# CSE610 Homework #1: Capture The Flag

Due: April. 9, 11:59 PM

- **Late submission policy.** Please refer to the course web page.
- **Submission guidelines.**
  - You should submit both your report and flags.
    - Solve the problems and submit the flags to our homework webpage: `http://10.20.12.187:4000`. This server can only be accessed from the UNIST internal network. Please use a VPN to access from outside.
    - You should upload a single PDF file on BlackBored. Your report (`your_ID-last_name.pdf`) can be written in either English or Korean.
  - The name of the PDF file should have the following format: `your_ID-last_name.pdf`. For example, if your name is Gil-dong Hong, and your ID is 20231234, then you should submit a file named "`20231234-Hong.pdf`".
  - If your solution includes some code (Python, C, etc.), you should embed them in your PDF.
- **Capture The Flag (CTF) guidelines.**
  - You can find each problem on our homework webpage: `http://10.20.12.187:4000`
  - If you solve each challenge, you will be able to obtain a flag. Submit the found flag to the website. Each flag is in the following format: `flag{[0-9a-f{32}]}` (e.g., `flag{1a79f4d60de6718e8e5b326e338ce573}`)
  - Your score in the CTF scoreboard is nothing to do with the actual score for your homework. The CTF score is just for fun.
  - Do not attack the CTF environments, including web services!
  - If you think the services are not working properly or have any questions, please send email to Prof. Wi.
- **XSS attack guidelines.**
  - For XSS challenges (Problems 6–10), we prepared an imaginary victim and stored the flag in this victim's cookie. Therefore, you need to find a vulnerability in the web service, create a malicious URL, and send the URL to this user so that you can read the cookie.
  - Especially, you should manually send a malicious URL to this victim user via this webpage: `http://10.20.12.187:4005/check.php`. When you send the URL to the victim user, she will automatically visit the URL. This process might take at most 10 seconds.
  - To read the user's cookie, you may need your own server (Recall the `attacker.com` in pages 16 and 26 of the lecture slide "Cross-Site Scripting (XSS)" [1]). We recommend using the following service: `https://webhook.site/`. Through this service, you will get your unique URL that acts like `attacker.com`, and you will be able to see logs (e.g. query strings, form contents, etc.) of all accesses to that URL.

## Problem 1. Login (5 points)

In this challenge, you should sign in to the web service as admin without knowing the password. Once you successfully sign in to the service, you will find the flag.

(Hint) In the database, there exists a table with the following structure.

```
CREATE TABLE user (
    idx INTEGER,
    username TEXT,
    password TEXT
);
```

(a) (1 points) Describe the SQL query used in this web service

(b) (1 points) Describe the vulnerability in this web service.

(c) (2 points) Explain in detail how you can exploit this vulnerability to get the flag.

(d) (1 points) What is the flag?

## Problem 2. Password (10 points)

In the same web service used in Problem 1, you now need to find out the password of `admin`, which is the flag of this challenge.

(a) (8 points) Explain in detail how you can exploit this vulnerability to get the flag (admin's password).

(b) (2 points) What is the flag?

## Problem 3. Password++ (20 points)

As in the previous challenge, your goal is to find out the password of admin, which is the flag of this challenge.

(a) (1 points) Guess the SQL query used in this web service.

(b) (1 points) What is the HTTP method of the request sent by the browser when you click the "Sign in" button (Hint: Refer to the network tab in the browser developer console)?

(c) (1 points) What information is included in the body content (i.e., payload) of the HTTP request sent by the browser when you click the "Check!" button (Hint: Refer to the network tab in the browser developer console)?

(d) (7 points) Describe the vulnerability in this web service.

(e) (7 points) Explain how you can exploit this vulnerability to get the flag (admin's password). If you used a script to exploit it, please include the script in the writeup.

(f) (3 points) What is the flag?

## Problem 4. Uploader (10 points)

In this challenge, you have to upload an arbitrary PHP file that reads the `/var/www/flag.txt` file via a shell command.

(a) (4 points) Describe in pseudocode what content filtering checks are implemented on the server for uploaded files.

(b) (4 points) Explain how you can exploit this vulnerability to get the flag.

(c) (2 points) What is the flag?

## Problem 5. Uploader++ (20 points)

This challenge has more advanced content-filtering checks than the previous challenge. You have to upload an arbitrary PHP file that reads the `/var/www/flag.txt` file via a shell command.

(a) (8 points) Describe in pseudocode what content filtering checks are implemented on the server for uploaded files.

(b) (8 points) Explain how you can exploit this vulnerability to get the flag.

(c) (4 points) What is the flag?

## Problem 6. Search V1 (5 points)

In this challenge, you need to read the victim's cookie!

(a) (2 points) Describe the vulnerability in this web service. You need to specify which type of XSS this vulnerability is.

(b) (2 points) Explain how you can exploit this vulnerability to get the flag.

(c) (1 points) What is the flag?

## Problem 7. Search V2 (10 points)

In this challenge, Search V1 has been more safer. You need to read the victim's cookie!

(a) (3 points) Compared to Problem 6, what defense mechanism has been added? Describe the added defense logic in detail.

(b) (3 points) Describe the vulnerability in this web service.

(c) (2 points) Explain how you can exploit this vulnerability to get the flag.

(d) (2 points) What is the flag?

## Problem 8. Service Center (20 points)

In this challenge, you need to read the victim's cookie!

(a) (7 points) Describe the vulnerability in this web service. You need to specify which type of XSS this vulnerability is.

(b) (10 points) Explain how you can exploit this vulnerability to get the flag.

(c) (3 points) What is the flag?

## Problem 9. Search V3 (30 points)

In this challenge, Search V2 has been more safer. You need to read the victim's cookie!

(a) (9 points) Describe the vulnerability in this web service. You need to specify which type of XSS this vulnerability is.

(b) (7 points) Explain the deployed defense logic to prevent vulnerabilities.

(c) (10 points) Explain how you can exploit this vulnerability to get the flag.

(d) (4 points) What is the flag?

## Problem 10. Search V4 (35 points)

In this challenge, Search V1 has been more safer. You need to read the victim's cookie!

(a) (9 points) Explain the deployed defense technique to prevent vulnerabilities.

(b) (9 points) Explain how the defense mechanism can defend against vulnerabilities.

(c) (15 points) Explain how you can exploit this vulnerability to get the flag.

(d) (6 points) What is the flag?

## References

[1] CSE610: Web Programming and Security. 2024. 5. Cross-Site Scripting. https://websec-lab.github.io/courses/2024s-cse610/slides/lecture5-xss.pdf.