Activity: Vulnerable Web Application Setup

Overview

The purpose of this activity is to get you acquainted with the overall layout of several web applications.

Setup Instructions

Use **Windows** for this rather than Ubuntu.

This activity is for groups of 2-3 people. Do this activity using Chrome.

1. Download the **XAMPP portable installation** (https://uncw.instructure.com/courses/16302/files/408711/download?wrap=1) . Unzip it - this will take a few minutes. **XAMPP** (https://www.apachefriends.org/index.html) is a package of technologies needed to run a stereotypical data-driven web site.
2. Go to your unzipped XAMPP directory, and run `setup_xampp.bat`. This will go through a couple of tests, and will eventually ask you if you want to "refresh" (option 1). The "Press any key to continue..." screen indicates this step is finished. Press any key to close the console window.
3. Create a GoogleDoc called "Web Application Vulnerabilities" and share it with the instructor.
4. Run the following. **You will be asked may ask for administrator privileges to open up the firewall -- do not allow!** Even if you're on your own machine, you don't want to expose your local machine as a server to vulnerable web applications.
   - `apache_start.bat`. As it runs, XAMPP will ask for administrator access - hit `Cancel` to that (even if you're on your own machine - no need to expose yourself as a server on the network).
   - `mysql_start.bat` - this starts the MySQL relational database system
   - You should have two console windows open with something similar to "App" is starting..." Leave these consoles open. Closing them will terminate their respective applications.
5. Start up a browser and go to **http://127.0.0.1:10000** (http://127.0.0.1:10000/) . Note the passwords for the various intentionally-vulnerable apps.
   - Note: The IP address 127.0.0.1 is the "loopback" address for your computer. 10000 is a socket in the Operating System on which apache is listening for connection. The default socket for HTTP connections is 80, but we have specified a different one in our XAMPP configuration.
6. Log in to Damn Vulnerable Web Application (DVWA). This is an interactive site for learning how to exploit various web application vulnerabilties.
7. Go to DVWA Security, and change the security level to "Low".
8. Go to Setup, and hit "Create/Reset Database"

Mac or Linux?

The instructions above are for Windows. If you want to run DVWA on Linux, here:

- A **student's instructions for Linux** (https://github.com/meyersbs/linux-dvwa-instructions) (can be useful for Macs too)
- **XAMPP project** (https://www.apachefriends.org/faq_osx.html)
- **Original DVWA repo** (https://github.com/ethicalhack3r/DVWA)

Inspecting HTTP traffic and web page contents

All of the major browsers have a "Developer Mode" that lets you look at the details of the HTTP Request and Response as well as the contents of the website that the server sends to you (the client). The sites below tell you how to use these tools

- **Chrome Development Console** (https://developers.google.com/web/tools/chrome-devtools/console/get-started) : Press Command+Option+J (Mac) or Control+Shift+J (Windows, Linux, Chrome OS) to jump straight into the Console panel.
- **Firefox Developer Tools** (https://developer.mozilla.org/en-US/docs/Tools) : You can open the Firefox Developer Tools with Ctrl + Shift + I or F12 on Windows and Linux, or Cmd + Opt + I on OS X.
- **F12 developer tools in Internet Explorer or Edge** (https://docs.microsoft.com/en-us/microsoft-edge/devtools-guide) : Press F12
- **Safari Web Developer Tools** (https://support.apple.com/guide/safari/use-the-developer-tools-in-the-develop-menu-sfri20948/mac) : From the Safari “Develop” menu select “Show Web Inspector” or use the keyboard shortcut Option+Command+i

Working with HTML, JavaScript, and MySQL

Our DVWA installation consists of an Apache HTTP server, PHP application software, and a MySQL database. You will need to have some working knowledge of the HTML, JavaScript, and SQL languages to exploit the web app or to understand other's exploits that you test out.

The simplest and cleanest references for these are at w3schools.com in my opinion. Brush up on HTML first before proceeding to JavaScript.

- **HTML5 Tutorial** (https://www.w3schools.com/html/html_intro.asp) - Check out Basic, Elements, Attributes at a minimum. You can then look up the other portions to learn more about specific tags as needed.
- **JavaScript tutorial** (https://www.w3schools.com/js/js_intro.asp) - Originally based on Java and supported by every major browser. Read from Where To thru Events. Variables, operators, and artihmetic should look familar. Pay special attention to the Events tutorial.
- **SQL Tutorial** (https://www.w3schools.com/sql/sql_wildcards.asp) - Everything up through SQL Wildcards will be useful. Also jump to the very last tutorial: SQL Comments.

Working with DVWA

From the DVWA website: "Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goals are to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and aid teachers/students to teach/learn web application security in a class room environment."

Do not allow DVWA through the firewall.

Change the "Difficulty Level" on the "DVWA Security" page on the site. Start out with things on Low, then gradually increase the security level to see if you can bypass each layer of protection.

Each of the Vulnerability pages has a "View Source" and "View Help" button. The "View Source" button will show you relevant PHP source code that may clue into how the page is vulnerable. The "View Help" button will give you some hints about how to create an exploit depending on the difficulty level.