Group C

JIANFEI FENG  
JIAQI NI  
MENGCHUAN LIU  
JIEFAN LIN  
Kalina Petrova

SEPR FINALLY REPORT

# Introduction

With the development of IT technology, more and more website get attacked and cracked by hackers, therefore, developers are looking for a solution to secure their website. In this project, we are trying to develop a website with some simple security techniques and this report is to describe how we defend others groups’ hack.

This report will divide into 3 parts. The first part we will discuss more about our web application functionalities, development process and architecture design. Then we will discuss about the Threats and Analysis part, this part will focus on explain theoretical research and analysis conducted over all source of risks, attacks on website. The last part is to describe the flow how to defend the attack.

Finally, we have personal evaluation for each group member and each one will give a describe what they learned from this course.

# Application Overview

This application we build is a simple social website which allowed users to register, login, write comment. The application is general made by html5, css3 and PHP code.

# Architecture design

# Threat Analysis

## CIA Analysis

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Data elements** | **Data classification**  **C.I.A** | **Explanation** |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Authorization Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Group** | **Customer** | **Anonymous** | **Employee** | **Administrator** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## 

## Risk analysis

|  |  |  |  |
| --- | --- | --- | --- |
| **Group** | **Risk** | **Impact** | **Security Needed** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Secure Design

## Attack tree

|  |  |
| --- | --- |
| Name | Tree |
|  |  |
|  |  |
|  |  |
|  |  |

## SQL Injection

SQL injection is an attack in which it inject SQL query/command as an input possibly via web pages, the hacker might try to insert malicious SQL command into the provided inputs on the web server. For example, hacker can insert a SQL command instead of inserting username in textbox to trick the web server.

Possible solutions:

* + Not using string concatenation or string replacement.
  + Validate all input.
  + Use “Prepared Statements” to build SQL statements.

## Cross-Site Scripting (XSS)

Cross Site Scripting (XSS) is an attack/vulnerability that permits an attacker to inject code (typically HTML or Javascript) into contents of a website not under the attacker's control. Hacker can stead the cookie from user and send request to server which is authorized. The server can’t distinguish the request made by user or hacker, so in this way hacker can easily trick the server.

Possible solution

* Using htmlentities() method which will do the code conversion.
* htmlspecialchars().

## Session hacking

Session hacking sometimes also called cookie hacking, which is the exploitation of a valid computer session to gain unauthorized access to information or services in a computer system. The hacker can get unauthorized access to webserver after they gained valid token session ID.

## Cross-Site Request Forgery (CSRF)

Cross-site request forgery, also known as one-click attack or session riding and abbreviated as CSRF or XSRF, is a type of malicious exploit of a website where unauthorized commands are transmitted from a user that the website trusts. Unlike cross-site scripting (XSS), which exploits the trust a user has for a particular site, CSRF exploits the trust that a site has in a user's browser

## Functionality

### Registration

This is the first function which allowed user to register an account to login our website. There are several mandatory information which need user to fill in such as username, password, repeat password and email address. For each property, it must to achieve some specific requirement, for example, the password must be at least 8 characters long and contain at least 1 upper case, 1 lower case and 1 digit. After user finishing registration, the data will be stored in the database.

### Login

This function allowed users to login our website after finishing registration. The user need to input username and password, this information will be passed into database to check if it is correct. If it is not correct, system will require try again. If it is correct, user can go login our website and give comment.

### Comment

This function allowed user to write comment and send to message board after user login their account. The message board will keep these comment, so other user can see it after they login.

### Modify Password

This function require user input their old password first, and then input the new password. The new password need keep the same rules as old one.

### Log out

This is how user sign out their account and clear the sessions

### Upload picture

This function allowed user upload a JPG, JPEG, GIF, PNG file on our application.

## Misuse case

## Code example