Fall 2021 - COMP 424
Course Project Specifications
A Secure Web Server and Login System

In this project, you are to prepare a web server and a website for a small business company. The company is a typical small business company with medium hit-rate site visits for their webpage and very interested in security while not significantly compromising usability. Your design decisions should take these into consideration.

1. Preparing the web server

You will prepare a web server on the latest version of Ubuntu (a Linux distribution) along with necessary security tools to protect it from popular attacks using the most popular firewall (IPTables) and intrusion detection system (Snort). Note that the web server is also a SSH server, so you are required to install OpenSSH, and allow SSH traffic to go through as well. You will install, configure, and implement your designed policies using these two security tools. You are required to install LAMP (Linux, Apache, MySQL, PHP stack) on Ubuntu with necessary configurations suitable for your design and implementation.

You will write two shell scripts to automate the installation and configuration of your system for disaster recovery purposes with comments for every single command:

installation.sh: It will include all the commands regarding installation of all the necessary services and tools. Also, all configuration scripts for LAMP.
implementation.sh: It will include all the commands regarding implementing policies for your firewall, IDS, etc. You may use a stream editor such as “sed” to implement them using your automated scripts, if that requires editing specific files.

2. The webpage

The company has a simple webpage paradigm: A login page, and upon successfully logging into the system, it will allow the user to download the company_confidential_file.txt. This page will also display the successfully logged-in user: "Hi, (First-Name Last-Name)", "You have logged in X times" and "Last login date: Y". You will implement the webpage using PHP, MySQL, HTML, JavaScript, and CSS.

1) The login page

An "https" webpage redirected from localhost would be a simple login page where the user can enter the username and password and a “Login” button. Also, available on that page are two links: (i) New User? Sign Up and (ii) Forgot Username or Password?. Keep in mind that you may need to create and embed your own certificate in Firefox, so that the browser can verify the public-key for encryption.

2) New user sign up

Your new user sign up page must have at minimum the following features, yet you may add more features to increase the security of your authentication system.

- Typical information about the user: First Name, Last Name, Birth date, E-mail.
- User's password: Password and re-enter password along with real-time proactive password metric feedback (i.e., is this password weak, strong, etc.). You will decide what the required password selection rule should be.
- Some type of challenge-response test to determine whether or not the user is human (e.g., CAPTCHA and reCAPTCHA).
- Activation of account via e-mail.
- A set of security questions for password retrieval.

3) Forgot username and password
You will decide what *secure* method should be used to retrieve the username and password.

Extra Credit (Up to 15%): Two-Factor Authentication

Final notes:

You must log appropriately when suspicious activities, intrusions, or attacks are detected. Also, your implementation is required to log all of successful and unsuccessful login attempts.

And lastly, you should address the following attacks in your design and implementation, if they are relevant:

1) Brute force
2) SQL Injection
3) Buffer Overflow
4) XSS
5) Cross Site Request Forgery

Due Dates:

- First team presentation scheduled on Oct. 7th.
- Second & final presentation scheduled on Nov. 4th.
- Each presentation will be between 8 to 10 minutes per team including a live demo of what was accomplished so far.