Project Assignment

Due : 11:59 PM, June 30th , 2025

Group Formation: This is a group project that involves minimum 3 members You must sign up on Canvas in the "people" icon under PROJECT-GRP<ID> group. Each member will be responsible to perform a specific task in the project. You may create custom roles for your projects. Some of the examples of roles are:

- The Research Analyst gathers and analyzes necessary data and information.
- The Technical Specialist handles technical configurations, codes and problem-solving.
- The Document Specialist maintains clear and comprehensive project documentation.

Choice of Projects: Choose one of the following projects for your group.

1. **Network Security**
   - Using VMs, install a web server. You can use Wordpress or any other suitable packages.
   - The content of web page hosted by the server can be minimal.
   - The server needs to store a password file which contains two elements—username and password. If you feel you need a separate database server to do this part, go ahead.
   - Password should NOT be stored in cleat text—they need to be hashed.
   - Your web page should be accessible via the internet.
   - Your web page needs to have access restrictions.
     - When the user tries to open the webpage, the user needs to be authenticated.
     - Prompt the user for username and password.
     - Find the hash of the user entered password and compare it with the stored hash value.
     - If there is a match, the user should have access to the webpage
   - Your webserver needs to be behind a firewall. You may use iptables in Linux to configure firewalls.

2. **Secure Website**
   - Build a website application that provides secure access to a database of employees, name, salary, date of birth, department, home folder, SSN etc.
   - An employee should be able to see only their own portal and information.
   - An administrator should have access to all the information of all employees.
   - You may consider using a SQL database to provide information about each of the employees.
   - You must make appropriate DNS changes to your /etc/hosts files to allow website access
   - You must secure the website using the PKI-based certificates for secure access.
   - You may use resources from Lab 1 and 2 to build your project.

3. **Password Manager**
   - Your application needs to have a suitable graphic user interface or a command line interface.
   - The user should be able to create an account for the program or application.
     - The program should be able to support multiple users, each user identified by their account username and password.
     - The user logs in using the account information.
     - The account credentials should be saved on the local computer, but not in plain text. The password must be hashed.
     - The password must not be visible when the user is typing the password.
   - After the user logs in, the user should be able to store their username and password for different services/websites (e.g. Google, Facebook, etc.) in the program.
     - The user must be able to add services as well as view credentials for services that have already been added.
     - The credentials should be saved on the local computer, but not in plain text.
       - The program should encrypt the password using a user-specific key. Do not hash the password since the password needs to be recovered in plain text.
       - The password must not be visible to the user until the user chooses to display it.
     - When the user wants to view the stored password, the program must ask the user for the key, decrypt the password and provide it in clear text.

4. **Two Way Authentication System**
   - Your application needs to have a suitable graphic user interface or a command line interface.
   - Your authentication system program needs to allow users to create an account. It needs to contain at least three elements—a username, a password and phone number.
   - You can store the account information in a local computer. Password and phone number should NOT be stored in cleartext. The password needs to be hashed, the phone number needs to be encrypted.
   - Your authentication system program needs to allow users to log in.
     - Your system needs to ask the user for username and password.
     - You need to compare the hash value of the user entered password with the stored hash value.
     - If there is no match, inform the user of the wrong password.
     - If there is a match, use appropriate web application or package to send a secret code of your choice to the user on the stored phone number.
     - Prompt the user to enter the secret code
       Check if the user entered code matches the code generated by the system.
     - If it matches, the user has authenticated. Inform the user.
   - You do not need anything behind the authentication system for this project, just build the authentication part.

Submission Guideline:

- **Code, Infrastructure, VM or files:** You need to submit the GitHub link to you code and related files on Canvas. If your project involves VM, you must upload the VM to the OneDrive and share the VM with appropriate permissions for Grader and the Instructor. Also, you must provide steps to test your project.
- **Final Report:** You must submit the final report on Canvas and include the below information
    - Project name, group number and group member names
    - Objective of the project (1 paragraph)
    - Group member roles and the tasks they performed during the project (1 paragraphs)
    - Implementation design of the project and explanation. (1-2 Page along with pictures)
    - Project output screenshots and their explanations (Screenshots and explanations)
    - Challenges and limitations of your project (1- 2 paragraphs)
- **Project Demo:**
    - You must prepare a 3-5 minute demonstration video of your project and submit that on Canvas assignment submission page.

Note: You can use any online resource for the project. However, you must cite or reference the resource in a reference section and indicate the amount of code you followed from that project.

Grading Rubric:

- Successful, Code Infrastructure, VM, file executions: 30 Points
- Project Demo: 30 Points
- Final Report: 30 Points
- Group Work: 10 Points