

**UNIVERSITY INSTITUTE OF ENGINEERING**

**Department of Computer Science & Engineering**

**Subject Name:** Web and Mobile Security Lab

**Subject Code:** 20CSP-338

**Submitted to: Submitted by:**

Er. Himanshi Name: Sahil Kaundal

UID: 21BCS8197

Section: 616

Group: A

**INDEX**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Ex. No** | **List of Experiments** | **Conduct (MM: 12)** | **Viva**  **(MM: 10)** | **Record (MM: 8)** | **Total**  **(MM: 30)** | **Remarks/Signature** |
| 1.1 | Open any website on computer system and identify http packet on monitoring tool like Wireshark. |  |  |  |  |  |
| 1.2 | Design a method to simulate the html injection and cross site scripting to exploit the attackers. |  |  |  |  |  |
| 1.3 |  |  |  |  |  |  |
| 2.1 |  |  |  |  |  |  |
| 2.2 |  |  |  |  |  |  |
| 2.3 |  |  |  |  |  |  |
| 2.4 |  |  |  |  |  |  |
| 3.1 |  |  |  |  |  |  |
| 3.2 |  |  |  |  |  |  |
| 3.3 |  |  |  |  |  |  |

**Experiment 2**

**Student Name:** Sahil Kaundal **UID:** 21BCS8197

**Branch:** BE CSE (Lateral Entry) **Section/Group:** 616/A

**Semester:** 5th **Date of Performance:** 22/08/2022

**Subject Name:** WMS Lab **Subject Code:** 20CSP-338

1. **Aim/Overview of the practical:**

Design a method to simulate the html injection and cross site scripting to exploit the attackers.

1. **Task to be done/ Which logistics used:**

To test HTML and XSS injection

**3. Apparatus / Simulator Used:**

* Windows 7 & above version.
* bWAPP
* Google Chrome
* Encoder Decoder

**Introduction:**

[**Acunetix**](https://www.acunetix.com/web-vulnerability-scanner/?utm_medium=3rdparty&utm_source=softwaretestinghelp&utm_campaign=cross-site-scripting) is a web application security scanner that gives you a 360-degree view of the organization’s security. This end-to-end web security scanner can identify over 7000 vulnerabilities like XSS and misconfigurations. It has capabilities for scanning all pages, web apps, complex web applications, etc.

Acunetix offers specialized technologies that let you detect more and fix faster

## Html Injection

* The attacker finds the vulnerable web application.
* The attacker sends the modified URL to the user by any means, usually via email. This URL has text injected.
* By clicking on the URL user is navigated to the attackers webpage, looks like legitimate one.
* User asked the information like username, password, card pins etc.
* This information gets transferred to the attackers server.

for example

**www.testing.com/siteAdcontent?divMessage=**<h1>Click Here!!</h1> It is possible to modify it as −

* **www.testing.com/siteAdcontent?divMessage=**<hack><h1>Do not Click!!</h1><hack>

**Cross Site Scripting (XSS)**

* It happens whenever an application takes untrusted data and sends it to the client browser without validation. This allows attackers to execute malacious scripts in the victim's browser which can result in user sessions hijack, defacing web sites or redirect the user to malicious sites.

**Types of XSS**

* **Stored XSS -** Stored XSS also known as persistent XSS occurs when user input is stored on the target server such as database/message forum/comment field etc. Then the victim can retrieve the stored data from the web application.
* **Reflected XSS -** Reflected XSS also known as non-persistent XSS occurs when user input is immediately returned by a web application in an error message/search result or the input provided by the user as part of the request and without permanently storing the user provided data.
* **DOM Based XSS –**This type of attack occurs when the DOM environment is being changed, but the client-side code does not change. When the DOM environment is being modified in the victim’s browser, then the client-side code executes differently

**4. Program/ Steps/ Method:**

1. open the link in your browser <http://128.198.49.198:8102/mutillidae/index.php?page=home.php&popUpNotificationCode=HPH0>.

OR

open this website **OWASP Mutillidae II: Web Pwn in Mass Production**

Now, we’ll be redirected to the web page which is suffering from an **HTML Injection vulnerability**which allows the user to submit his entry in the blog as shown in the screenshot.

2. Now, let’s try to inject malicious code Enter the HTML code inside the given text area in order to set up the HTML attack.

3. That html code is thus now into the application’s web server, which gets rendered every time whenever the victim visits this malicious page, he’ll always have this code which looks official to him.

**With the help of Bwapp:**

* Go to bWAPP site and login on it.
* Select HTML Injection (Reflected Get)
* Enter First name and Last name using HTML tags
* Encrypt first name and last name with the help of Encoder Decoder (In case when security level is set on medium)
* Click on Go

**XSS attack:**

1. open the link <https://xss-game.appspot.com/level1>

or Google XSS game website

2. If the search field is vulnerable, when the user enters any script, then it will be executed.

Consider, a user enters a very simple script as shown below:

<script>alert(‘ XSS attack’)</script>

1. Then after clicking on the **“Search”** button, the entered script will be executed

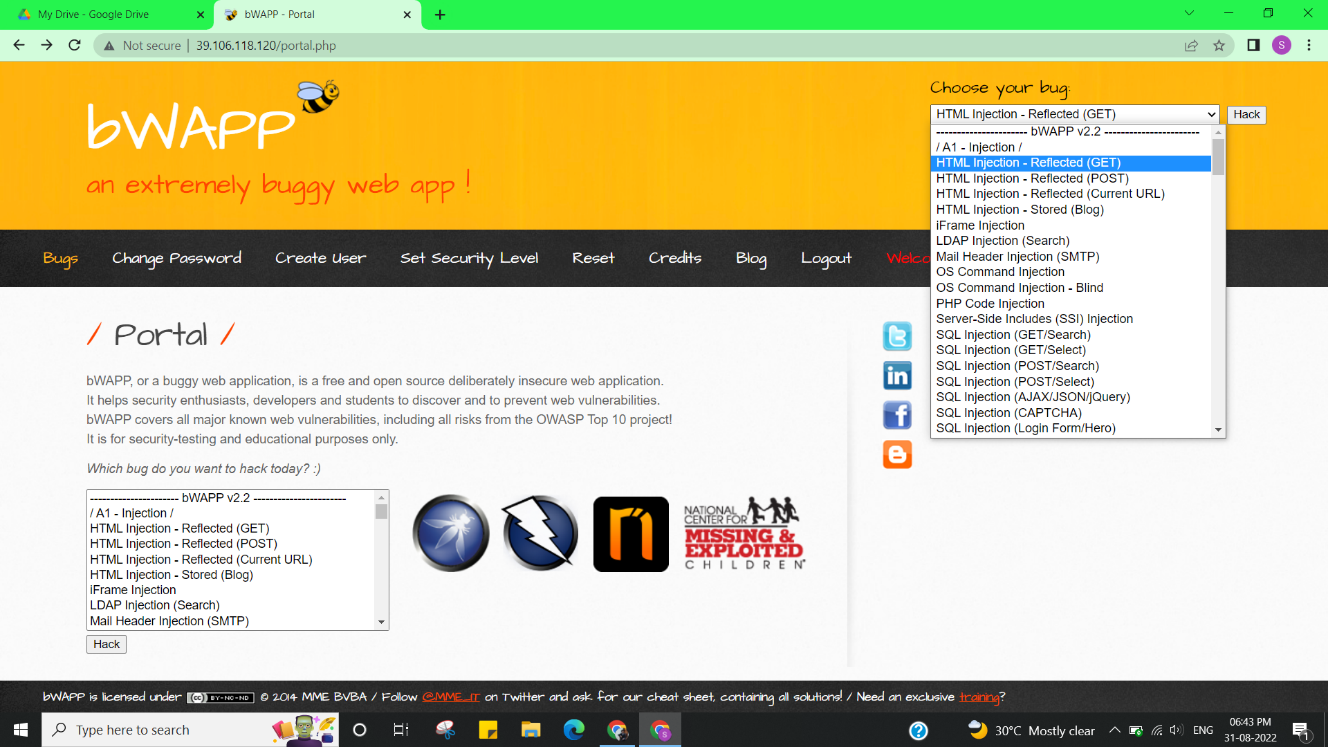
 the script typed into the search field gets executed. This just shows the vulnerability of the XSS attack. However, a more harmful script may be typed as well.

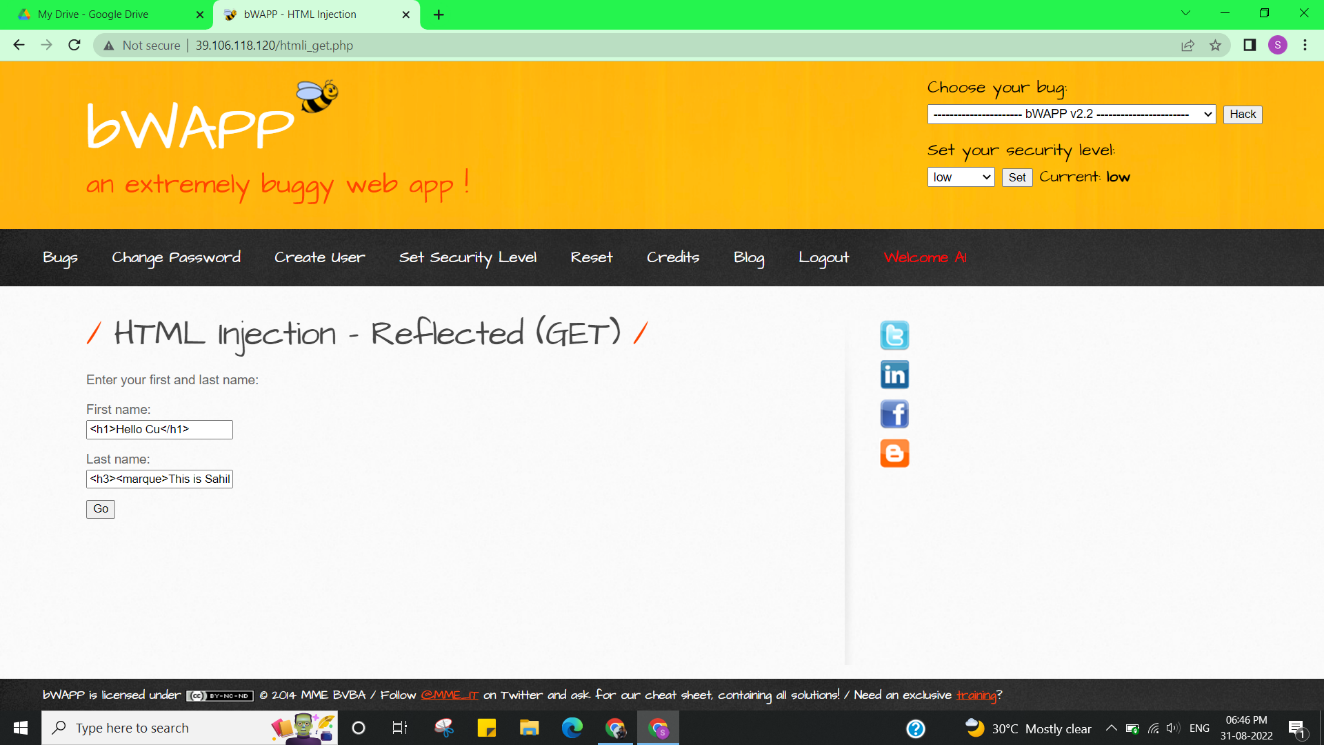
**With the help of Bwapp:**

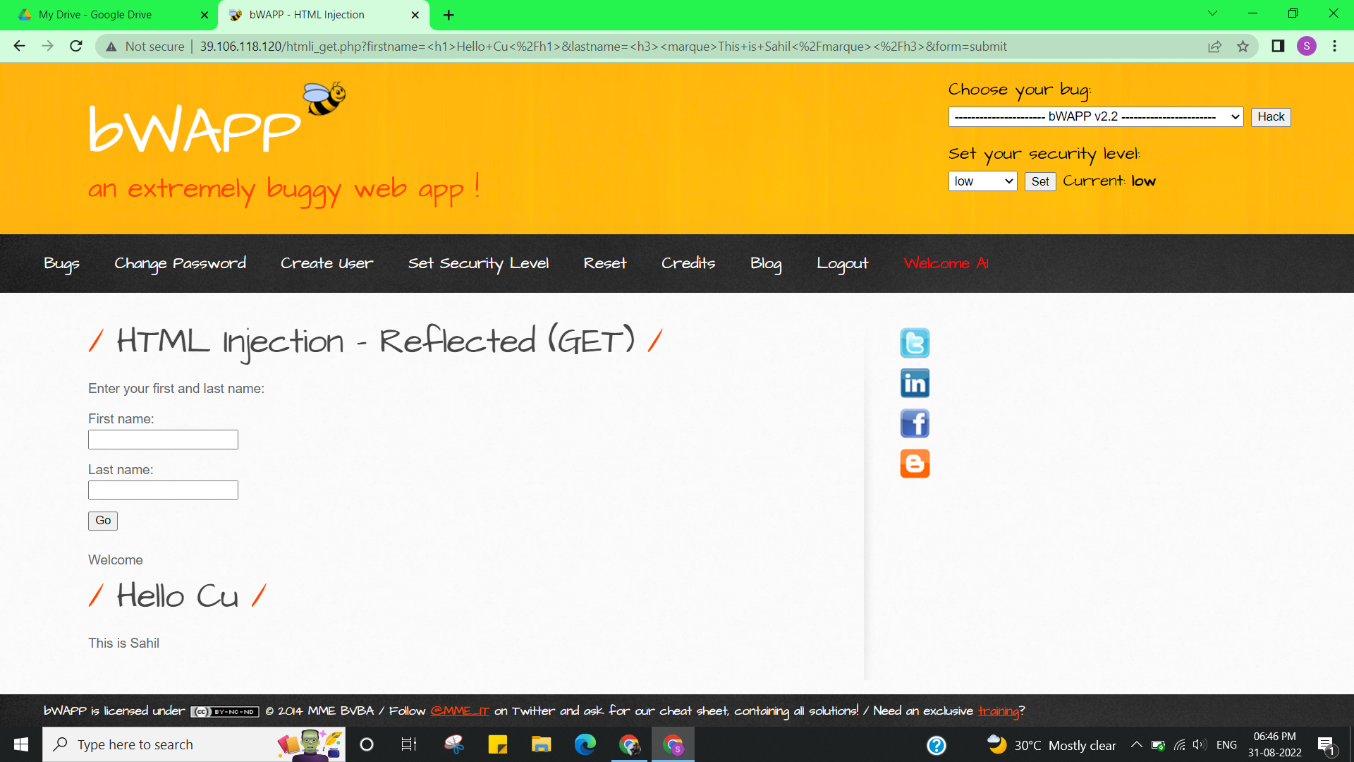
* Go to bWAPP site and login on it.
* Select Cross Site Scripting (Reflected Get)
* Enter First name and Last name using javascript
* Click on Go

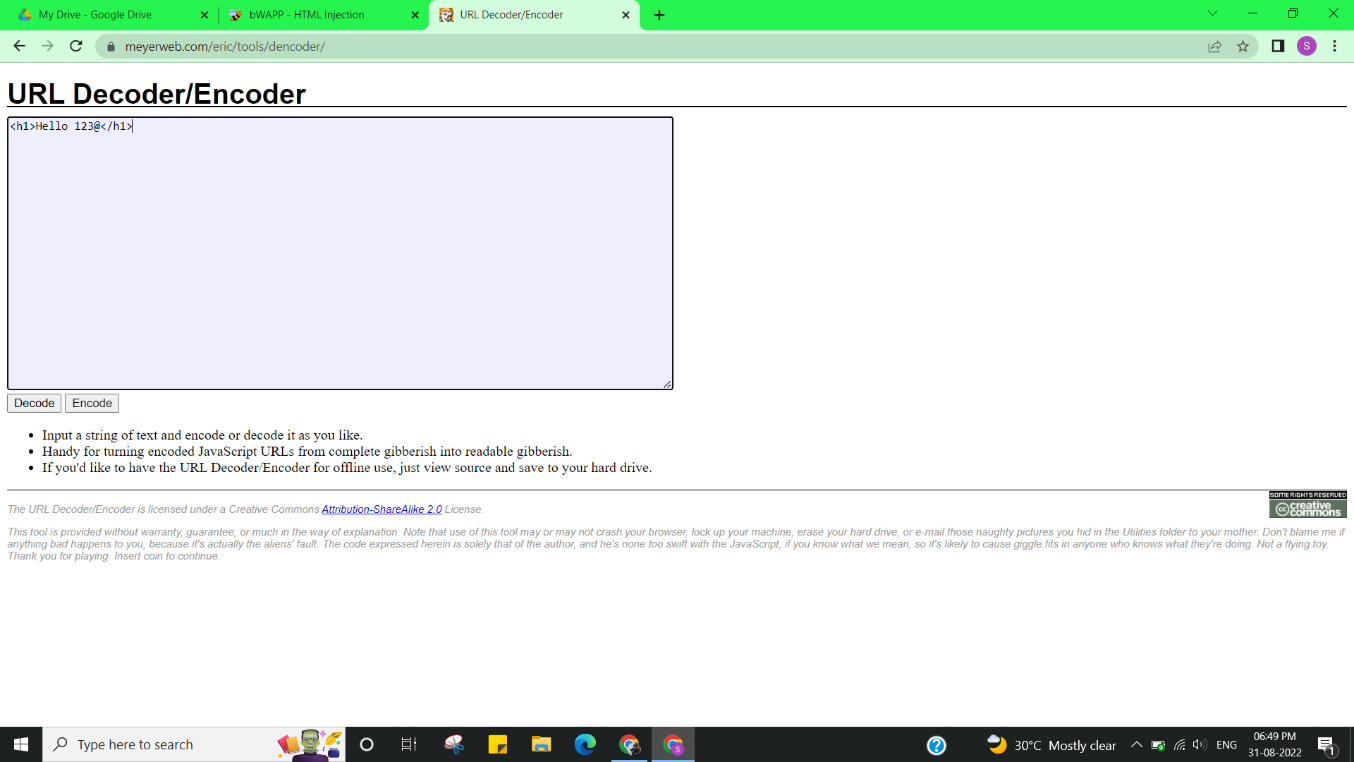
1. **DBMS script/Result/Output/Writing Summary:**

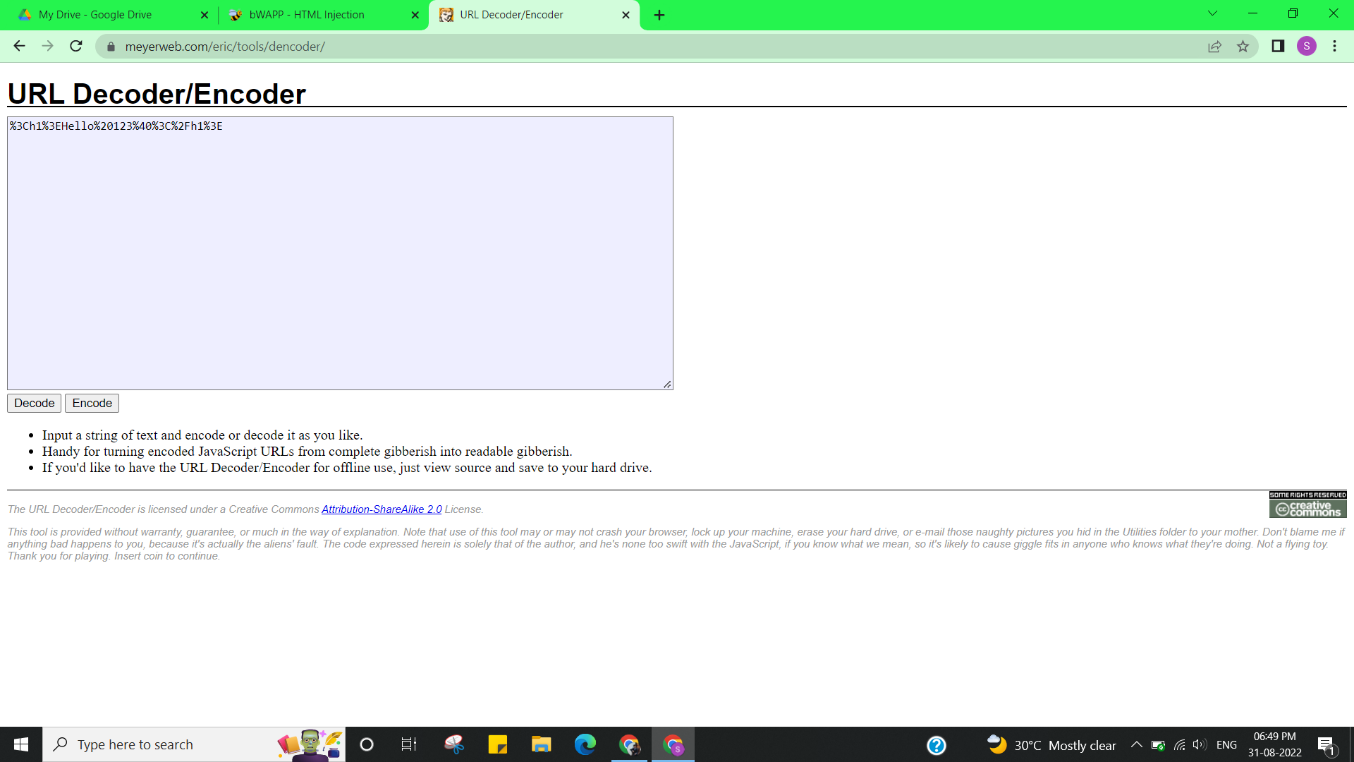
HTML Injection

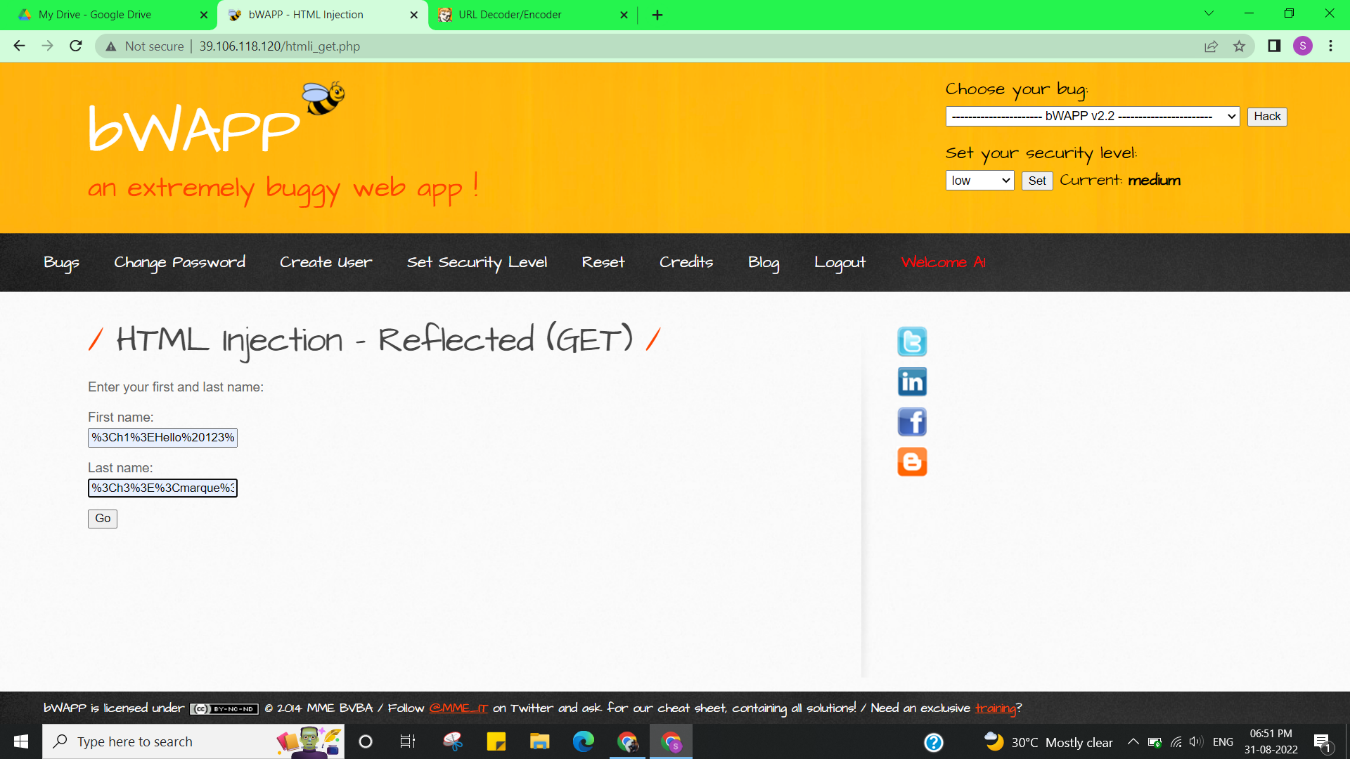


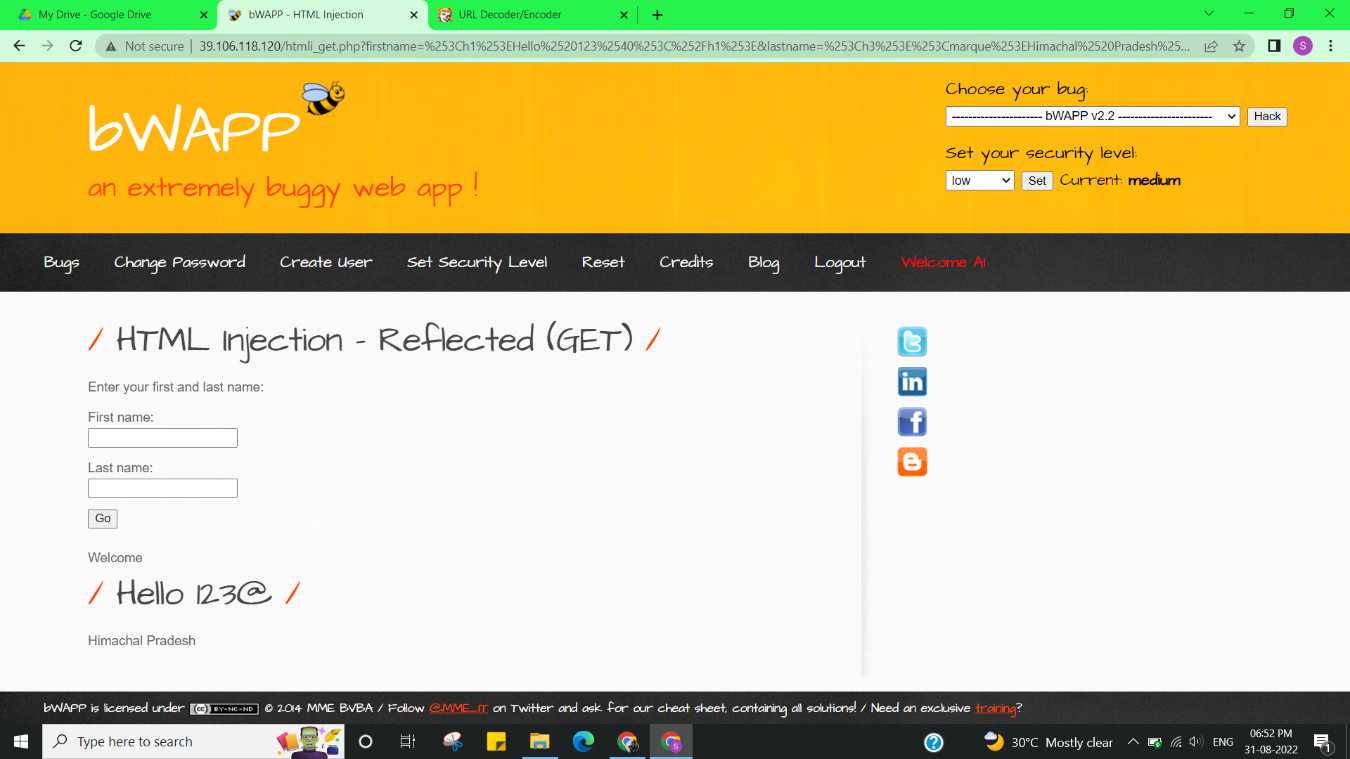




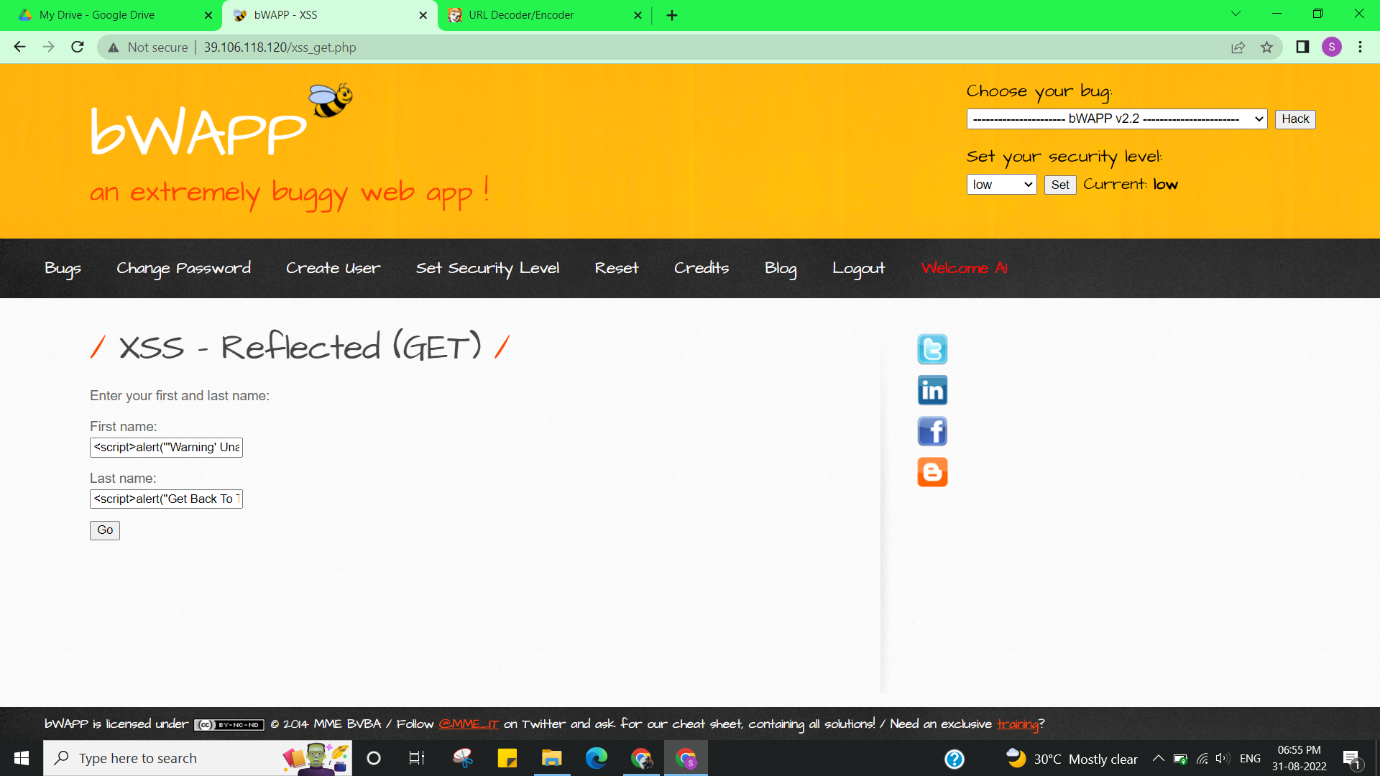


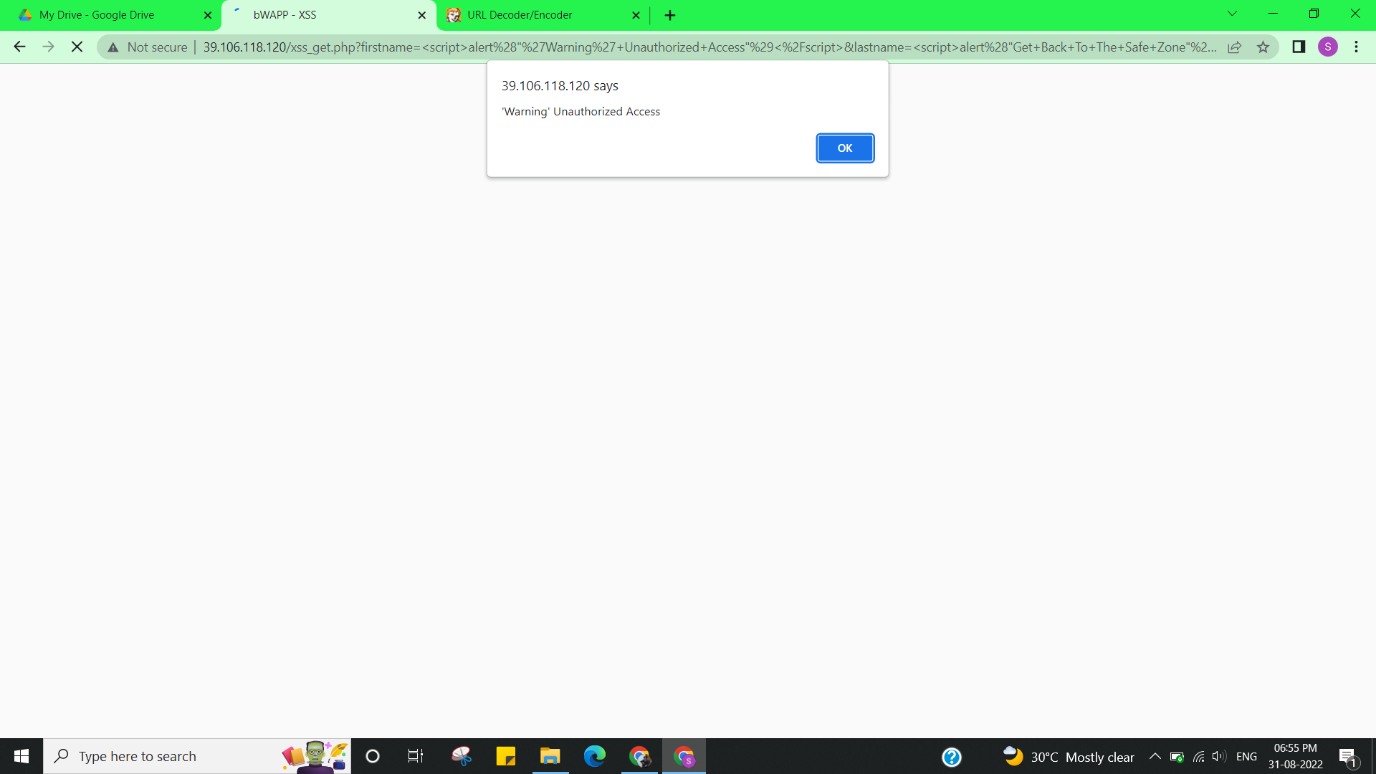


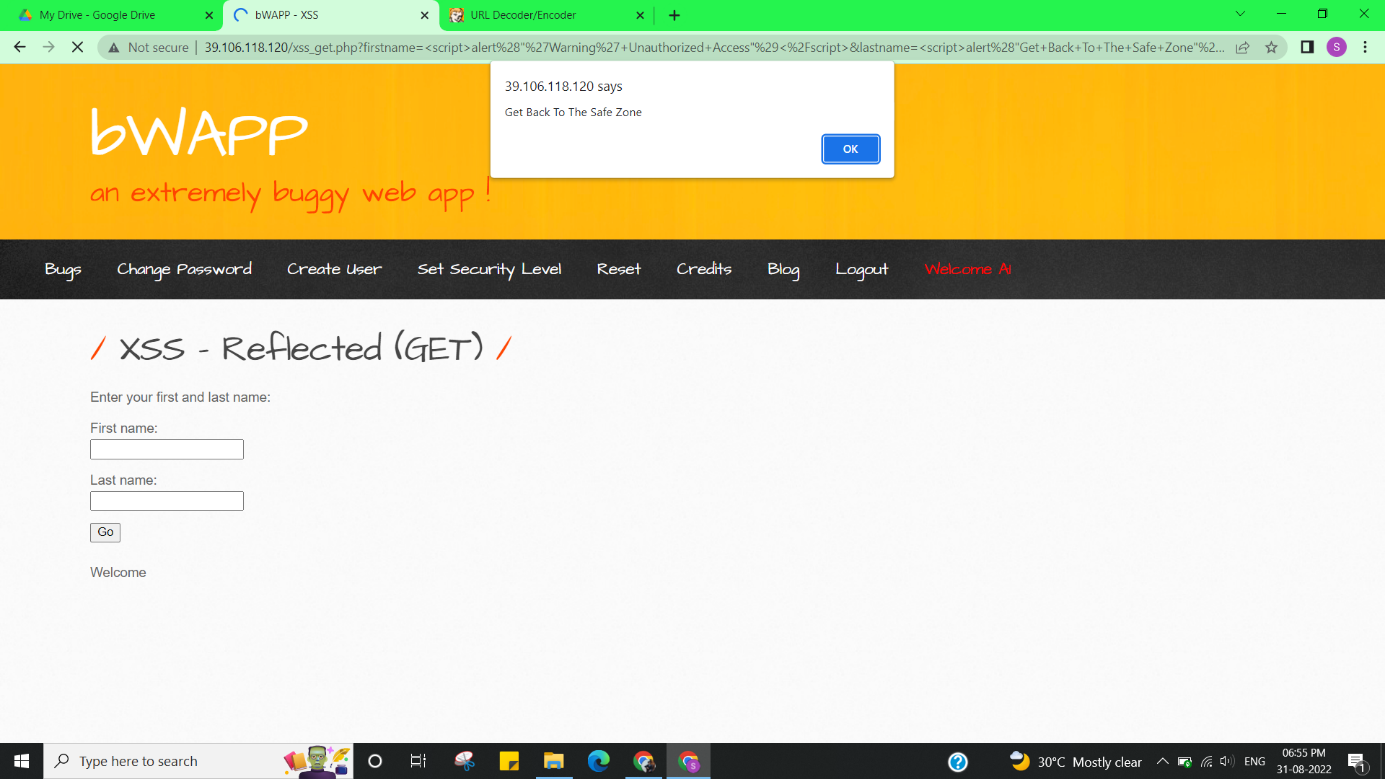




**XSS attack output**







**Learning outcomes (What I have learnt):**

1. Design a method to simulate the html injection and cross site scripting to exploit the attackers.
2. We learn what is html injection and XSS injection. An overview of how these attacks are constructed and applied to real system. If the app or website lacks proper data sanitization, the malicious link executes the attacker's chosen code on the user's system. As a result, the attacker can steal the user's active session cookie and can be the harmful for the website.

**Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
| 1. |  |  |  |
| 2. |  |  |  |
| 3. |  |  |  |
|  |  |  |  |