Cybersecurity Training Program

by

Kristen J. Smith

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

© Copyright 2015 Kristen J. Smith

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

Kristen J. Smith

Mark Stockman

3/09/15

Date

4/20/15

Date

University of Cincinnati
College of
Education, Criminal Justice, and Human Services

April 2015
# Table of Contents

Abstract .................................................................................................................................................. 1  
Problem Statement ............................................................................................................................... 2  
Solution ............................................................................................................................................... 3  
Research ............................................................................................................................................... 3  
Development ....................................................................................................................................... 4  
User Profile(s) and Design Protocols .................................................................................................. 5  
System/Program Flow Diagram .......................................................................................................... 6  
Use Case Diagram ............................................................................................................................... 7  
Complications & Roadblocks ................................................................................................................ 7  
Conclusion ........................................................................................................................................... 9  
Works Cited .......................................................................................................................................... 10  
Appendix A .......................................................................................................................................... 11
Abstract
The cybersecurity training program served as a tool for both students and entry-level cybersecurity professionals to aid in the hands-on experience that’s necessary to succeed in the cybersecurity field.

This project was chosen because there is a lack of easily attainable training resources for the cybersecurity field. Most of the current training programs that are the most reputable and accepted within the cybersecurity field cost thousands of dollars, in addition to travel and lodging costs. The lack of resources may discourage future cybersecurity and information security professionals from pursuing their interest in the field.

The program created will hopefully fill that gap, and allow students and entry level personnel to gain basic hands-on technical security experience in a safe environment.
Problem Statement
The cyber and information security fields are rapidly growing, and the need for properly trained personnel is at an all-time high. However, the academic world hasn’t really caught up to this yet, and therefore there are very few undergraduate and graduate degree programs dedicated to these fields. To make up for the lack of formal training, many interested individuals are either testing the waters on their own (sometimes running into trouble in the process) or paying thousands of dollars for training and the accompanying certifications.

While researching across the Internet for current security training solutions, there were many tutorials on blogs, forums/discussion boards, and video sites that could be easily found by anyone. However, what was most disturbing about these tutorials were the comment sections. Many individuals that asked for assistance or clarification for the tutorial were ridiculed or told that their skill level was not up to par in order to successfully complete the tutorial.

Although it’s okay to assume that followers of the tutorial will have a certain level of knowledge about the topic being presented, if intelligent and informed questions are being asked, it would be in the best interest of all involved (the tutorial’s author, and current and future followers) to at least lend a hand in narrowing down what the cause(s) of the errors or other issues may be.
Solution
The Cybersecurity Training Program addresses the aforementioned difficulties that are encountered when learners attempt to gain hands-on knowledge about security topics.

The program provides users with labs that cover basic security topics, such as phishing and browser hijacking/exploitation. The program is designed for those who are new to the world of security and would like a guided approach to learning more about the field and the technologies used in it. The ideal user group would be post-secondary students and entry-level security professionals. The labs are in-depth enough to give users a thorough understanding of the topic being learned, but easy enough that each one shouldn’t take more than an hour or two to complete.

Research
The research conducted throughout the course of this project has primarily consisted of finding and documenting current security training programs and certifications and reading countless articles, discussion forums, and blogs dedicated to the security industry in order to gain a better understanding of what’s expected from employees (especially those that are entry-level), and also research related to the actual development process for the program. All technologies used for the creation of this project are new to me, and I’ve taught
myself everything I know during this time. Needless to say, these have been incredibly daunting tasks.

Research conducted has also consisted of casual “interviews” with colleagues and friends, inquiring about their interest in security, their current level of knowledge any security topics, and any steps they’ve taken to further their knowledge and hands-on experience. From these interviews, I was able to conclude that nearly all interviewees (with the exception of maybe one or two individuals) have run into the same roadblocks: although they’re interested in the field, training solutions are very hard to come by, and thus discourage them from pursuing training due to time constraints, monetary constraints, or both. Reputable, reliable training in the security field costs way too much money and requires countless hours of studying, with very little hands-on exposure.

**Development**
The ultimate goal for this project is to develop a program that will simulate various security scenarios through activity packets. These activity packets are small programs designed to be launched within the program and will contain a lesson, a virtual machine, and the activity itself. Some activities will utilize AI to mimic a more realistic training environment. User’s activity progress will be tracked and a report will be available giving the percentage completed, time elapsed, and tasks successfully completed will be marked with a green checkmark.
However, due to the time constraint of the course, this full product isn't feasible to be completed in time for Tech Expo. Instead, I've resolved to simply building the framework of this program. My framework will consist of the program’s UI and basic functionality.

The framework is coded in C# using Microsoft Visual Studio 2013, and the labs are currently using Kali Linux for the operating system and Oracle VirtualBox as the virtual machine manager. The UI is still in the beginning phases, but the goal is to have a clean, minimalist style interface that's obviously intuitive and inviting.

User Profile(s) and Design Protocols

<table>
<thead>
<tr>
<th><strong>User Profile Form</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Application:</strong> Cybersecurity Training Simulator</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Potential Users:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Entry-level cybersecurity professionals and those studying for security certifications.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Software and Interface Experience:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>The program will utilize an intuitive UI that will provide the user with options to select their preferred lab as well as other simple features, such as the ability to display a basic progress report.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Experience with Similar Applications:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Assuming the user has basic technical skills, there should be no problem using the program, considering that 99% of the work is done using virtual machines.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Task Experience:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>The user will need to have at least basic hands-on networking knowledge and have basic understanding of cybersecurity principles.</td>
</tr>
</tbody>
</table>
**Frequency of Use:**
Optimum use will be 2 – 3 times per week

**Key Interface Design Requirements that the Profile Suggests:**
- UI needs to be intuitive
- Ability to provide proof of completion and/or progress
- Comprehensive enough to be a viable study tool for advanced security certifications

**Design Protocol Form**

**Application:** Cybersecurity Training Simulator

**Platform Availability:**
This program will initially only be available for Windows PCs. Once the initial phase has been successfully implemented and runs smoothly (aka deliverables for Tech Expo), the next step will be to create a companion site along with an app. The purpose of these two services will be to allow the trainee to keep track of their progress on the go, as well as an easy way to print certificates of completion via the companion website. The actual simulator program will not be available on mobile devices or as a web application.

**Software Experience:**
The program will utilize an intuitive UI that will provide the user with options to select and launch their preferred training session.

---

**System/Program Flow Diagram**
The following flowchart is to show how the program is intended to flow, with the demo labs created for the Expo as examples.
Complications & Roadblocks
When researching and elaborating on the original project design, although the project was going to be heavily software-related, it had not been anticipated that the few coding tasks needed would be so robust and require such in-depth knowledge of the coding language. One of the initial aspects of the project,
progress reporting, proved to be the most difficult part of the project, and eventually ended up being considered too involved for a project with such a short timeframe to completion. The progress reporting would have required scripting within the virtual machine to capture data that would have then been pulled back into the desktop software, assigned appropriate variables, and then somehow written into a PDF file generated by the program for the user to download at the click of a button, or possibly preview from within the program.

Unfortunately, the difficulty of this immense task was not realized until the project was fairly advanced. At that time, it was too late to exchange the progress reporting functionality for something else, so that feature was completely wiped from the plan, and instead, more focus was placed on the labs. Instead of just one demo lab, it was decided that three labs would be created in order to demonstrate the breadth of knowledge that the program would be used to cover. Focusing more on labs, instead of on sophisticated program features allowed for more in-depth research into the security realm.

One major roadblock with the labs was with the initial phishing lab. The name of the tool is being withheld, however the specific tool that one of the demo labs was being created for was a relatively new, open-source tool on the market (about 1.5 years) and was severely lacking in accurate and up-to-date documentation for both installation, configuration, and troubleshooting. Many secondary tasks needed to get the tool up and running were missing from the
available documentation (both in written and video formats), so that information had to be filled in from personal research and trial-and-error. However, the lack of vital knowledge about the tool ultimately caused the tool to be scrapped from the project, and another, much simpler tool, used in its place that still demonstrated the techniques and information that the original lab was intended to, but with a much more beginner-friendly process.

Conclusion
This is a project that is intended to carry on past Tech Expo and graduation. I intend to continue building this program, expanding its functionalities, and eventually market it to the public. Due to the research I’ve gathered both on my own and through informal interviews/speaking with multiple individuals, I’ve found that this tool is something that is not only desperately needed in the security realm but also something that is wanted, which is the most important piece to this big puzzle. A program (and eventually a training suite) such as this wouldn’t make much of an impact if the audience for it isn’t present.

I’ve also learned that this project can offer so much more than what was originally anticipated, and I have an enormous task ahead of me in regards to building and marketing a finalized product. This senior design project has helped me put this whole idea into perspective and has given me insight into how I can move forward in the right direction.
Works Cited


https://github.com/pentestgeek/phishing-frenzy
http://www.techexams.net/forums/infosec/
http://www.reddit.com/r/AskNetsec
http://stackoverflow.com/
http://cs.stackexchange.com/
http://security.stackexchange.com/
http://programmers.stackexchange.com/
http://unix.stackexchange.com/
Appendix A

Note: *Full text for code and labs available in the attached documents*

Code Snippets

Form1.cs

```csharp
namespace CybersecuritySimulator
{
    public partial class MainForm : Form
    {
        public MainForm()
        {
            InitializeComponent();
            string beefIntro1 = "D:\VirtualBox VMs\Kali2\Kali2.vbox";
            string beefIntro2 = "C:\Users\Krissy\Dropbox\SD\proj. mgmt. 2\labs\IntroToBeEF.docx";
            string phishSE1 = "D:\VirtualBox VMs\Kali-Phishing\Kali-Phishing.vbox";
            string phishSE2 = "C:\Users\Krissy\Dropbox\SD\proj. mgmt. 2\labs\Phishing.docx";
            string MITM1 = "D:\VirtualBox VMs\Kali-MITM\Kali-MITM.vbox";
            string MITM2 = "C:\Users\Krissy\Dropbox\SD\proj. mgmt. 2\labs\MITM.docx";
            //TimeSpan time;

            private void BEEF(object sender, EventArgs e)
            {
                System.Diagnostics.Process.Start(beefIntro1);
                System.Diagnostics.Process.Start(beefIntro2);
            }

            private void PHISH(object sender, EventArgs e)
            {
                System.Diagnostics.Process.Start(phishSE1);
                System.Diagnostics.Process.Start(phishSE2);
            }

            private void MITM(object sender, EventArgs e)
            {
                System.Diagnostics.Process.Start(MITM1);
                System.Diagnostics.Process.Start(MITM2);
            }

            //private void timeElapsed(object sender, EventArgs e)
            //{
            //   timeDisplay = time;
            //}
        }
    }
}
```
namespace CybersecuritySimulator
{
    partial class MainForm
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed; otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

    #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        /// <param name="disposing">true if the container should be disposed; otherwise, false.</param>
        private void InitializeComponent()
        {
            this.label1 = new System.Windows.Forms.Label();
            this.buttonBeef = new System.Windows.Forms.Button();
            this.buttonPhishing = new System.Windows.Forms.Button();
            this.buttonMITM = new System.Windows.Forms.Button();
            this.timeDisplay = new System.Windows.Forms.Label();
            this.SuspendLayout();
            //
            // label1
            this.label1.AutoSize = true;
            this.label1.Location = new System.Drawing.Point(4, 8);
            this.label1.Name = "label1";
            this.label1.Size = new System.Drawing.Size(365, 31);
            this.label1.TabIndex = 35;
            this.label1.Text = "Cybersecurity Training Simulator";
            //
            // buttonBeef
        }
    #endregion
    }
}
Program Screenshots

Main Menu (demo)

Cybersecurity Training Simulator

- Browser Exploitation: Intro to BeEF
- Social Engineering: Phishing
- Network Security: Man-In-The-Middle
Browser Exploitation: Intro to BeEF

Overview

The Browser Exploitation Framework (BeEF) is an open-source penetration testing tool used to test and exploit web application and browser-based vulnerabilities. It uses web app and browser vulnerabilities to assess the security of a target and carry out further intrusions. All of this is done by "hooking" browsers.

The BeEF server is a Ruby on Rails application that communicates with the "hooked browser" through a web-based user interface.

This lab covers the basics of BeEF and will instruct you on how to launch the tool, accessing the interface, and executing a simple intrusion on a localhost browser.

Lab

1) Log into virtual machine (root;Bearcat1)

2) Open Terminal and type the following commands, in order:

   cd /usr/share/beef-xss
   ./beef

After you’ve done this, you’ll see the following screen after the tool loads:

```
[13:01:32] [*] Bind socket [tcp@0.0.0.0:2080] listening on [0.0.0.0:2080].
[13:01:32] [*] Browser Exploitation Framework (BeEF) 6.4.4.9-alpha
[13:01:34] [*] BeEF is loading. Wait a few seconds...
[13:01:41] [*] 2 network interfaces were detected.
[13:01:41] [*] running on network interface: 127.0.0.1
[13:01:41] [*] Register API: key: 99219b6d34c7ae45e492d26e47fe5b5eRam
[13:01:41] [*] BeEF server started (press control+c to stop)
```
Social Engineering: Phishing

Overview
Phishing is probably one of the most common attacks that are executed and are used to lure people into providing personal information. The great thing about phishing is that it’s fairly automatic. You can set up a phishing site, some sort of output file or database for the personal information that will be entered, and then just send out a mass email to as many email accounts as possible to heighten your chances of someone taking the bait and providing information.
This lab just scrapes the surface of a full phishing attack, and demonstrates how easy a believable phishing site could be created in a matter of seconds.

Lab:
1) Log into virtual machine (root;Bearcat1)
2) Because we’ll be using websites to gather information, we’ll need to start Kali’s web server, Apache:
   apachectl start
3) Once Apache has been started type setoolkit to launch the tool, and agree to the Terms of Service to begin!

The Social-Engineer Toolkit is designed purely for good and not evil. If you are planning on using this tool for malicious purposes that are not authorized by the company you are performing assessments for, you are violating the terms of service and license of this toolset. By hitting yes (only one time), you agree to the terms of service and that you will only use this tool for lawful purposes only.

Do you agree to the terms of service [y/n]: y
Overview

A Man-In-The-Middle (MITM) attack is an attack that, when executed, intercepts information between two parties that are under the impression that they are communicating with each other directly. MITM attacks are typically used to harvest login information and sensitive personal information.

Lab: Part 1 (IP forwarding)

*Take note of all IP addresses needed for the lab here

Linux IP (attacker):
Gateway IP:
Victim Machine IP:

1) Log into virtual machine (root;Bearcat1)

2) Open Terminal and type in the following command

   ```
   cat /proc/sys/net/ipv4/ip_forward
   ```

3) The default value is “0”. It should be set to 1. To change the value to 1 enter the following command:

   ```
   sudo echo 1 >> /proc/sys/net/ipv4/ip_forward
   ```

4) Next, check the ip_forward (cat /proc/sys/net/ipv4/ip_forward) file to make sure the value equals 1 now.

Part 2 (ARP poisoning/redirections)

This part of the lab will include executing an arp poisoning attack that will redirect the traffic from the victim PC to our linux machine. Also, please note that the victim machine and attack machine must both be on the same switch or access point in order for this attack to be successful.