Introduction to Cyber Security
– Web Security –

Deadline: 2nd February, 2025

Introduction

To provide the World Wide Web, web servers have to supply HTML pages to their users. However, nowadays, the content served on the World Wide Web rarely consists just of statically served web pages - instead, content has become dynamic, enticing interaction with the users ranging from simple web forms up to intricate web applications. New technologies have developed at a rapid pace to suit this growing need of dynamic content. With this diverse technology stack, the classes of vulnerabilities found in web applications has also become quite diverse. This lab deals with several common types of vulnerabilities encountered in the World Wide Web. You will educate yourself in some basic vulnerabilities, like the abuse of missing input filtering and/or user identification, as well as some of the more advanced topics like SQL-Injection and Cross-Site-Scripting (XSS).

To learn about these issues in a hands-on approach, you will be provided with your own personal instance of OWASP Juice Shop, a deliberately insecure web application that is open to all kinds of common attacks on the World Wide Web. The platform provides a set list of challenges that you can work on in any order. Your goal is to execute as many of these attacks as possible, while gaining an understanding of *how* each of these attacks works. If you encounter difficulties, study the given material carefully and do not give up too quickly – after all, tenacity is an important skill for security experts!

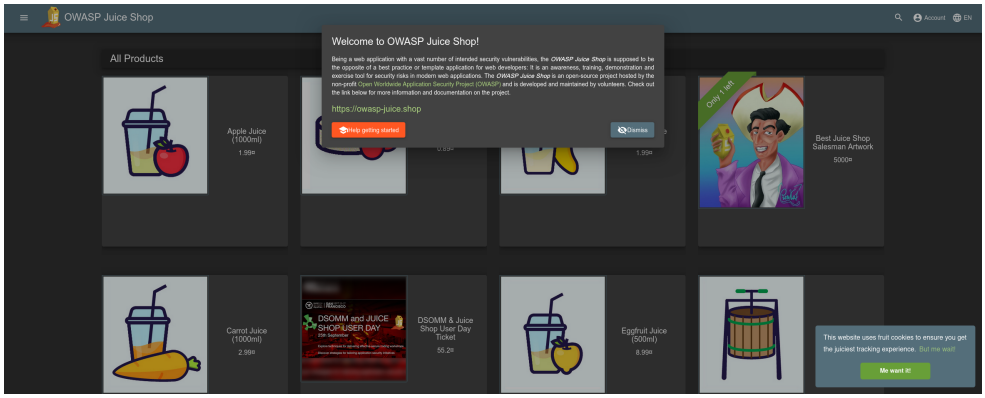
Notes

Please note that these practical tasks assume basic knowledge to have been learned in previous studies. This also holds true for topics that will be covered in the lecture or exercise classes of Introduction to Cyber Security at some point, but have not yet been held. If you find yourself missing the knowledge required to solve this task sheet, you must attain it on your own through the process of self study.

For the purposes of this task sheet, you will need to research several practical types of vulnerabilities in web applications, including SQL injections, XSS vulnerabilities, forgery of JWT tokens, and more. The hidden score board of your OWASP Juice Shop instance will serve as a good starting point for you to find hints on how to approach each of the tasks.

1 Preparation

For this fourth and final lab, you are given access to an instance of OWASP Juice Shop, a deliberately insecure web application developed by the Open Source Foundation for Application Security (OWASP). Like in the previous labs, your instance can only be accessed from *within the BTU network*.[1] The challenge instance can be found on the challenge server "borabora" by browsing to the following address *within your browser*: `borabora.informatik.tu-cottbus.de:<your_personal_port>`. Your personal port is the same as the one you used in the previous labs. When you connect to it, you should be able to see a landing page such as this:



Note: In case your browser reports the error: *"This address is restricted / This address uses a network port which is normally used for purposes other than Web browsing"*, please get in contact with your lab advisor to fix the issue. In such a case, you will be given a different personal port for this lab.

[1] You can either connect directly to *eduroam* on campus or use a VPN connection. If you use Linux as your daily driver, the OpenConnect VPN tool may be of use to you.

2 Main Task

After browsing to your challenge instance, you find yourself at the website of a shop that sells juice and various related items. However, the website is riddled with security issues that are just waiting to be exploited. Your goal is to attack this website as much as possible. In particular, you will solve various challenges that are given to you on this website through a scoreboard. How to reach that scoreboard? That is going to be your first task. You may want to search for hidden sub pages, or hidden / normally invisible information on the frontpage that can help you find the URL. Possibly, the scripts running on the front page can also hold additional information.

Once you have found the scoreboard page, your first challenge is already solved. You are now presented with a list of further attack goals, each categorized by difficulty (number of stars) and general exploit type. Try your hand at as many of these as you can, making sure that you don't just get the attack to work, but that you also understand *how* it works. How could each exploit you achieve come into play on a real website? How could it be avoided? **Your ultimate goal as part of this lab is to finish enough challenges to gain at least 200 points.** Points will not be represented within the score board, but they are awarded by us for every challenge you solved, depending on its difficulty. The following table shows how many points you receive for each challenge you solved by the end of the deadline:

Number of stars	1		2		3		4	5	6
There is a tutorial	✓	✗	✓	✗	✓	✗	✗	✗	✗
Points	1	2	2	6	4	11	16	22	30

Please note that only the hacking challenges count towards your points gained. However, to gain a deeper understanding of the attacks, it is recommended to also work through the coding challenges once you have finished a hacking challenge. To allow you to easily check your current amount of points, we have provided a service that tells you your current score. It can be reached through netcat using the command:

```
~$ nc borabora.informatik.tu-cottbus.de <your personal port + 1>
```

For example, if your personal port is 1005, you can reach the service at port 1006. You will receive an output akin to:

```
~$ You have solved 1 hacking challenge(s), amounting to 1 point(s).
```

Important Note!! Your progress in OWASP Juice Shop is saved both on the server and in a special progress cookie within your browser. Unfortunately, the way in which OWASP Juice Shop is designed *forces* a reset of the server-side progress if the server ever is restarted. In this case, only your personal progress cookie can restore your progress. The failsafes we could implement to counteract this issue are not ideal. Therefore, to best protect your progress even in the case of a server crash (e.g. due to very unusual payloads), please **make sure that your cookies for the challenge instance are not deleted**!

Some tools that may be of use in your web hacking journey include, but are certainly not limited to:

- The development console of your browser (launched using F12)
- The OWASP Zed Attack (Zap) Proxy
- The Burp Suite Proxy

Final Note: OWASP Juice Shop is a very popular tool for demoing exploits and learning about web security, with many resources relating to it available on the web. Still, to learn the most from this lab, we urge you to try solving the challenges provided with as little help as possible. We won't fault you if you took enough time to try, got stuck after exhausting every idea and finally took a glance at the hints and resources to make some progress - in fact, this way of working can be a good learning experience if afterwards, you take some time to reflect on how you could have adjusted your approach to solve the task on your own. Researching the *general concepts* of the vulnerabilities you work with is of course allowed and even greatly encouraged!

3 Submission and Lab Defense

You do not need to submit anything for this lab. We will investigate your score board and export an overview over the challenges you solved on our own once the deadline is over. Only the challenges that have been solved on your personal instance by the end of the deadline count!

Prepare yourself for a lab defense of up to 30 minutes. In the lab defense, we will go through your solutions and discuss the way in which you solved the tasks. Ensure that you are familiar with all of the concepts that play a role within this lab and are able to defend why you took each step you took.

For this particular defense, we are going to ask you to demonstrate an arbitrary selection of the attacks / challenges you successfully worked on during the lab. We are going to make this selection from the list of challenges that are marked as solved in your score board by the end of the deadline. You should be able to explain your attack, taking special care as to *why* the attack is working. In addition, you should think about ways to prevent the vulnerabilities you exploit. In case you utilized special tools for your attacks, you must be aware of what the tool does to achieve its task and possible implications for your attacks. The examiners might also ask conceptual questions around the main vulnerabilities playing a role in this lab.