Creation of Pentesting Labs

By

Kyle Barta

Submitted to
The Faculty of the Department of Information Technology
In Partial Fulfillment of the Requirements for
The Degree of Bachelor of Science
In Information Technology

University of Cincinnati
Department of Information Technology
College of Education, Criminal Justice, and Human Services

April 2013
Creation of Pentesting Labs

By

Kyle Barta

Submitted to
the Faculty of the Department of Information Technology
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Technology

© Copyright 2013 Kyle Barta

The author grants to the Department of Information Technology permission
to reproduce and distribute copies of this document in whole or in part.

Kyle Barta _______________________________ 4/25/13
Kyle Barta
Date

Russell E. McMahon ______________________ 4/16/2013
Professor Russell McMahon
Date
# Table of Contents

Abstract .......................................................................................................................... 1

1.0 Project Description .................................................................................................. 2

1.1 Problem Statement ................................................................................................. 2

1.2 Proposed Solution .................................................................................................... 2

1.3 User Profile ........................................................................................................... 3

2.0 Design Protocols ..................................................................................................... 3

2.1 Technical Elements ............................................................................................... 3

2.1.1 Networking ....................................................................................................... 4

2.1.2 Security ............................................................................................................ 4

2.1.3 Database .......................................................................................................... 4

2.1.4 Linux Administration ....................................................................................... 4

2.1.5 Website Design ............................................................................................... 5

2.2 Solution Details ..................................................................................................... 5

3.0 Deliverables ........................................................................................................... 6

4.0 Project Planning ..................................................................................................... 6

4.1 Project Timeline ..................................................................................................... 6

4.2 Project Hardware .................................................................................................. 8

4.3 Project Software .................................................................................................. 8

4.4 Budget .................................................................................................................. 9

5.0 Content ................................................................................................................ 10

5.1 Lab 1 – Nessus and Nmap .................................................................................... 10

5.2 Lab 2 – Introduction to Metasploit ....................................................................... 13

5.3 Lab 3 – ARP Spoofing, ARP Poisoning, and Sniffing ........................................... 15
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Spring Schedule</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>Spring Schedule and Gantt Chart</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>Budget Table</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>Installing Nessus and adding an admin user to it</td>
<td>11</td>
</tr>
<tr>
<td>5</td>
<td>Running a SYN Scan using Nmap</td>
<td>12</td>
</tr>
<tr>
<td>6</td>
<td>Examining Policies in Nessus</td>
<td>13</td>
</tr>
<tr>
<td>7</td>
<td>Examining the Vulnerabilities in a Windows XP System on the Network</td>
<td>14</td>
</tr>
<tr>
<td>8</td>
<td>Setting the payload for our chosen exploit.</td>
<td>14</td>
</tr>
<tr>
<td>9</td>
<td>Exploiting our Windows XP machine</td>
<td>15</td>
</tr>
<tr>
<td>10</td>
<td>ARP Spoofing in Backtrack</td>
<td>16</td>
</tr>
<tr>
<td>11</td>
<td>ARP Poisoning in Ettercap</td>
<td>16</td>
</tr>
<tr>
<td>12</td>
<td>Sniffing with Wireshark</td>
<td>17</td>
</tr>
<tr>
<td>13</td>
<td>Sniffing with Dsniff</td>
<td>18</td>
</tr>
<tr>
<td>14</td>
<td>Grabbing Banner with Netcat</td>
<td>19</td>
</tr>
<tr>
<td>15</td>
<td>Placing our netcat backdoor</td>
<td>20</td>
</tr>
<tr>
<td>16</td>
<td>Editing the registry to make our backdoor open on startup</td>
<td>20</td>
</tr>
<tr>
<td>17</td>
<td>Accessing the Windows XP system through our backdoor</td>
<td>21</td>
</tr>
<tr>
<td>18</td>
<td>Obtaining the SAM and System file</td>
<td>22</td>
</tr>
<tr>
<td>19</td>
<td>Cracking Password hashes with John the Ripper</td>
<td>23</td>
</tr>
<tr>
<td>20</td>
<td>Cracking password hashes using Ophcrack and rainbow tables</td>
<td>23</td>
</tr>
<tr>
<td>21</td>
<td>Conducting a brute force attack using Hydra</td>
<td>24</td>
</tr>
<tr>
<td>22</td>
<td>Using the Social Engineering Toolkit to spoof a website</td>
<td>25</td>
</tr>
<tr>
<td>23</td>
<td>Spoofing DNS using Ettercap</td>
<td>26</td>
</tr>
<tr>
<td>24</td>
<td>Testing our spoofed website and exploit</td>
<td>26</td>
</tr>
</tbody>
</table>
Figure 25 - Testing for SQL Injection vulnerability ................................................................. 27

Figure 26 - Finding the current database using Sqlmap ............................................................ 28

Figure 27 - Web Vulnerability Scanning with Nikto .................................................................... 29

Figure 28 - Our Cross Site Scripting Example ........................................................................... 30

Figure 29 - Going through the solution to the final lab .............................................................. 31
Abstract

Penetration Testing, or pentesting, is an extremely interesting topic. However, there aren’t many options out there as far as a starting place for students and beginners. I quickly realized this when I became interested in the topic. Frequently in my searching I’d found that professionals always point to the BackTrack Linux distribution as an excellent pentesting suite which much to offer. After booting up BackTrack the first time, I was really overwhelmed! I noticed there are a ton of different tools available, and I had no idea what any of them were. Further, there isn’t really a starting point or any direction regarding any of these tools or what they’re used for. As a result, I have created some pentesting labs that go over the usage of some of the most popular tools in BackTrack. These labs not only introduce popular tools to the user, but they also challenge the user to use the tools introduced in actual scenarios. This serves as an excellent gateway and starting point for anyone interested in pentesting and information security. What better way to learn the tools than using them yourself?
1.0 Project Description

1.1 Problem Statement

The importance of Cyber Security has never been higher. The financial sector of the business world has seen a drastic increase in cyber-attack in the past six years, and it’s expected to continue to grow. These attacks often involve corporate account takeovers. In September 2011, there were 400 cases being investigated by the FBI in the area of corporate account takeovers alone. The total attempted theft of these cases was $255 million, and there was an actual loss of $85 million. This is merely in one sector of business (Snow).

If IT Students want to get in to the important and growing field of Information Security, they need a means of learning how to use the pentesting tools in BackTrack. The tools included in BackTrack are widely regarded as the most important to a pentester. Additionally, I’ve found that in any area of Information Technology, the best way to learn is to get your hands dirty and use the tools given to you. It can be overwhelming when you first load up BackTrack if you aren’t sure where to start, so this project gives students that starting point and guides them through how to use the basic tools individually and together.

1.2 Proposed Solution

The proposed solution for this problem is to create a series of 10 labs that takes the user through 10 different pentesting Labs using the pentesting Linux distribution BackTrack. This will
serve as a bridge for students interested in Information Security, allowing them to know many of the techniques used in pentesting.

1.3 User Profile

The users for this project include IT Students, and Security Certification Track Students. These students need experience using VMWare, Windows, and especially Linux since the primary work is done in BackTrack 5, a Linux distribution. Any background in using BackTrack or any of the tools in the labs certainly make them easier, but is not necessary. IT Students should find moderate difficulty in the labs, particularly if they’re not familiar with the tools at all. However, given that video tutorials will be made available, I do not foresee this as a problem. These labs may be used more than once if a student forgets how to use a certain tool, but likely will just be used once or twice to learn the basics before moving on to a more in-depth usage of each individual tool. A professor, however, may use these in their classes and as such they have a higher frequency of use there.

2.0 Design Protocols

2.1 Technical Elements

The technical elements of this project include networking, information security, database, and Linux administration.
2.1.1 Networking

In this project, networking was necessary to address the different Virtual Machines. In addition, some network configuration was required including the networking of a database server and web server. The virtual machines must be accessible by the BackTrack virtual machine in order to complete the labs correctly.

2.1.2 Information Security

In this project, information security was used throughout. All of the techniques taught in the various labs are information security based. This includes knowledge of how to scan for ports, scan for vulnerabilities, exploit machines, maintain access to systems, crack passwords offline and online, and exploit websites. Each of these techniques are demonstrated and illustrated.

2.1.3 Database

For some of the labs, a knowledge of how to create and edit and maintain a MySQL database was necessary. There was a Linux server that housed a MySQL database for a couple of the labs that was used as an example.

2.1.4 Linux Administration

In this project, an extensive knowledge of Linux was displayed. BackTrack itself is a Linux distribution, so understanding how to navigate through Linux and accomplish tasks was
necessary. Additionally, an Ubuntu server was used for many of the examples. On this server, services such as SSH, FTP, Apache, and MySQL were all installed at various points throughout the labs.

2.1.5 Website Design

Basic HTML was used to create some placeholder websites used in a few of the examples for the labs.

2.2.0 Solution Details

My goals when beginning this project were to provide a gateway to the pentesting world, or a starting point for students interested in pentesting. I wanted this project to enable students to understand some key concepts in pentesting, while teaching them to use some of the most popular tools in BackTrack. Certainly in pentesting you can always go deeper and get more complex, so I had to limit this project to be a hands-on bridge between someone that is already a networking student and someone that is on their way to becoming an Information Security professional.

As a result of these goals, I’ve created 10 pentesting labs using BackTrack 5, Ubuntu, and Windows XP virtual machines on VMWare Workstations. Each lab includes instructions for the student that will help guide them, as well as the VMWare files necessary for completion. Additionally, video tutorials were created to assist the user in the labs in case they run into problems. In the final lab, answers are necessary as it is more of a test than a demonstration. As such, these answers have been provided and the video tutorial is instead a video solution. The
result is that these labs will give a student confidence in the usage of some of the most popular tools included in the BackTrack distribution of Linux.

3.0 Deliverables

The deliverables of this project include 10 labs, each going over different tools and techniques used in pentesting. Each will come with a few things:

- The Lab Files necessary for lab completion. These pre-configured virtual machines not only streamline each lab by removing the set up process, they frequently offer a sense of mystery since the user didn’t set them up themselves. This allows the user to experience each lab fresh.
- The Lab Instructions. These are simply the walkthrough instructions for the lab, taking you through the main content.
- The Video tutorial. This is helpful in that if a user gets confused, they can reference the video to see exactly what I did to produce the desirable results of the lab.

4.0 Project Planning

4.1 Project Timeline

My time spent on this project this semester varied mostly between research, lab planning, lab testing, and video creation. The table below as well as the following Gantt Chart explain my time spent this semester on the project.
<table>
<thead>
<tr>
<th>Task Name</th>
<th>Duration</th>
<th>Start</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finalize Labs 2-4</td>
<td>6 days</td>
<td>Mon 1/21/13</td>
<td>Mon 1/28/13</td>
</tr>
<tr>
<td>Send out labs to be tested/critiqued</td>
<td>9 days</td>
<td>Tue 1/22/13</td>
<td>Fri 2/1/13</td>
</tr>
<tr>
<td>Creation of Videos for Labs 2, 3</td>
<td>5 days</td>
<td>Mon 1/28/13</td>
<td>Fri 2/1/13</td>
</tr>
<tr>
<td>Preparation for Demonstration</td>
<td>2 days</td>
<td>Fri 2/1/13</td>
<td>Sun 2/3/13</td>
</tr>
<tr>
<td>Creation of Labs 5-6</td>
<td>5 days</td>
<td>Tue 2/5/13</td>
<td>Sun 2/10/13</td>
</tr>
<tr>
<td>Testing Plan/Report</td>
<td>2 days</td>
<td>Sun 2/10/13</td>
<td>Mon 2/11/13</td>
</tr>
<tr>
<td>Creation of Videos for Labs 4, 5</td>
<td>5 days</td>
<td>Mon 2/11/13</td>
<td>Fri 2/15/13</td>
</tr>
<tr>
<td>Brainstorm/Evaluation of Final 4 Labs</td>
<td>2 days</td>
<td>Fri 2/15/13</td>
<td>Sun 2/17/13</td>
</tr>
<tr>
<td>Finalize Abstract</td>
<td>2 days</td>
<td>Sun 2/17/13</td>
<td>Mon 2/18/13</td>
</tr>
<tr>
<td>Creation of Labs 7-8</td>
<td>5 days</td>
<td>Mon 2/18/13</td>
<td>Fri 2/22/13</td>
</tr>
<tr>
<td>Creation of Videos for Labs 6-7</td>
<td>2 days</td>
<td>Fri 2/22/13</td>
<td>Sun 2/24/13</td>
</tr>
<tr>
<td>Creation of Final Labs 9-10</td>
<td>5 days</td>
<td>Mon 2/25/13</td>
<td>Fri 3/1/13</td>
</tr>
<tr>
<td>Creation of Videos for Labs 8-10</td>
<td>2 days</td>
<td>Fri 3/1/13</td>
<td>Sun 3/3/13</td>
</tr>
<tr>
<td>Finalize the Project</td>
<td>7 days</td>
<td>Fri 3/1/13</td>
<td>Sat 3/9/13</td>
</tr>
<tr>
<td>Tech Expo Poster</td>
<td>7 days</td>
<td>Fri 3/1/13</td>
<td>Mon 3/11/13</td>
</tr>
<tr>
<td>Prepare for Presentation</td>
<td>10 days</td>
<td>Tue 3/12/13</td>
<td>Mon 3/25/13</td>
</tr>
<tr>
<td>Prepare for Tech Expo</td>
<td>6 days</td>
<td>Mon 3/25/13</td>
<td>Mon 4/1/13</td>
</tr>
</tbody>
</table>

**Figure 1** – Spring Schedule

**Figure 2** – Spring Schedule and Gantt Chart
4.2 Project Hardware

The PC used to house my Virtual Machines used an intel i5-3570k Quad Core Processor and 8GB of RAM. This was able to handle my virtual machines with ease. However, the lab machines are also Quad Core machines and could easily handle 3 VMs concurrently. The only other hardware used was a 500 GB USB 3.0 external hard drive to house all of the deliverables.

4.3 Project Software

This project used VMWare Workstation in order to create a lab environment using 3 virtual machines. The virtual machines used operating systems such as Windows XP, Ubuntu, and BackTrack 5 R3, and Badstore. On the Ubuntu server, Apache, MySQL, SSH, and FTP were all services that were installed. On the BackTrack machine, various tools were used such as Nessus, Nmap, John the Ripper, Ettercap, The Social Engineering Toolkit, Metasploit, Hydra, Ophcrack, and Nikto. Each of these are commonly used tools within the BackTrack suite. Additionally Badstore is an intentionally bad web server used to test certain website attack vectors such as SQL Injection and Cross-Site Scripting.

The most important thing to note about all of the software used is that as the labs progressed, these were being configured differently with varying levels of security. For one of the labs, for example, a Windows XP system is entirely unpatched and this allowed me to demonstrate some basics behind Metasploit.
4.4 Budget

The budget for this project was nearly free. Both BackTrack 5 and Ubuntu are free Linux distributions, and the Windows XP VM is available for use in the Networking lab. VMWare Workstation was used for the labs, which can cost around $250. However, the license is free for IT students at the University of Cincinnati. The resources I’ve used have been free either through research on the Web or through books available through Safari (a free license due to being a student at the University). The only cost was the cost to attend a CinPA meeting in Cincinnati ($5) as well as the cost of an external drive to house all of the data from my project ($99), and these costs weren’t absolutely necessary. In short, necessary costs were $0 but the project in total cost me $80.

<table>
<thead>
<tr>
<th>Item</th>
<th>Budgeted Cost</th>
<th>Estimated Actual Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMWare Workstation</td>
<td>$0</td>
<td>$250</td>
</tr>
<tr>
<td>BackTrack 5</td>
<td>$0</td>
<td>$0</td>
</tr>
<tr>
<td>Ubuntu</td>
<td>$0</td>
<td>$0</td>
</tr>
<tr>
<td>BadStore</td>
<td>$0</td>
<td>$0</td>
</tr>
<tr>
<td>Windows XP License</td>
<td>$0</td>
<td>$50</td>
</tr>
<tr>
<td>CinPA Attendance</td>
<td>$5</td>
<td>$5</td>
</tr>
<tr>
<td>External Hard Drive (500 GB)</td>
<td>$99</td>
<td>$99</td>
</tr>
<tr>
<td>Total</td>
<td>$104</td>
<td>$404</td>
</tr>
</tbody>
</table>

Figure 3 – Budget Table
5.0 Content

5.1 Lab 1 – Nessus and Nmap

Nessus is used in pentesting as a vulnerability scanner. This means you can essentially scan a target machine to see if it’s vulnerable or likely to be vulnerable to the different attack vectors Nessus knows.

Nmap is a very important port scanner used by pentesters. It allows you to see what ports are open, closed, or secured on a machine. Additionally, there are different scans used in Nmap that could gain you different results or offer you varying levels of stealth while performing the scan.

In this lab, the user is first instructed on how to install Nessus on a BackTrack5 R3 machine. This includes getting a license, installing it, and then making sure it works.
Next, the user is instructed on how to use Nmap to scan ports of systems on the network. It goes through how to use a SYN Scan, a TCP Null scan, and an idle scan (or zombie scan). Each of these simply offers different levels of stealth, and different results due to the manner in which the scan is performed. It also touches on the various options that Nmap offers, and explains the usefulness of each of these scans.
Finally, in this lab, the user is instructed on how to use Nessus to scan systems for vulnerability. Policies are discussed, as well as scanning a host and viewing/interpreting the results found.
5.2 Lab 2 – Introduction to Metasploit

Metasploit is a tool used to develop and execute exploit code against a target machine. It’s one of the primary tools used by pentesters and is very powerful in the hands of someone who knows how to use it effectively.

In this lab, the user is instructed on how to start up the Metasploit Framework console, as well as some of the basics of using a Metasploit exploit. This goes right in line with using Nessus, and in the lab the user is taught how to find exploits, use exploits, and set important parameters for exploits such as the payload. Finally, the user is taught how to execute an exploit on a system.
Figure 7 – Examining the Vulnerabilities in a Windows XP System on the Network

Figure 8 – Setting the payload for our chosen exploit
5.3 Lab 3 – ARP Spoofing, ARP Poisoning, and Sniffing

ARP Spoofing is essentially pretending you’re the victim machine so the router or switch sends you traffic meant for the victim. ARP Poisoning is similar in that it does the same thing, however, an ARP Poison usually denies the victim the traffic once we see it. And finally, sniffing is a technique used to passively watch traffic that’s being sent in and out on a network.

In this lab, the usage of ARP Spoofing is discussed. This is done manually through the command line. Additionally, ARP Poisoning is discussed. The ARP Poison is done through the Ettercap tool. Ettercap is a tool usually used for sniffing or performing man-in-the-middle attacks (which is exactly what it sounds like).
Figure 10 – ARP Spoofing in Backtrack

Figure 11 – ARP Poisoning using Ettercap
Later, sniffing is discussed. This is a useful passive technique used by hackers to gain important information sometimes including user credentials if a service is unsecure. Sniffing is discussed by using Wireshark and Dsniff.

Figure 12 – Sniffing with Wireshark
5.4 Lab 4 – Netcat, and Maintaining Access

Netcat is a tool used for grabbing banner, and creating a backdoor. Grabbing banner is simply obtaining the application version of a service used on a server or workstation. A backdoor is done in order to maintain access to the system. It allows the system to respond when you try to access it, and is a key tool for gaining information over a period of time.
In this lab, the user is taught how to use netcat. It involves each of the processes above in detail, and will ultimately allow you to connect to the victim computer through a command line even if it’s no longer exploited or if they patched a vulnerability you previously could exploit. Additionally, it shows you how to upload files to an exploited machine using Metasploit in order to place the netcat backdoor.
Figure 15 – Placing our netcat backdoor

Figure 16 – Editing the registry to make our backdoor open on startup
Cracking passwords simply involves gaining the password of a user without their permission. This can be done in numerous ways, but in this lab it’s done by cracking a hashed password file. On a Windows machine, a SAM file contains all of the usernames and passwords on the local computer. However, it cannot be accessed when the system is live, and even if you did access it they’re stored in a hashed or encrypted format.
Figure 18 – Obtaining the SAM and system file

This lab goes over how to obtain a SAM file from a system when you have physical access to it. And then it goes over how to crack the SAM file using John the Ripper, an offline password cracking utility. Additionally, it goes over using Ophcrack and rainbow tables to crack the hashes. Each of these would be used in different situations, and that is explained as well.
Figure 19 – Cracking password hashes with John the Ripper

Figure 20 – Cracking password hashes using Ophcrack and rainbow tables
5.6 Lab 6 – Brute Forcing Against Remote Services

A brute force password attack is essentially an attack that tries to guess thousands of different passwords for either a single username or multiple usernames. This is generally used against remote services in order to try and force a log in to a server or service with the username.

In this lab, the user is taught how to use the Hydra tool to perform a brute force attack against our Ubuntu server running SSH. The process involves tweaking settings and running the attack using a dictionary or password file. Hydra will attempt every password in the file for the username(s) listed. The process can take a very long time, but it’s an important tool to learn how to use.

![Figure 21 – Conducting a brute force attack using Hydra](image-url)
5.7 Lab 7 – The Social Engineering Toolkit

The Social Engineering Toolkit has a variety of tools and options within it that can gain access to a victim’s computer or credentials. Specifically, this lab focuses on spoofing a website to gain access to their system or to gain their credentials via the Credential Harvester. In order to do this, we must ARP Poison the network as well as spoof DNS with Ettercap. This means that when a user attempts to connect to a website such as gmail, they will be redirected to our spoofed gmail website.

![Figure 2](image)

Figure 2 – Using the Social Engineering Toolkit to spoof a website

In this lab, this process is detailed. Two “payloads” are discussed in this. The first one exploits the victim computer if they run a script on our spoofed website. The other merely sends their credentials to our BackTrack machine if they decided to enter them in to our spoofed website.
Figure 23 - Spoofing DNS using Ettercap

Figure 24 – Testing our spoofed website and exploit
5.8 Lab 8 – Introduction to SQL Injection and Sqlmap

SQL Injection is a very popular and important technique when testing websites. If a website uses a database backend for many of its processes, it’s possible that the web server is vulnerable to this type of attack. Sqlmap is a tool within BackTrack that uses SQL injection techniques to learn more about the vulnerable database(s) on a website.

![Software error]

**Figure 25** – Testing for SQL Injection vulnerability

In this lab, the basics of SQL Injection are demonstrated and explained. Badstore Linux is used as the victim Web Server, since it is configured poorly on purpose to demonstrate techniques such as this. Additionally, once some understanding of SQL Injection is established, we learn more about the databases on Badstore by using Sqlmap.
5.9 Lab 9 – Nikto, Cross Site Scripting, and Cookie Information

Nikto is a web vulnerability scanner. It’s used on websites in a similar manner to how Nessus is used. It simply scans for known vulnerabilities that can be exploited using website pentesting techniques such as SQL Injection or Cross Site Scripting. Cross Site Scripting is executing scripts on an insecure website, allowing you to manipulate what happens on the website when users do certain actions. For example, you can gain access to a user’s cookie information if a site is insecure in this manner, and thus gain access to that user’s account.
In this lab the processes detailed above are demonstrated against the Badstore Linux machine.
5.10 Lab 10 – Examination

In this lab, the user is tested on all of the tools and knowledge they’ve gained throughout the first 9 labs. The user is given lab files that are already configured, and simply told to break into the systems (without physical access) using the tools and techniques they now know how to use. Additionally, in order to ensure they actually got in, important information is hidden within the network that the user needs to list in their lab report. This will ensure the user completes the lab.
6.0 Testing

There are a few testing methods I’ve used for my project. I tested for functionality, for purpose, and for usefulness.

6.1 Functionality

In this test I was merely looking for whether or not the labs worked as they were intended. The particular issue when testing was the configuration of the systems and making sure that when the user used my Virtual Machines they were getting the same experience I did when I created the labs. This is where I found the most trouble in the creation of my labs. Frequently I would go through the process of configuring the virtual machines for my lab, only...
to find out that it was inconsistent or didn’t work as intended. The solution to this was frequently to start over and create snapshots of the virtual machine, or simply “pause” the machine where it was working and copy that as the default lab file.

6.2 Purpose

The second measure was whether or not the lab had a clear purpose. In this I showed my labs to a couple students as well as my technical advisor and one of the professors at the University of Cincinnati. Each of them took a look at my labs to make sure the content had a clear purpose and wasn’t redundant or unnecessary.

6.3 Usefulness

The third measure is how useful is the information in these labs. This measure was less about the whether or not the information was redundant, and more about if it was important for a student new to pentesting. This was a measure that was tested by the feedback I’ve received from Professor Mark Stockman who is currently using these labs in one of his classes, and will be using them in the future as well. Fortunately this did not cause me any issues, since the content I used in my labs was gathered from reliable sources such as SANS and Offensive Security.

6.4 Issues

The main issues I had occurred in the functionality phase of testing. Frequently, I would get a lab to work as intended and then go back to do it again on video only to realize it didn’t
work the same way because of changes made to the machine. This cost me a lot of time, because I’d essentially have to reconfigure each lab machine a few times in order to allow the user to have the same experience.

Beyond that, in testing I was given suggestions to alter some of the labs. These were minor changes, and had no major effect on the project as a whole.

7.0 Conclusion and Recommendations

This project has been a success. The lab content was seen as well done by Professor Stockman, and he will be using them in his classes. Overall I’ve learned much through the creation of these labs that has personally helped me understand Information Security more. Additionally, the feedback I have gotten from Tech Expo lets me know that this project is necessary and helpful for anyone in the IT field to understand more of where a hacker’s mind will be when trying to infiltrate systems.

For anyone that would want to do a similar project to this for Senior Design, my recommendation would be to gauge the current state of Information Security. It’s always evolving, and even since starting my project there’s already a new pentesting Linux distribution that’s meant to replace BackTrack. BackTrack will likely be used and important for some time, because the techniques are still useful. However, remaining relevant is always important.
8.0 References


