# Lab 3 – Broken authentication and XSS part 1

*Updated: 2023-10-19.*

## Introduction to the lab

This individual assignment is divided into 3 parts, and it will take 3 classes. The remaining parts will be released in the following weeks.

#### Learning outcomes

- Learn the different strategies that can lead to broken authentication.
- Understand the impact of bad practices regarding authentication flows.
- Learn how to explore the different strategies of cross-site scripting.

#### Submission

You may submit as you go but be sure to complete and submit all activities up to 5 days after the third class (Wednesday at 11:50 pm).
We strongly recommend submitting the work at the end of each class as it is, and then, improving it.

#### Report structure

The report should have (at least) the following contents:

- Scope of this assessment;
- Summary of the activities of each module and evidence that you made each step (with screenshots);
- Print and attach the PDF after concluding the TryHackMe module;
- Answer the questions that are raised in the document.

## 2.1 TryHackMe – HTTP in detail (Optional)

This module is not mandatory. It allows you to refresh some concepts regarding HTTP protocols. If you feel confident with that matter, you should skip it (or not, practice a bit more do not hurt).

## 2.2 TryHackMe – Introduction to OWASP ZAP

This module may be a little outdated, but it should help you configure OWASP ZAP on your machine.

## 2.3 bWAPP

bWAPP is a free and open-source deliberately insecure web application. It helps security enthusiasts, developers and students to discover and to prevent web vulnerabilities. bWAPP prepares one to conduct successful penetration testing and ethical hacking projects. It has over 100 web vulnerabilities!

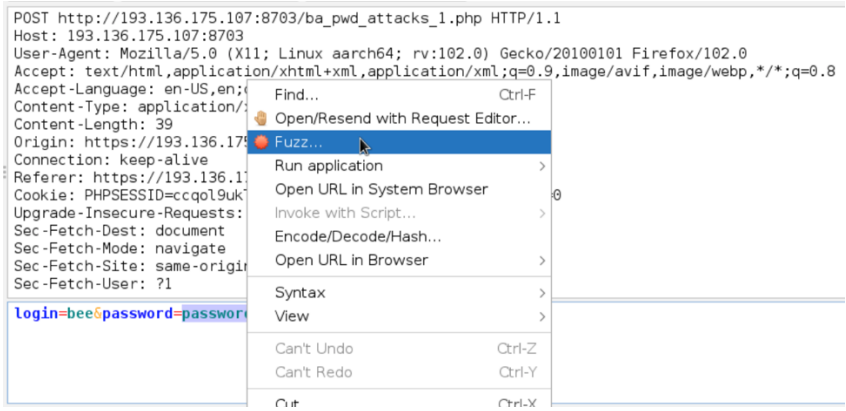
To use it, you must have docker installed and run the following command:

```
docker run -d -p 80:80 hackersploit/bwapp-docker
```

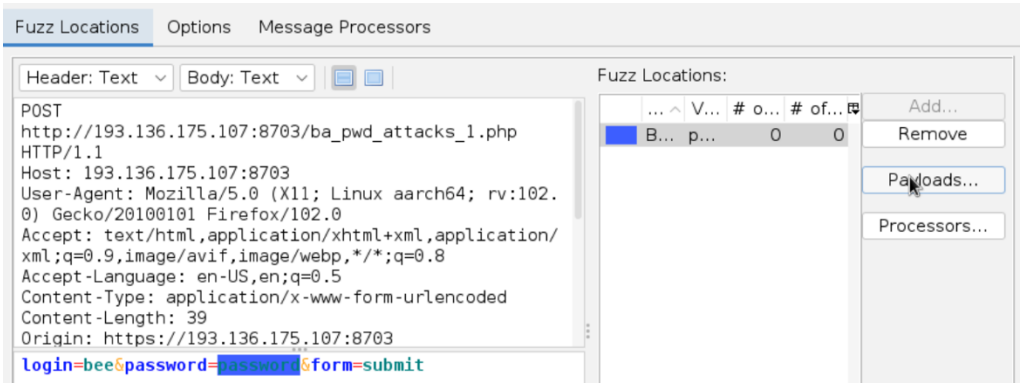
After running the image, navigate to http://127.0.0.1/install.php to complete the bWAPP setup process.


Now that the bWAPP is working, you should solve the following challenges:

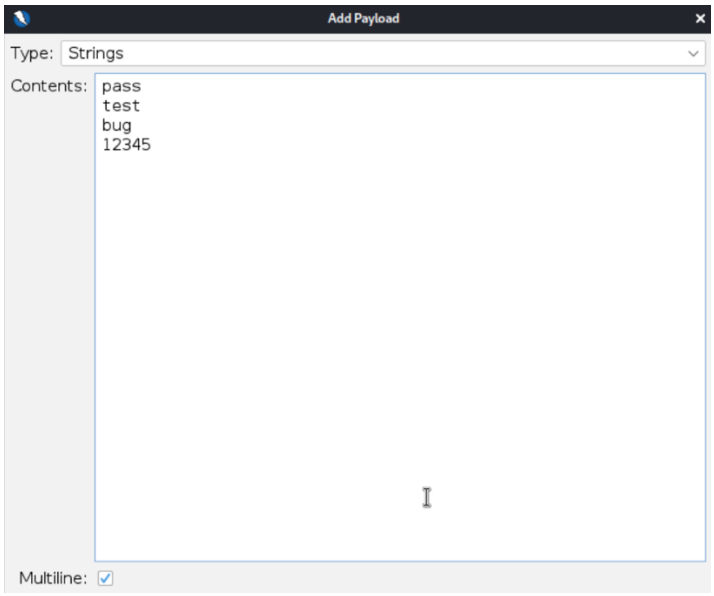
1. Broken Auth. - Password Attacks (You should use OWASP ZAP or Burp suite to solve this challenge).
    a. After intercepting the request, you can fuzz it.



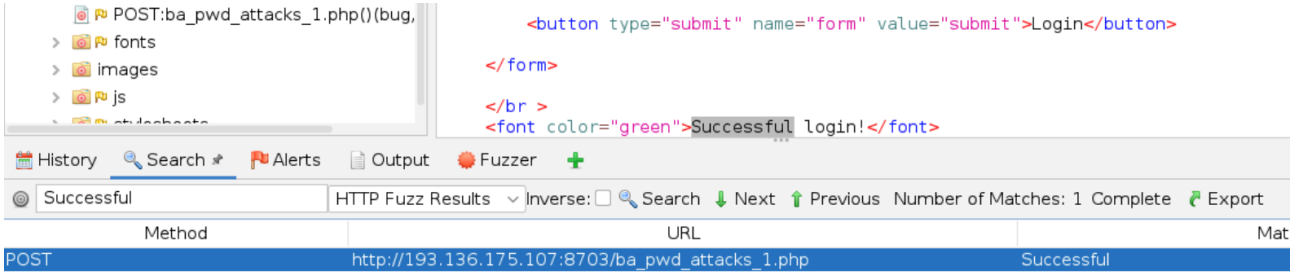

    b. You should select the password value and try to a list of words.



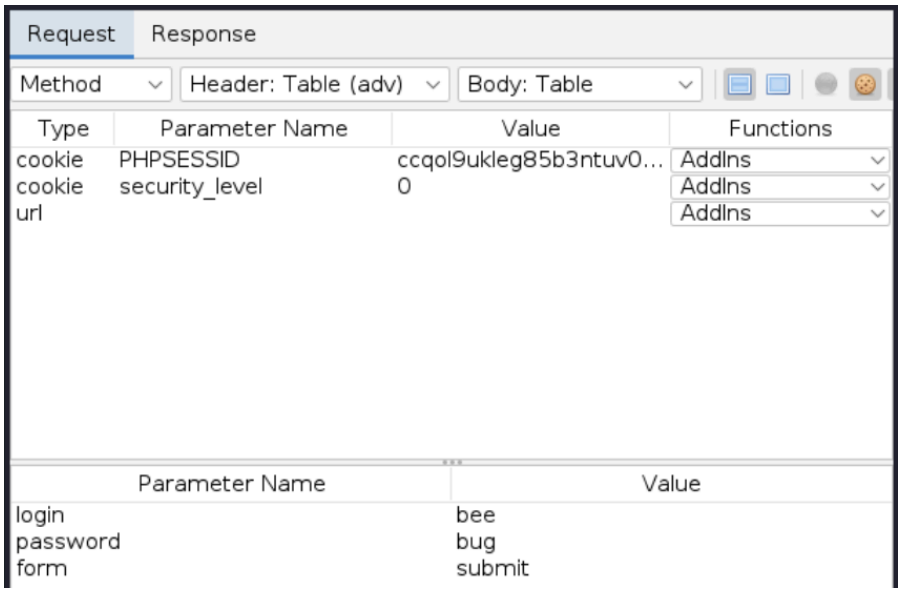

    c. Since we already know the password, make sure that you include “bug” in the list.




d. After executing it, you can filter the results to obtain the response with the message "Successful login!".



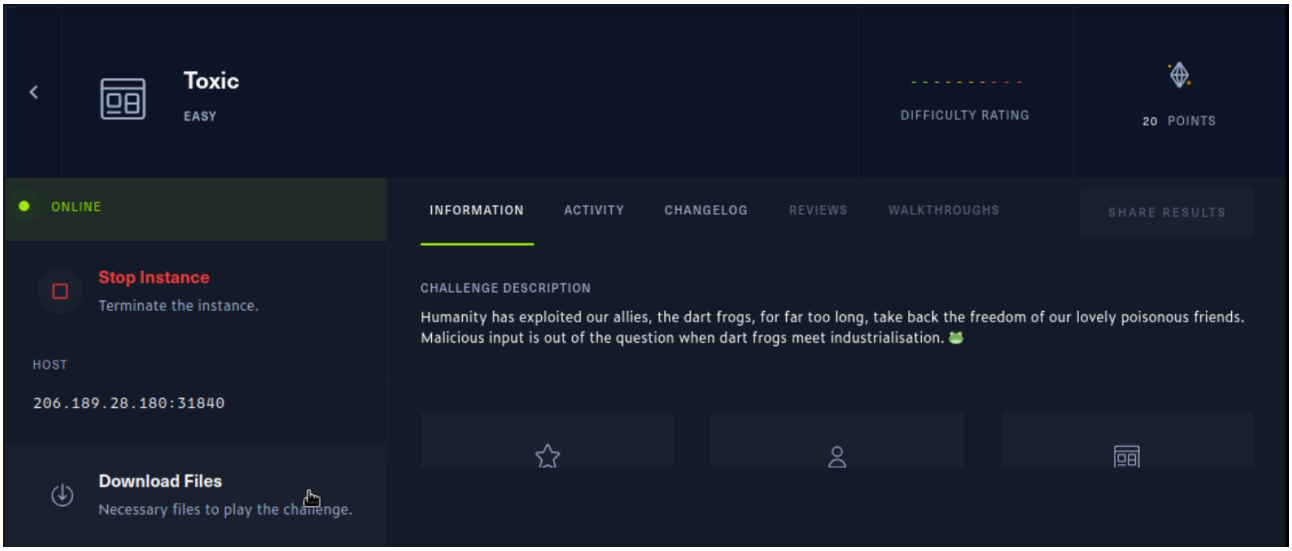

e. Then, just open the information about this request and check the parameters.




2. Session Mgmt. - Cookies (HTTPOnly) Medium
   a. When you click to see your cookies, it only displays two of them. Why?
   b. You should change something to present the three.
3. Session Mgmt. - Session ID in URL
   a. Where is the cookie exposed? Why is it wrong? (https://owasp.org/www-community/vulnerabilities/Information_exposure_through_query_strings_in_url)

## 2.4 HTB Challenge –Toxic

This exercise is a good example of how cookies can be manipulated to exploit a web application.



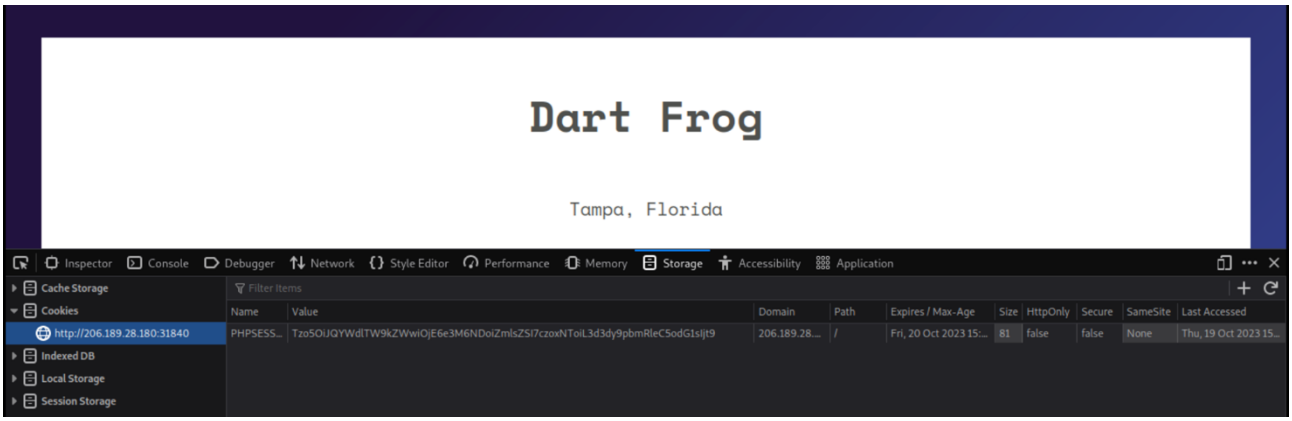

The challenges provide the source files, that you should run in your VM using docker. This will give you a local environment to test the payloads.

```
└─$ ll
total 24
-rwxr-xr-x 1 john john   96 Apr 30  2021 build-docker.sh
drwxr-xr-x 4 john john 4096 Apr 30  2021 challenge
drwxr-xr-x 2 john john 4096 Apr 30  2021 config
-rw-r--r-- 1 john john  718 Jun 22  2022 Dockerfile
-rwxr-xr-x 1 john john  179 Apr 30  2021 entrypoint.sh
-rw-r--r-- 1 john john   27 Apr 30  2021 flag
```

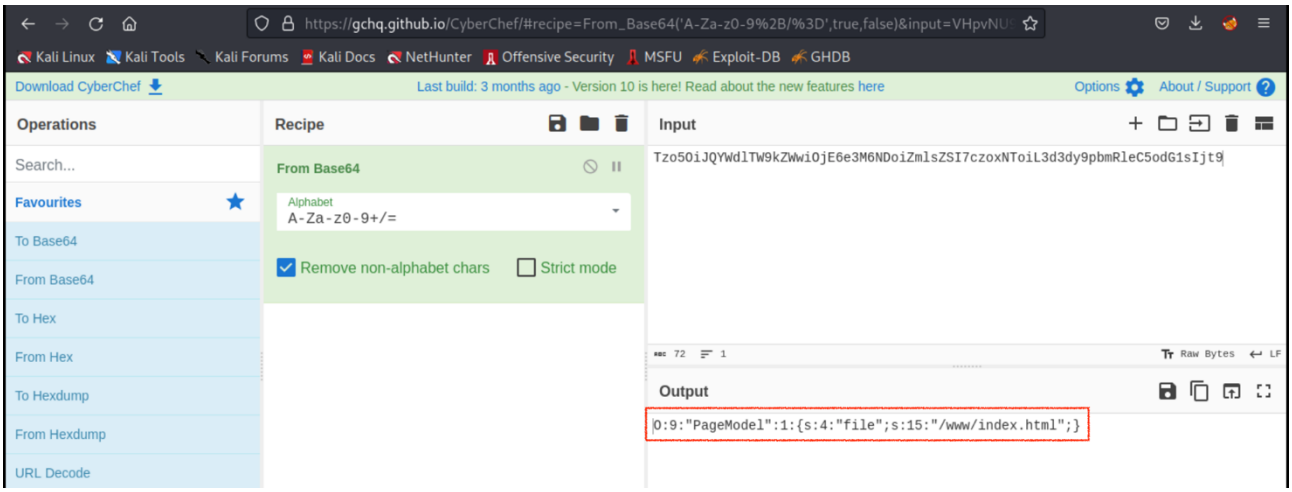
If you analyze the code, you may notice that the cookie is not very well constructed. Why?

```php
if (empty($_COOKIE['PHPSESSID']))
{
    $page = new PageModel;
    $page->file = '/www/index.html';

    setcookie(
        'PHPSESSID',
        base64_encode(serialize($page)),
        time()+60*60*24,
        '/'
    );
}

$cookie = base64_decode($_COOKIE['PHPSESSID']);
unserialize($cookie);
```

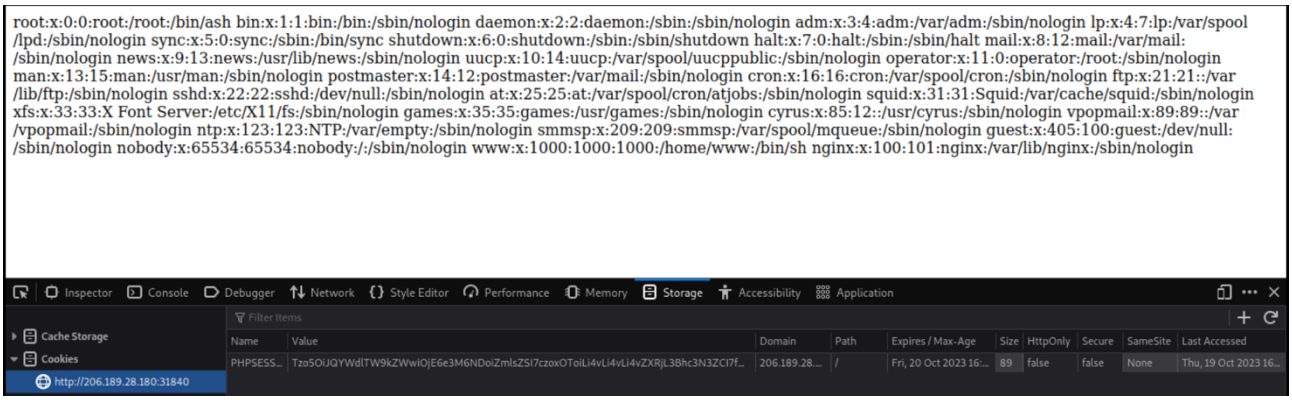
You can check it in the browser.




Let's work with the cookie and try to obtain the information, i.e., reversing it. You can use CyberChef online.




If you manipulate properly the path, you may reveal some files in the system.




Can you obtain the flag?