J. D. Tygar, and Marti Hearst. 2006. Why Phishing Works. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, New York, NY, USA, 581–590. https://doi.org/10.1145/1124772.1124861

[5] Trajce Dimkov, Wolter Pieters, and Pieter Hartel. 2011. Training Students to Steal: A Practical Assignment in Computer Security Education. In *Proceedings of the 42Nd ACM Technical Symposium on Computer Science Education (SIGCSE '11)*. ACM, New York, NY, USA, 21–26. https://doi.org/10.1145/1953163.1953175

[6] Trajce Dimkov, André van Cleeff, Wolter Pieters, and Pieter Hartel. 2010. Two Methodologies for Physical Penetration Testing Using Social Engineering. In *Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC '10)*. ACM, New York, NY, USA, 399–408. https://doi.org/10.1145/1920261.1920319

[7] Lee Epling, Brandon Hinkel, and Yi Hu. 2015. Penetration Testing in a Box. In *Proceedings of the 2015 Information Security Curriculum Development Conference (InfoSec '15)*. ACM, New York, NY, USA, Article 6, 4 pages. https://doi.org/10.1145/2885990.2885996

[8] Teresa Guarda, Walter Orozco, Maria Fernanda Augusto, Giovanna Morillo, Silvia Arévalo Navarrete, and Filipe Mota Pinto. 2016. Penetration Testing on Virtual Environments. In *Proceedings of the 4th International Conference on Information and Network Security (ICINS '16)*. ACM, New York, NY, USA, 9–12. https://doi.org/10.1145/3026724.3026728

[9] Amanda M. Holland-Minkley. 2006. Cyberattacks: A Lab-based Introduction to Computer Security. In *Proceedings of the 7th Conference on Information Technology Education (SIGITE '06)*. ACM, New York, NY, USA, 39–46. https://doi.org/10.1145/1168812.1168825

[10] Markus Jakobsson and Jacob Ratkiewicz. 2006. Designing Ethical Phishing Experiments: A Study of (ROT13) rOnl Query Features. In *Proceedings of the 15th International Conference on World Wide Web (WWW '06)*. ACM, New York, NY, USA, 513–522. https://doi.org/10.1145/1135777.1135853

[11] Ecir Uğur Küçüksille, Mehmet Ali Yalç, and Samet Ganal. 2015. Developing a Penetration Test Methodology in Ensuring Router Security and Testing It in a Virtual Laboratory. In *Proceedings of the 8th International Conference on Security of Information and Networks (SIN '15)*. ACM, New York, NY, USA, 189–195. https://doi.org/10.1145/2799979.2799989

[12] Anjana V. Kumar, Kalpa Vishnani, and K. Vinay Kumar. 2012. Split Personality Malware Detection and Defeating in Popular Virtual Machines. In *Proceedings of the Fifth International Conference on Security of Information and Networks (SIN '12)*. ACM, New York, NY, USA, 20–26. https://doi.org/10.1145/2388576.2388578

[13] Mary Micco and Hart Rossman. 2002. Building a Cyberwar Lab: Lessons Learned: Teaching Cybersecurity Principles to Undergraduates. *SIGCSE Bull.* 34, 1 (Feb. 2002), 23–27. https://doi.org/10.1145/563517.563349

[14] Raphael Mudge. 2011. Live-fire Security Testing with Armitage and Metasploit. *Linux J.* 2011, 205, Article 1 (May 2011). http://dl.acm.org/citation.cfm?id=1972989.1972990

[15] Syed A. Saleem. 2006. Ethical Hacking As a Risk Management Technique. In *Proceedings of the 3rd Annual Conference on Information Security Curriculum Development (InfoSecCD '06)*. ACM, New York, NY, USA, 201–203. https://doi.org/10.1145/1231047.1231089

[16] Douglas P. Twitchell. 2006. Social Engineering in Information Assurance Curricula. In *Proceedings of the 3rd Annual Conference on Information Security Curriculum Development (InfoSecCD '06)*. ACM, New York, NY, USA, 191–193. https://doi.org/10.1145/1231047.1231062