**ETHICAL HACKING INTERNSHIP**

**Project Problem Statement**

Every task in this problem is linked with each other so you have to make sure you follow every task in the given serial order. If you skip any of the tasks you can’t follow up the other.

**Task 1:**

In Session 22 we introduced you to portswigger labs. Portswigger is a website which has so many vulnerable labs which helps you to learn about other vulnerabilities in real life. You can visit Portswigger labs at <https://portswigger.net/>

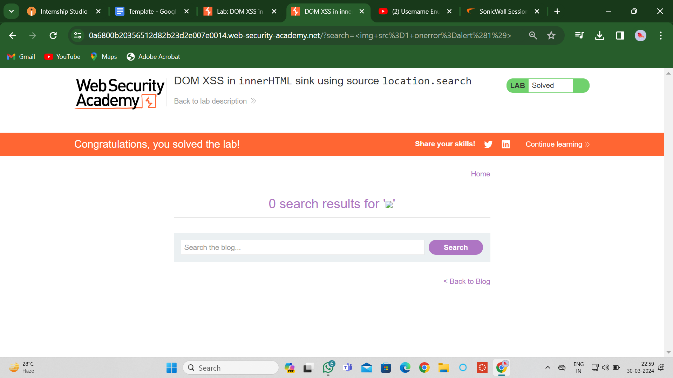
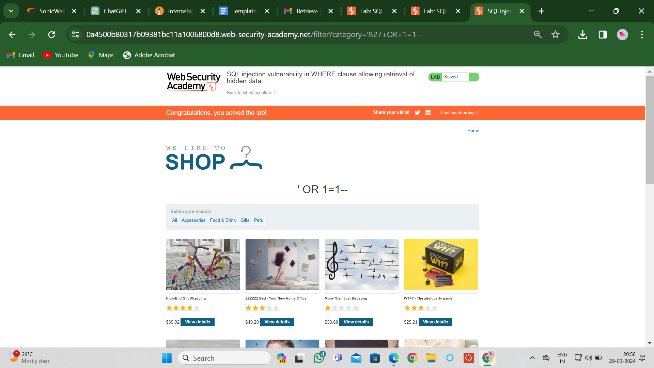
So the exact task for you now is there are several XSS labs on this website <https://portswigger.net/web-security/all-labs>. You can just choose any 5 of them and solve it. We are leaving the choice up to you.

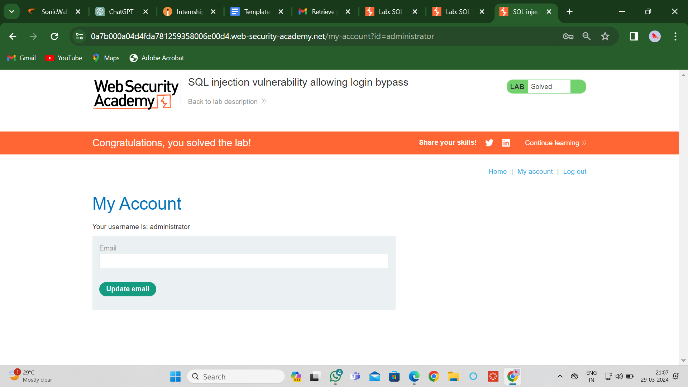
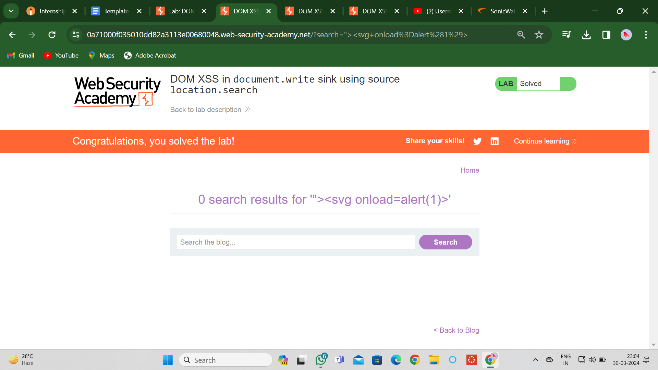
Every lab on the website has a hint section which you can use to solve the labs if you are stuck somewhere. Watch me solve one lab to give you a demo.

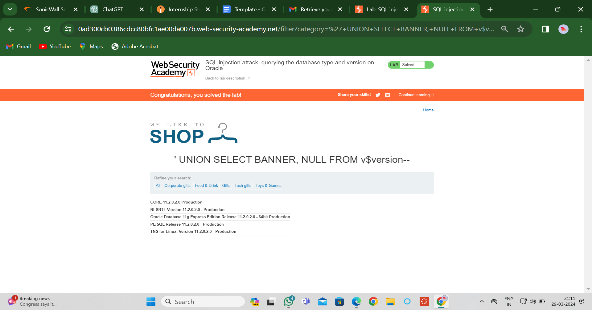
After solving you should see something like “Solved Status” on the top of the lab. That status is necessary to pass the task out.

If you need any more help solving labs, you can use Google to find out a solution video available on Youtube..

**Solution:**

****





**Task 2:**

In this task you are completely free. <http://testasp.vulnweb.com/>  - This is the website. Explore the website and try to find vulnerabilities in the website and report it to us. You will be evaluated on your methods and the report you submit. Don’t worry about evaluation, just report the vulnerabilities as you feel comfortable.

Make sure your report matches this >> [#751870 Reflected XSS in pubg.com (hackerone.com)](https://hackerone.com/reports/751870)

You are expected to include the following in the final Report that you have to submit.

1) Proper Steps

2) ScreenShot

3) Video

Solution:

**Vulnerability Report: Cross Site Scripting (XSS) on** [**http://zero.webappsecurity.com**](http://zero.webappsecurity.com)

(video is attached with the pdf report)

**Target Application**: <http://zero.webappsecurity.com>

Domain: Vulnweb.com

Subdomain: testasp.vulnweb.com

**Scan Summary**:

Overall Risk Level: Medium

Start Time: 20:24:31 (GMT+5:30)

Finish Time: 20:24:48 (GMT+5:30)

Scan Duration: 17 seconds

Tests Performed: 18/18

**Findings:**

1. Vulnerabilities Found for Server-Side Software
2. Communication is Not Secure
3. Missing Security Headers
4. Security.txt File is Missing

**Main Vulnerability:**

The primary vulnerability identified in the target application is related to the usage of an outdated version of the jQuery library (version 1.8.2). Several Common Vulnerabilities and Exposures (CVEs) have been associated with this version, including CVE-2012-6708, CVE-2020-11022, CVE-2020-11023, CVE-2019-11358, and CVE-2015-9251. These vulnerabilities expose the application to Cross-Site Scripting (XSS) attacks, allowing attackers to inject and execute malicious scripts within the application context.

**Risk Assessment:**

The presence of these vulnerabilities poses a significant risk to the security and integrity of the target application. Exploitation of the XSS vulnerabilities in the outdated jQuery library could lead to unauthorized access to sensitive data, manipulation of user sessions, and potential complete compromise of the application's functionality. Additionally, the absence of secure communication protocols and missing security headers further exacerbate the risk of exploitation by malicious actors.

**Recommendations:**

1. Update jQuery Library: Upgrade the jQuery library to the latest version (preferably version 3.5.0 or later) to mitigate the risk of XSS attacks associated with the identified CVEs.
2. Implement HTTPS: Configure the web server to use HTTPS to encrypt communication between the web browser and the server, reducing the risk of data interception and tampering.
3. Add Security Headers: Configure the server to include essential security headers such as Content-Security-Policy, Referrer-Policy, and X-Content-Type-Options to mitigate various attack vectors and enhance the overall security posture of the application.
4. Implement Security.txt File: Create and implement a security.txt file according to the standard guidelines, providing a designated channel for reporting vulnerabilities and security concerns.

**Conclusion:**

Addressing the identified vulnerabilities and implementing the recommended security measures is crucial to safeguarding the target application against potential security threats and ensuring the protection of user data and privacy. Regular security assessments and proactive measures are essential to maintaining a robust and secure web application environment.

This vulnerability report is based on the findings of a security scan conducted on http://zero.webappsecurity.com .

Reported by: [KHUSHBOO KUMARI]

Email-Id:[khushboo831015@gmail.com]

Date of Report: [01.04.24]