TASK REPORT

CYBERSECURITY INTERNSHIP

Submitted by

Satvik Shrivastava

23EO5-ST#IS#6653

Under the Supervision of

UPENDRA

Senior Security Analyst

KRISHNA

Junior Security Analyst

Registered And Head Office

D.NO: 11-9-18, 1st Floor,

Majjivari Street, Kothapeta,

Vijayawada - 520001.

+91 9550055338 / +91 7901336873

contact@suprajatechnologies.com

Table of Contents:

Introduction

Throughout my cybersecurity internship at Supraja Technologies, I was tasked with a comprehensive series of assignments that covered a wide spectrum of ethical hacking and cybersecurity practices. These assignments were designed to provide hands-on experience and practical skills in various areas of information security. I began with vulnerability assessments, identifying websites susceptible to common vulnerabilities such as Directory Traversal, HTML Injection, and File Upload flaws. The tasks then progressed to more complex challenges, including SQL Injection attacks, discovering Business Logic Errors, and exploiting Insecure Design Flaws. I gained experience with professional tools like Nessus and Acunetix for conducting network and web application scans. The assignments also covered specific attack techniques such as Parameter Tampering, Authentication Bypass, and various types of injection vulnerabilities including Host Header, iFrame, and Cross-Site Scripting (XSS). In addition to web application security, I delved into network security by performing subdomain enumeration, creating phishing pages, and identifying vulnerable protocols for sniffing attacks. Server hacking exercises involved exploiting specific targets and cracking passwords. I also explored Denial of Service (DoS) attacks, both theoretically and practically, using tools like Goldeneye and observing traffic with Wireshark. The assignments touched on system hardening by manipulating firewall rules and antivirus settings. Advanced topics included using Metasploit for creating backdoors and a comprehensive challenge involving hidden messages, file analysis, and exploiting vulnerable systems to find specific flags. This diverse range of assignments provided a well-rounded experience in various aspects of cybersecurity, from basic vulnerability assessment to advanced exploitation techniques, preparing I for real-world challenges in the field of information security.

23EO5-ST#IS#6653-Task1

A) Use Google Dorks to retrieve 3 websites of any country, after that collect the information of the above collected websites using the below mentioned tools:

- Whois
- ReverseIP Lookup
- Wappalyzer

STEP - 1 Using Google Dorks to retrieve 3 websites of a specific country.

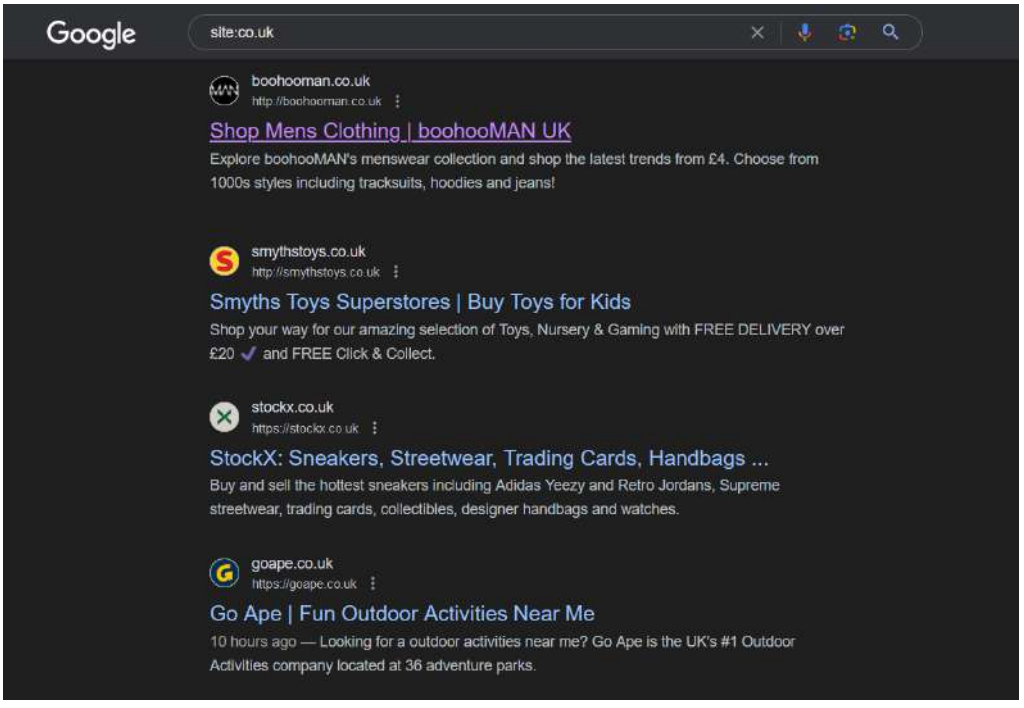
1.1 Open your favourite Browser.

1.2 On Google Search Engine, enter a Google Dork query to filter websites based on the country you want to target. Search **site:co.uk**

1.3 Scroll through the search results and identify any three websites you want to collect information about.

We are choosing-

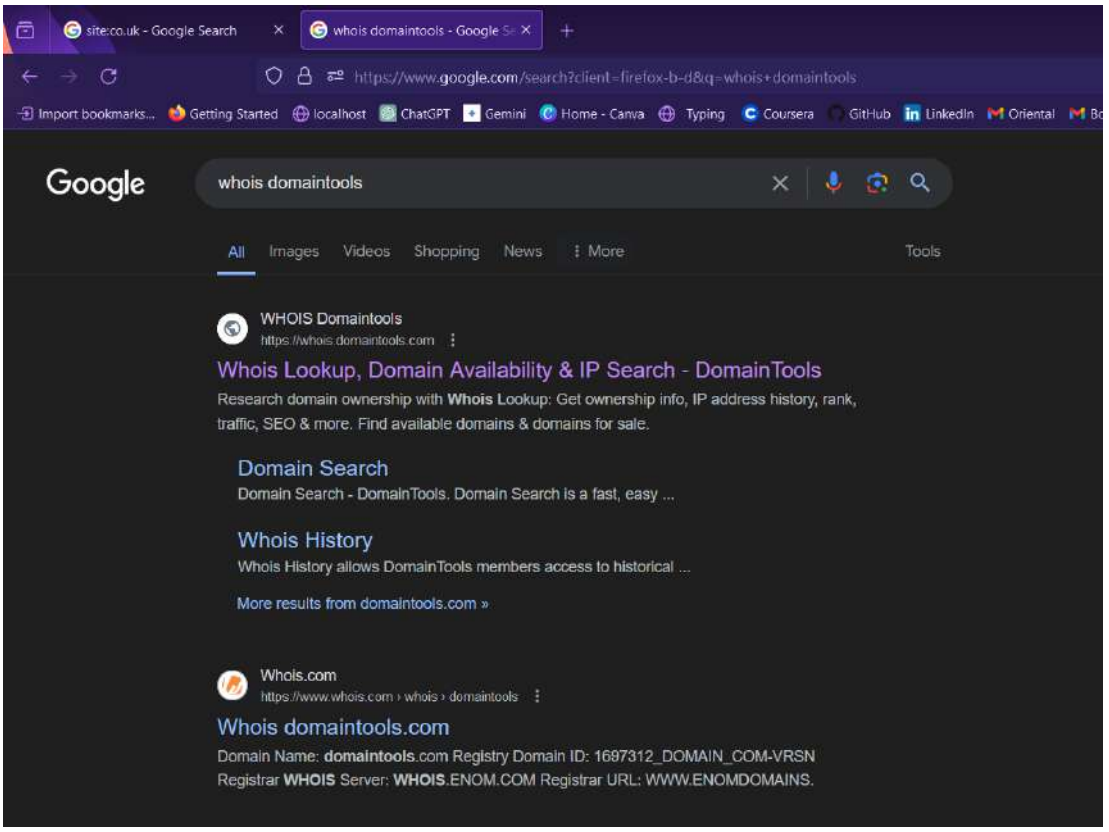
- boohooman.co.uk
- smythtoys.co.uk
- goape.co.uk



STEP -2 Collecting information about the selected websites using the following tools:

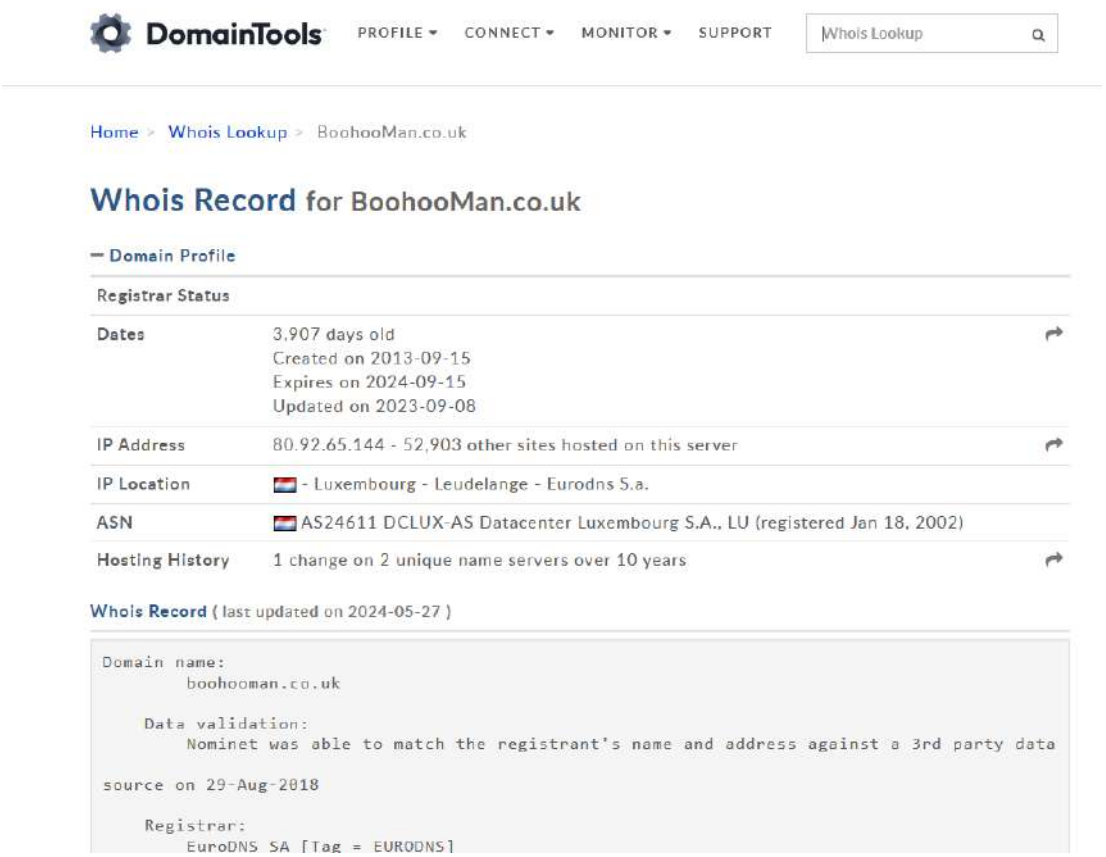
1. **Whois** - It provides information about domain name registrations.

2.1.1 On your search engine type **whois domaintools**. And open the first link.



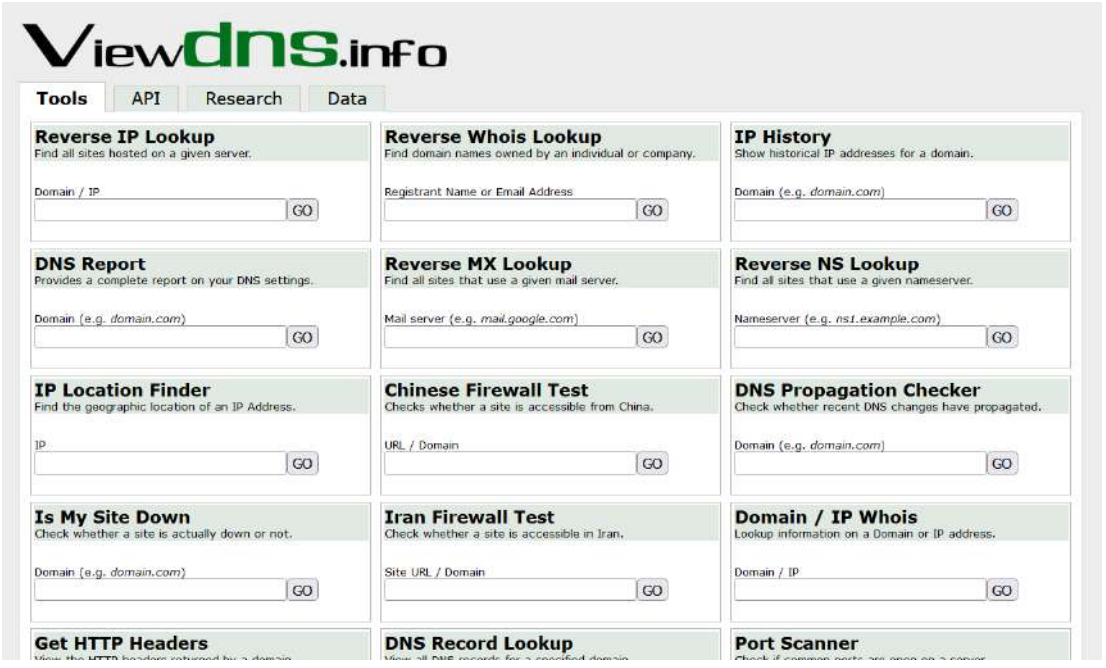
2.1.2 Just put in the domain you want to gather information about. And Click Search Button.

2.1.3 The Entire Domain Information will appear on your screen.



2. **Reverse IP Lookup -** Find all websites that are hosted on a particular IP Address.

2.2.1 Go to https://viewdns.info/ in your Browser. And click on Reverse IP Lookup section.



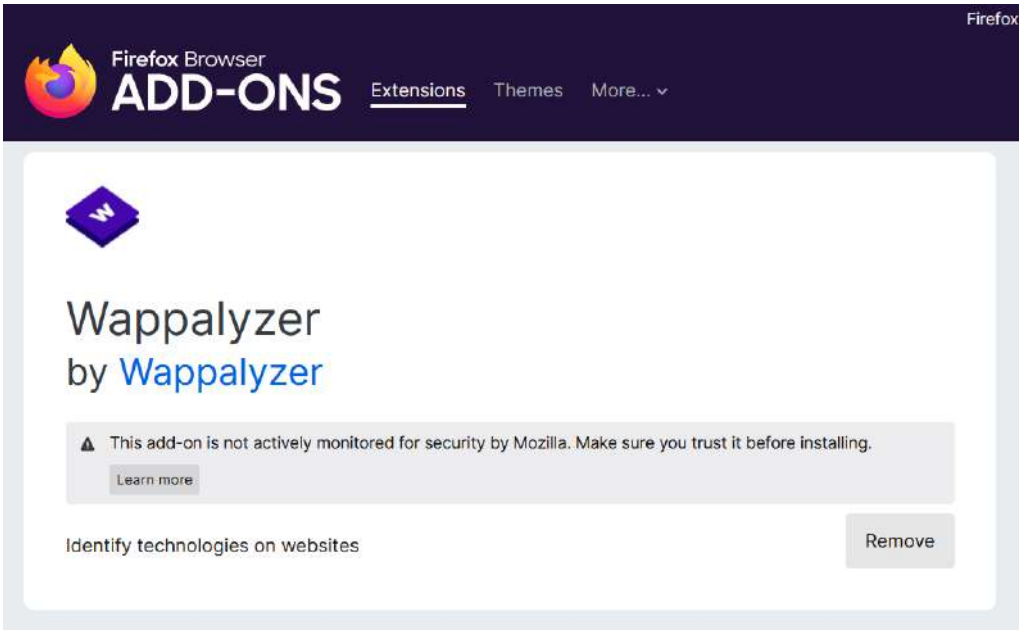
2.2.2 Enter the IP Address you want to query. Then click Go.

2.2.3 The list of all the sites hosted on the same domain will appear on screen.

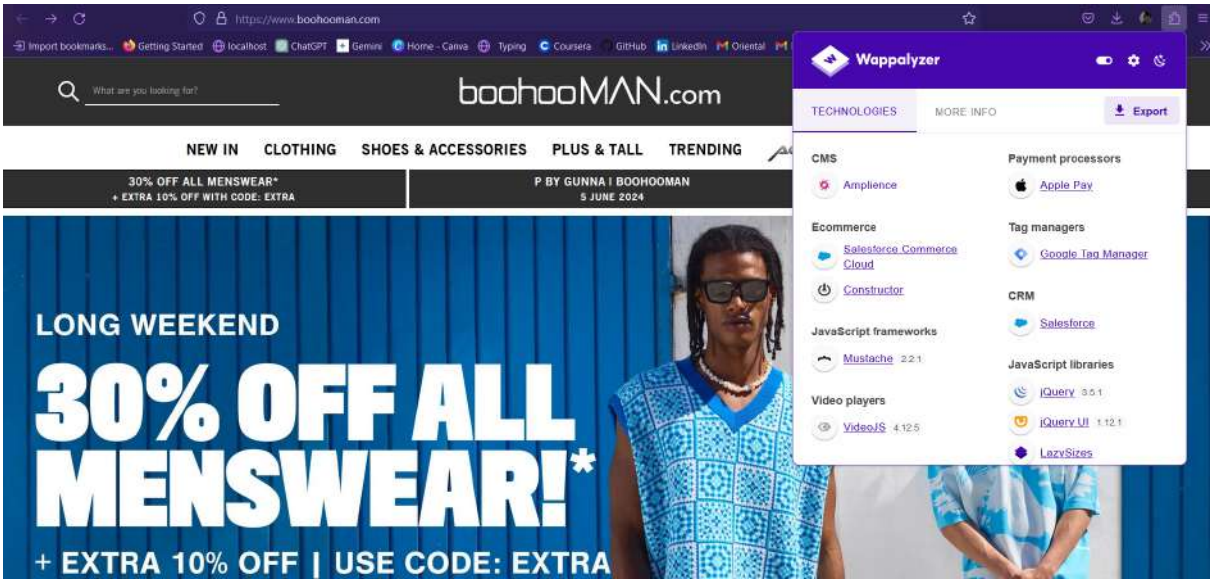
Domain	Last Resolved Date
0o7.biz	2012-10-20
1-tech.org	2024-05-27
100formations.org	2013-06-14
118018.com	2024-05-21
11daagsevlaanderen.net	2024-05-27
121docaffiliates.com	2013-03-02
12accede.com	2012-01-11
12enterchina.com	2012-11-12
1ebrand.com	2012-04-13
1stpolicy.com	2024-05-21
208.co.uk	2024-05-24
24-casinos.com	2024-05-21
24-poker.net	2024-05-27
24dreams.com	2024-05-21
24h-casinos.com	2024-05-21
24h-poker.org	2024-05-24
24roulette.net	2024-05-27
24x7availability.net	2011-12-19
3cx.mobi	2024-05-20
3d-job-center.net	2012-05-14
3d-shop-center.net	2012-05-14
3d-vit-coach.net	2012-05-07
3d-vit.net	2012-05-07
3dblu-rayplayers.com	2014-07-05
3dbluraymovie.com	2014-07-05

3. **Wappalyzer** (extension) - It is used to identify the Tech Stack used to build the website.

2.3.1 Install the Wappalyzer extension in your Browser (Available for Chrome and Firefox).



2.3.2 Now go to the particular website and open the Wappalyzer extension. The entire information regarding the Technology used in building and deploying the website will show up.



Conclusion

The Output Report of the Information Gathered from the above procedure is as follows-

1. **WEBSITE 1 - https://www.boohooman.co.uk**

```
Registrar Status
Dates   3,907 days old
Created on 2013-09-15
Expires on 2024-09-15
Updated on 2023-09-08
Whois History
IP Address  80.92.65.144 - 52,903 other sites hosted on this server


IP Location     Luxembourg - Luxembourg - Leudelange - Eurodns S.a.
ASN     Luxembourg AS24611 DCLUX-AS Datacenter Luxembourg S.A., LU (re
Hosting History     1 change on 2 unique name servers over 10 years
```

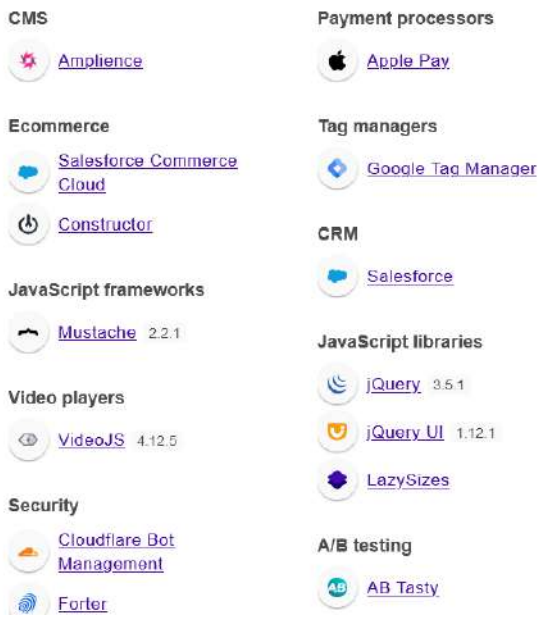
Reverse IP results for **80.92.65.144**

There are 52,896 domains hosted on this server.

Domain	Last Resolved Date
0o7.biz	2012-10-20
1-tech.org	2024-05-27
100formations.org	2013-06-14
118018.com	2024-05-21
11daagsevlaanderen.net	2024-05-27
121docaffiliates.com	2013-03-02
12accede.com	2012-01-11
12enterchina.com	2012-11-12
1ebrand.com	2012-04-13
1stpolicy.com	2024-05-21
208.co.uk	2024-05-24
24-casinos.com	2024-05-21
24-poker.net	2024-05-27
24dreams.com	2024-05-21
24h-casinos.com	2024-05-21
24h-poker.org	2024-05-24
24roulette.net	2024-05-27
24×7availability.net	2011-12-19
3cx.mobi	2024-05-20

Domain	Last Resolved Date
3d-job-center.net	2012-05-14
3d-shop-center.net	2012-05-14
...	...

Wappalyzer - Various Technologies used in this site is:



Tech Stack found using Wappalyzer Tool.

2. **WEBSITE 2 - https://www.SmythsToys.co.uk**

Registrar Status

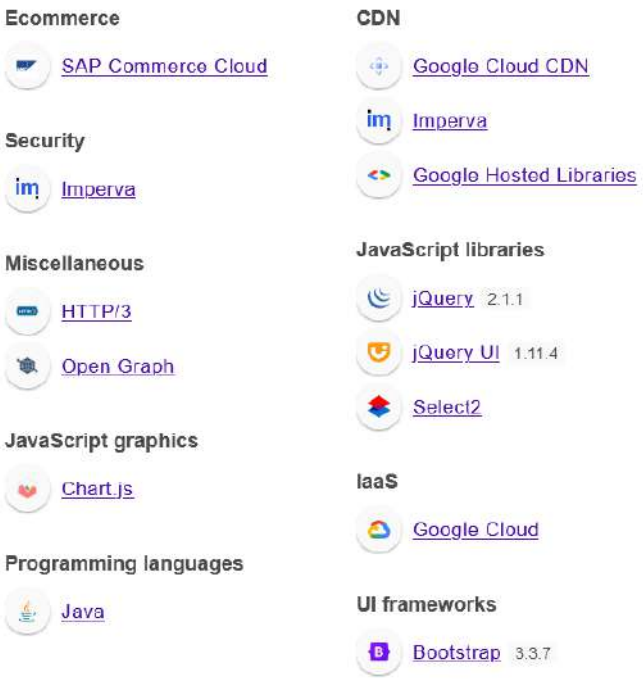
```
Registrar Status
Dates   7,440 days old
Created on 2004-01-13
Expires on 2026-01-13
Updated on 2023-09-25

IP Address  83.98.160.74 - 92 other sites hosted on this server


IP Location     Netherlands - Noord-holland - Amsterdam - Anu Internet
ASN     Netherlands AS8315 ACNBB Accenture B. V., NL (registered Jan 0
IP History  1 change on 1 unique IP addresses over 1 years

Hosting History     3 changes on 3 unique name servers over 10 years
```

Reverse IP results for **83.98.160.74**

There are 67 domains hosted on this server. The complete listing of these is below:

Domain	Last Resolved Date
picksnot.com	2024-05-21
schmidtstoys.de	2024-05-23
smithstoys.de	2024-05-23
smithstoys.fr	2024-05-25
smitstoys.de	2024-05-18
smyths.com	2024-04-27
smyths.ie	2024-05-26
smythsbabies.biz	2024-05-25
smythsbabies.co.uk	2024-05-24
smythsbabies.com	2024-05-21
smythssoftware.biz	2024-05-18
smythssoftware.co.uk	2024-05-24
smythssoftware.com	2024-05-21
...	...

Wappalyzer Various Technologies used in smythstoys.com is:



3. **WEBSITE -3 - https://www.goape.co.uk**

Domain Profile

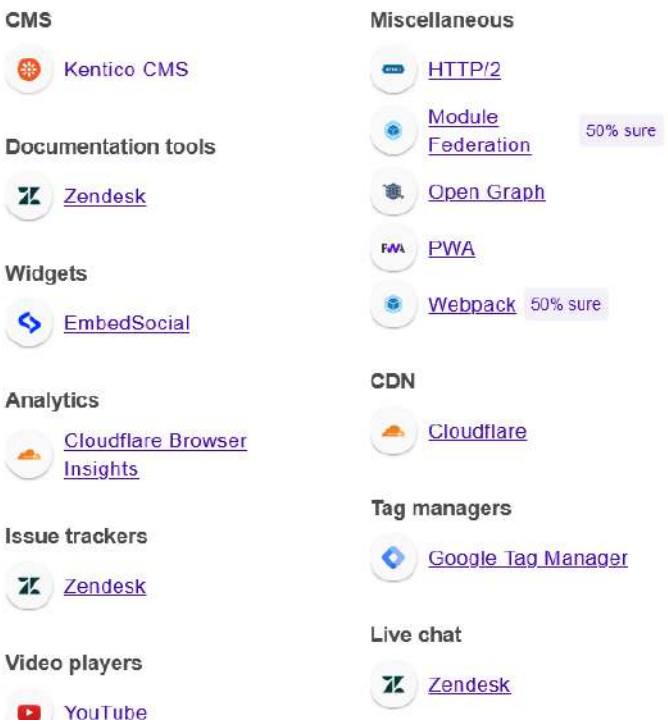
```
Registrar Status
Dates   8,858 days old
Created on 2000-02-25
Expires on 2025-02-25
Updated on 2024-02-21

IP Address  104.22.0.93 - 2 other sites hosted on this server


IP Location     United States - California - San Jose - Cloudflare Inc
ASN     United States AS13335 CLOUDFLARENET, US (registered Jul 14, 20
Hosting History     1 change on 2 unique name servers over 8 years
```

Reverse IP results for **104.22.0.93**

There are 3 domains hosted on this server. The complete listing of these is below:

Domain	Last Resolved Date
accentmagasin.se	2024-05-27
goape.co.uk	2024-05-26
thethao247.vn	2024-05-26

Wappalyzer Various Technologies used in ***goape.co.uk*** is:



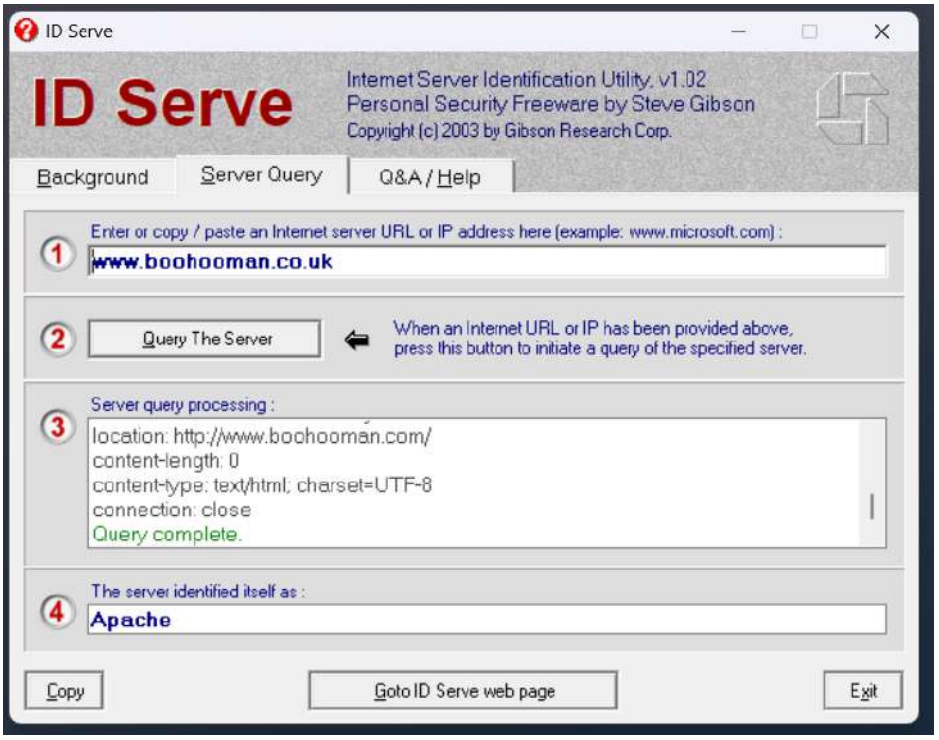
B) Identify the server Information and Wayback snapshots of the above sites using the below mentioned tools-

- **IDServe**
- **Wayback Machine**

STEP - 1 Identify server information using IDServe

1.1 Open IDServe.exe on your computer.

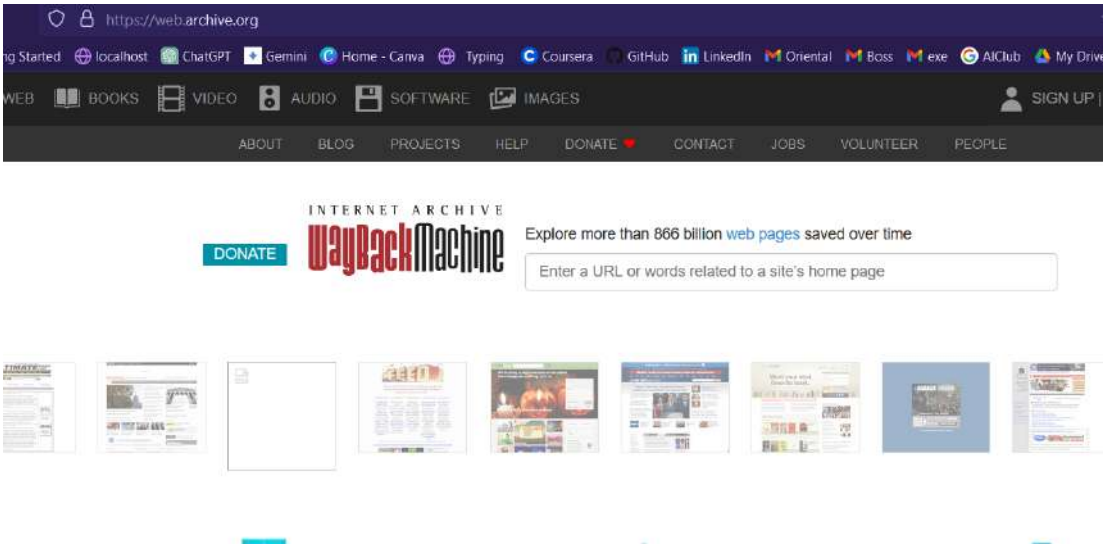
1.2 Paste the link and hit query.



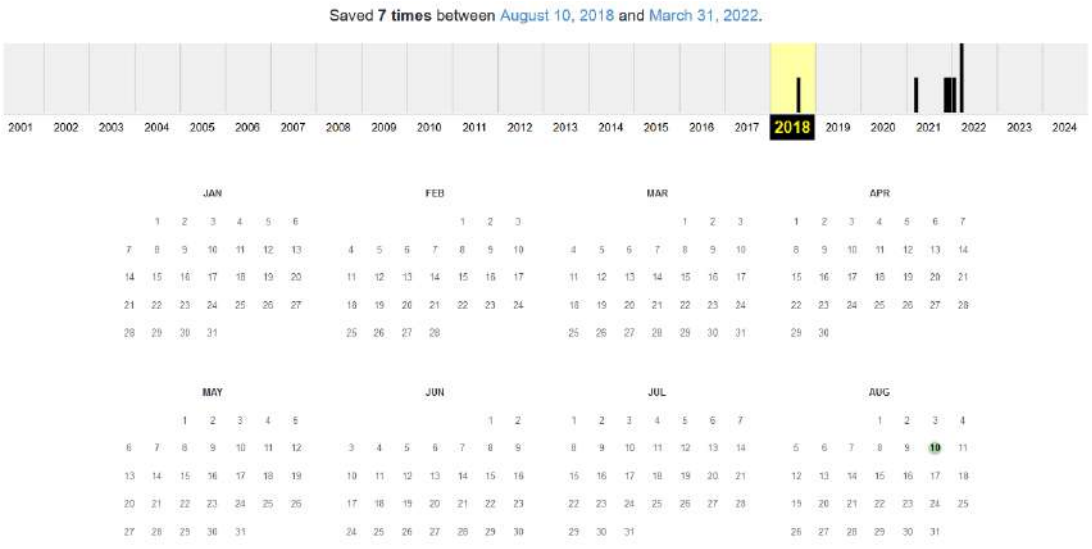
1.3 Upon clicking Query the Server you can see the Server is identified as **Apache** Server.

STEP - 2 Identify Snapshots using the Wayback Machine

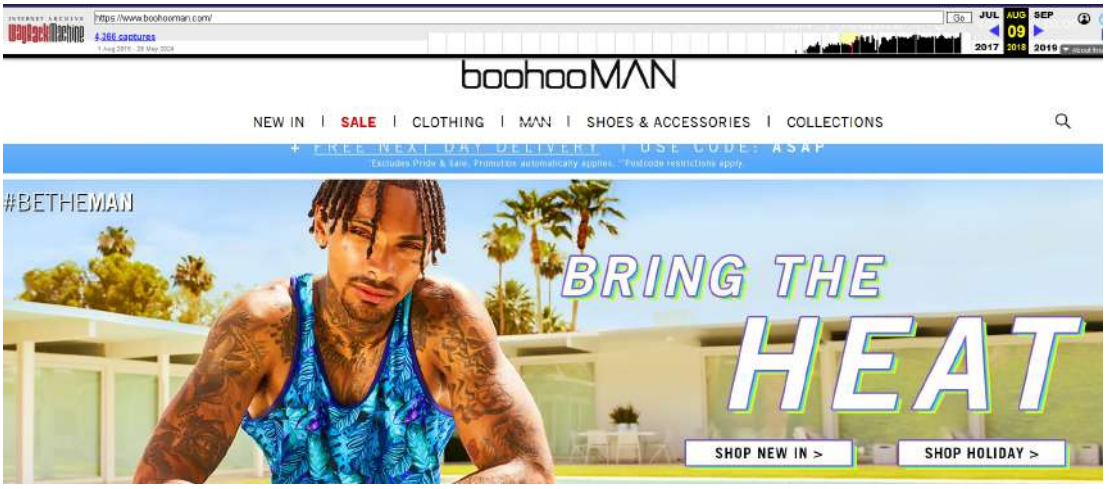
2.1 Go to https://web.archive.org/ on your browser.



2.2 Enter the url and click enter. (Here - **https://www.boohooman.co.uk). You can see various snapshots saved till date.**



2.3 Click to check any sanpshot.



Conclusion

The Output Report of the Information Gathered from the above procedure is as follows-

Server Information identified using **IDServe** Tool.

Website	Server
www.boohooman.co.uk	Apache
www.smythstoys.co.uk/	Apache
www.goape.co.uk	Cloudfare

Wayback URLs of **https://www.boohooman.com/**

August 10, 2018	https://web.archive.org/web/20180809080452/https://www.boohooman.com/
March 1, 2021	https://web.archive.org/web/20210301135811/https://www.boohooman.co.uk
November 28, 2021	https://web.archive.org/web/20211128024511/https://www.boohooman.co.uk
December 16, 2021	https://web.archive.org/web/20211216004943/https://www.boohooman.co.uk
March 7, 2022	https://web.archive.org/web/20220307102005/https://www.boohooman.co.uk
March 31, 2022	https://web.archive.org/web/20220331171259/https://www.boohooman.co.uk

Wayback URLs of **https://www.smythstoys.co.uk/**

March 21, 2004	https://web.archive.org/web/20061110204409/http://landing.domainsponsor.com/?a_id=36&domainname=smythstoys.co.uk
June 6, 2004	https://web.archive.org/web/20040606070528/https://www.smythstoys.co.uk
December 6, 2009	https://web.archive.org/web/20111125060715/http://www.smythstoys.co.uk/
November 25, 2011	https://web.archive.org/web/20111125060715/http://www.smythstoys.co.uk/
October 9, 2015	https://web.archive.org/web/20151009072015/https://www.smythstoys.co.uk
Feb 16, 2022	https://web.archive.org/web/20220216070450/https://www.smythstoys.co.uk

Wayback URLs of **https://www.goape.co.uk/**

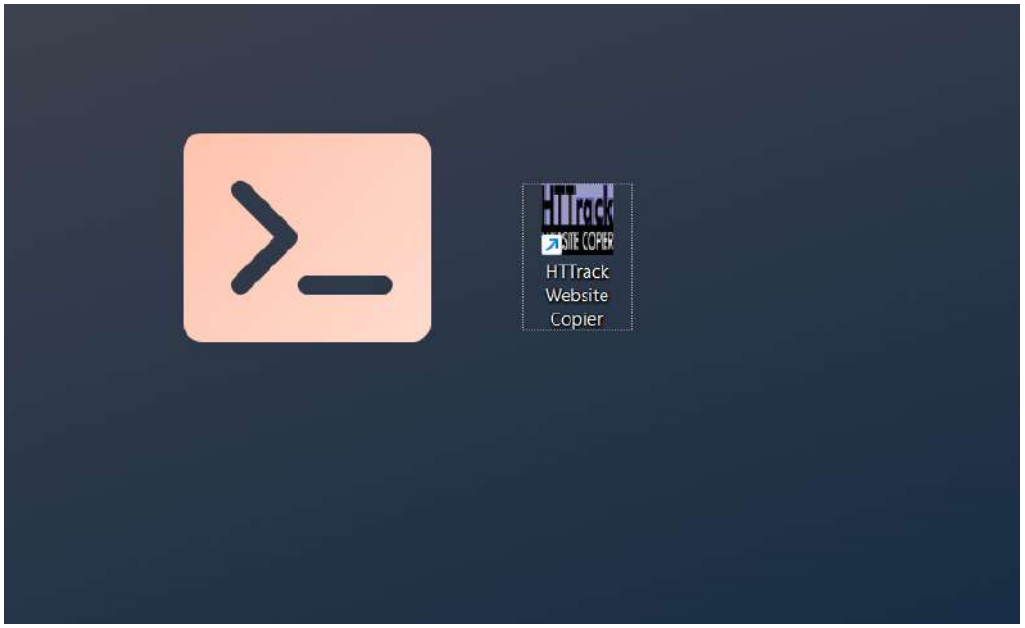
May 19, 2001	https://web.archive.org/web/20010519123145/https://www.goape.co.uk
December 1, 2003	https://web.archive.org/web/20031221015327/https://www.goape.co.uk
August 28, 2005	https://web.archive.org/web/20050828144207/http://www.goape.co.uk/
July 1, 2006	https://web.archive.org/web/20061205203335/https://www.goape.co.uk

C) Clone Two Pakistan Websites using HTTrack Tool-

We are going to clone these websites using HTTrack Website Copier.

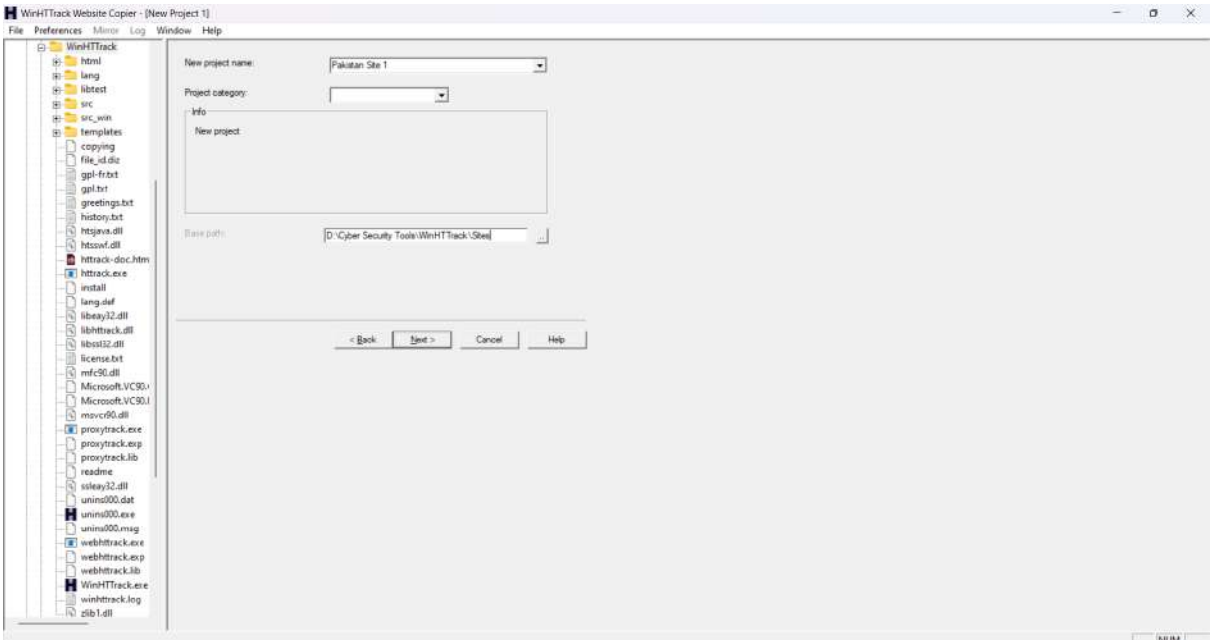
- **https://www.af.org.pk/**
- **https://www.ssuet.edu.pk/**

Step - 1 Install and Open HTTrack on your Computer

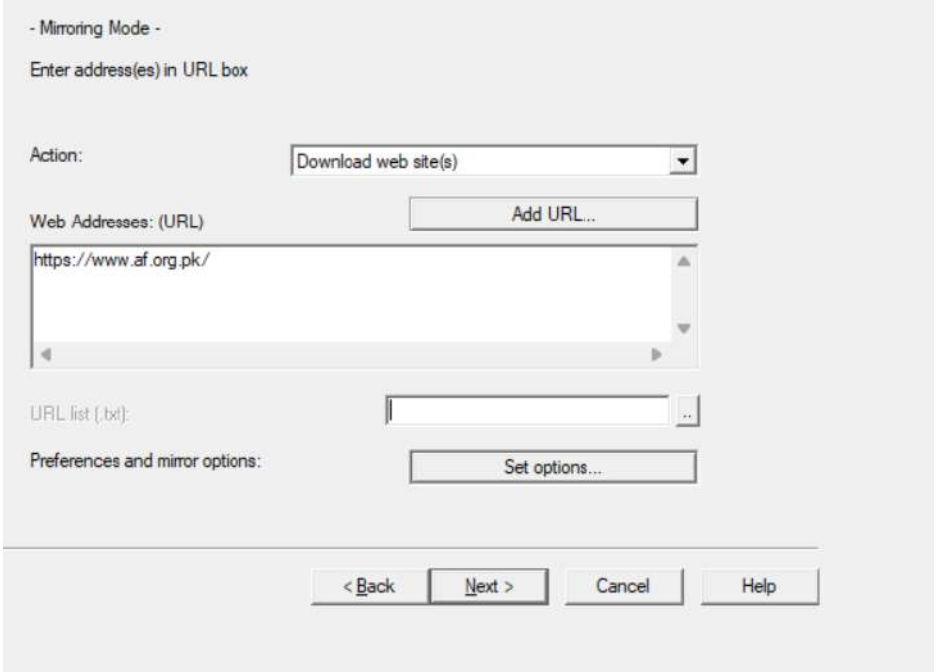


https://www.ssuet.edu.pk/

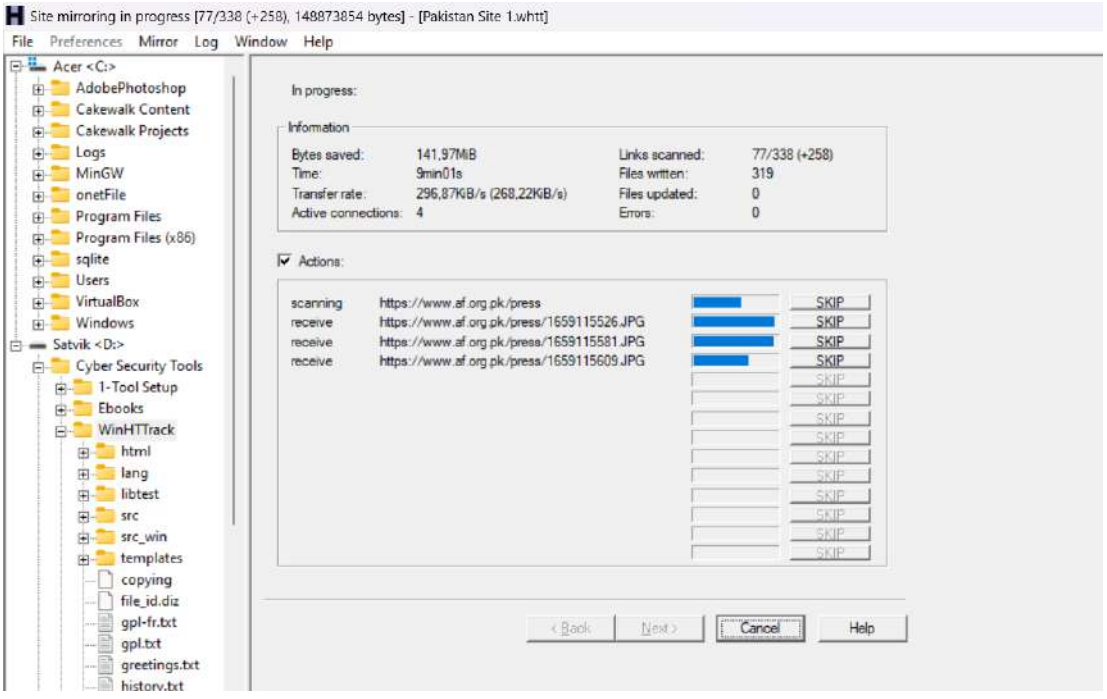
Step - 2 Give a suitable project name and click Next



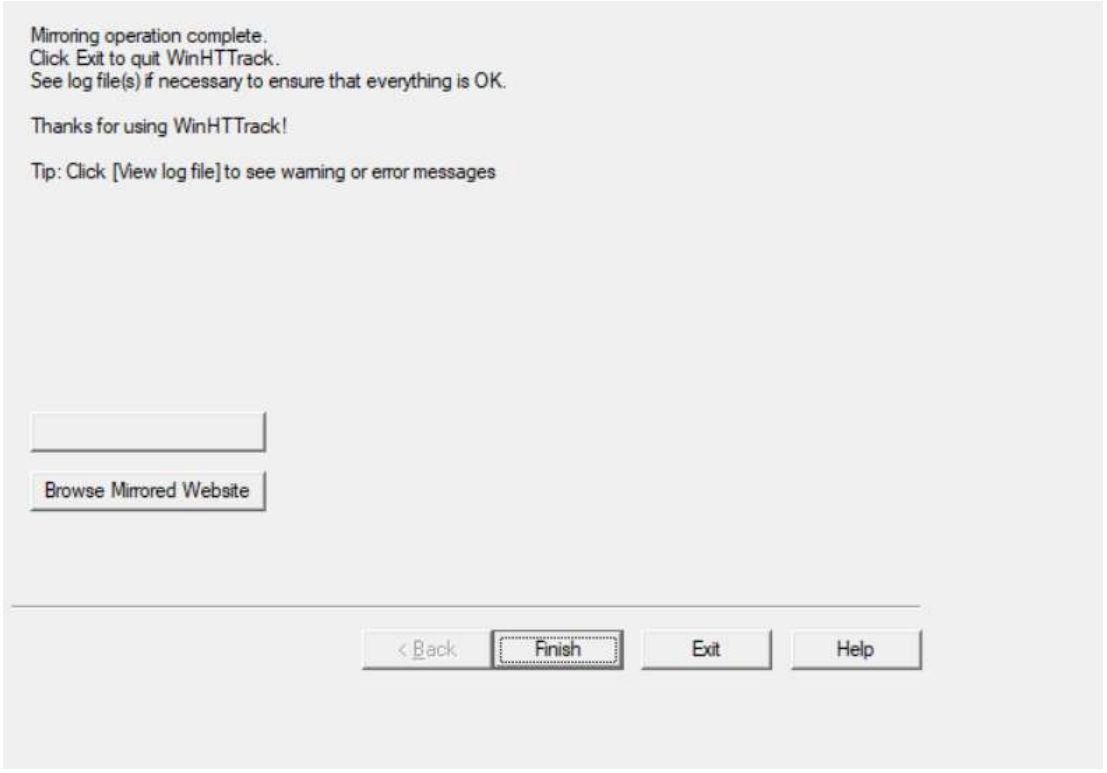
Step - 3 Enter the URL of the first website to be cloned. Then click Next.



Step - 4 After clicking Next, the website will start cloning into your machine, please be patient for the time being and let it fully clone.

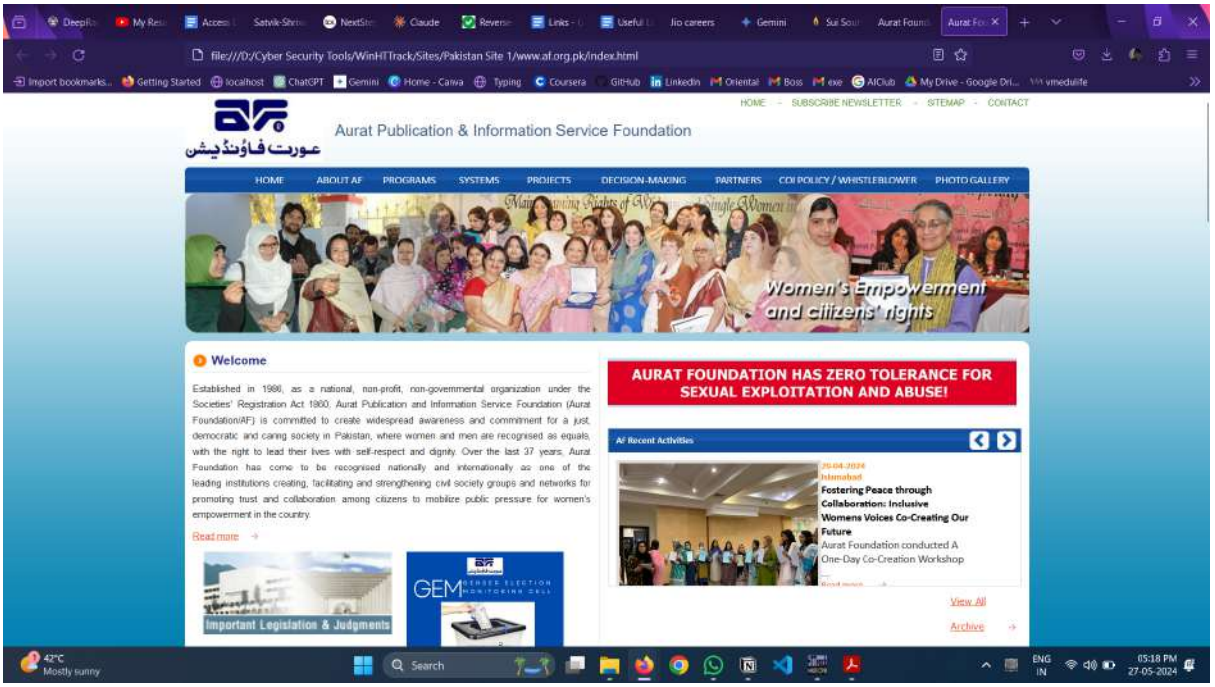


Step - 5 The website will get cloned when you will see this message. Now click on Browse Mirrored Website Button.



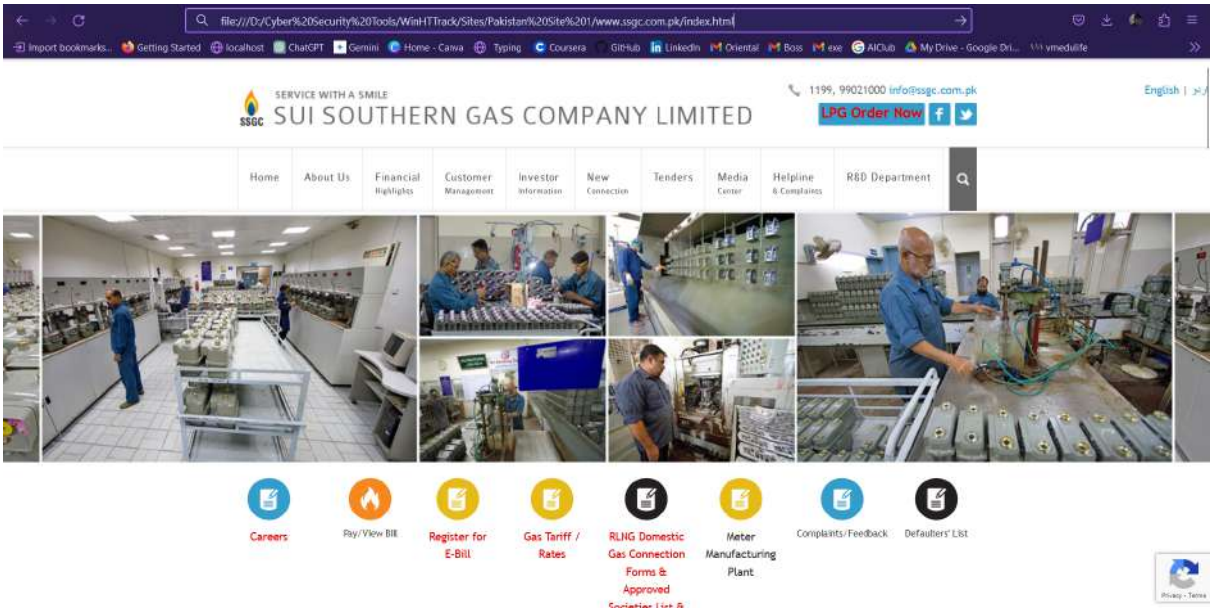
Step - 6 The Cloned website will appear on your Browser Window.

Cloned **https://www.af.org.pk/**



Similarly the other Website can also be cloned using the same procedure.

Here we have cloned **https://www.ssuet.edu.pk/**



23EO5-ST#IS#6653-ASSIGNMENT-2

Objective - Perform Scanning and Enumeration Tasks using *nmap* tool

nmap Tool-

nmap, short for **Network Mapper**, is a highly versatile, open-source tool that is widely used in cybersecurity for **network discovery** and **security auditing**. It sends packets to a specified host or network and analyses the responses to create a map of the network. *nmap* can be used to **discover hosts and services** on a network, providing information about the **operating system** and **hardware characteristics** of networked devices.

Tasks which can be performed with *nmap* include, but are not limited to: **scanning**, **enumeration**, network inventory, managing service upgrade schedules, monitoring host or service uptime, **finding open ports**, detecting security risks, and even some forms of **intrusion detection**.

Here is a list of commands that can be seen by **nmap -h :-**

A) Perform scanning on any 3 websites which have open ports.

- **Scan all 3 targets using the list command.**
- **Scan all 65,535 ports for 3 websites in a single command**

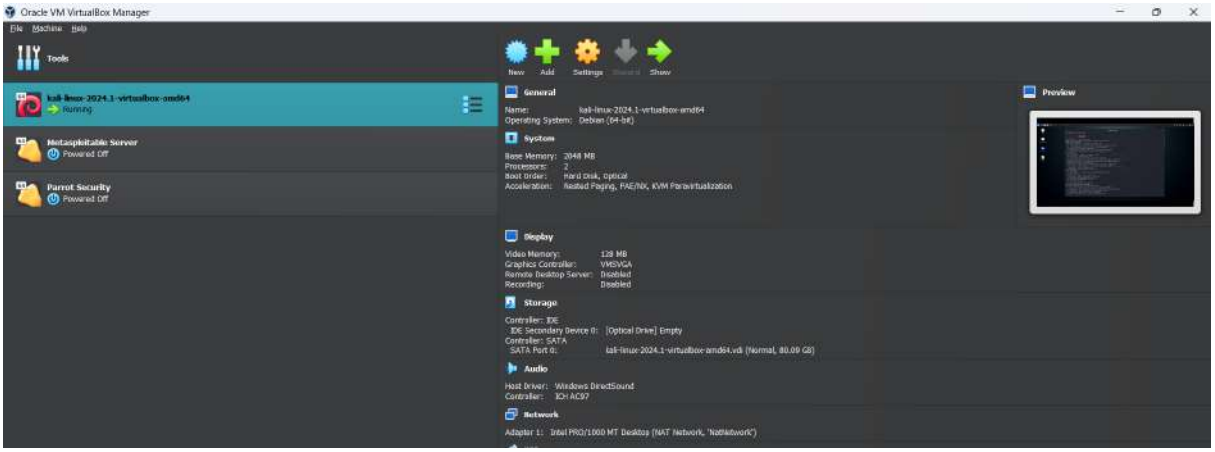
The Targets we are choosing here are-

- scanme.nmap.org
- testphp.vulnweb.com
- itsecgames.com

Scanning the Target websites using List command.

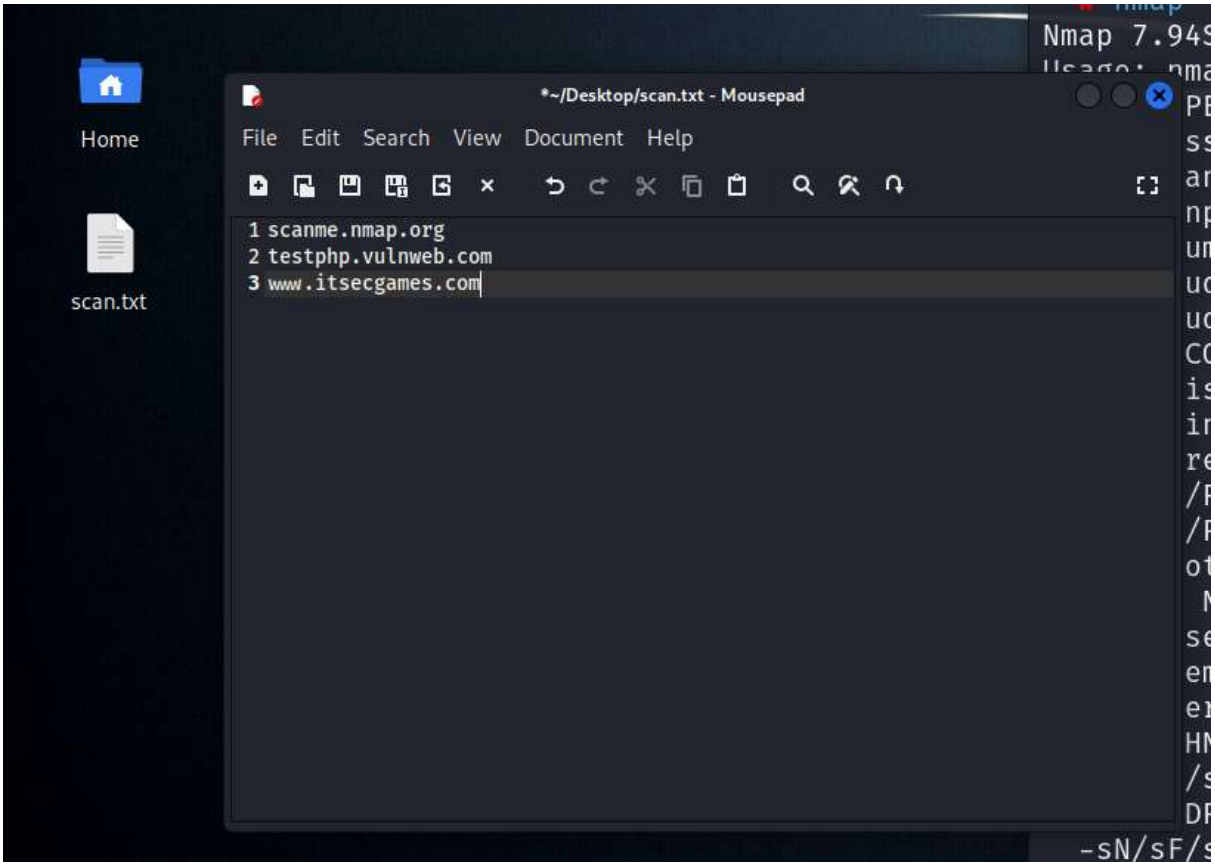
STEP 1

Start Kali Linux machine from your Virtual Box or VM Ware and check the network setting is in NAT Network.



STEP 2

Create a text file **scan.txt** and add the three targets, one per line:



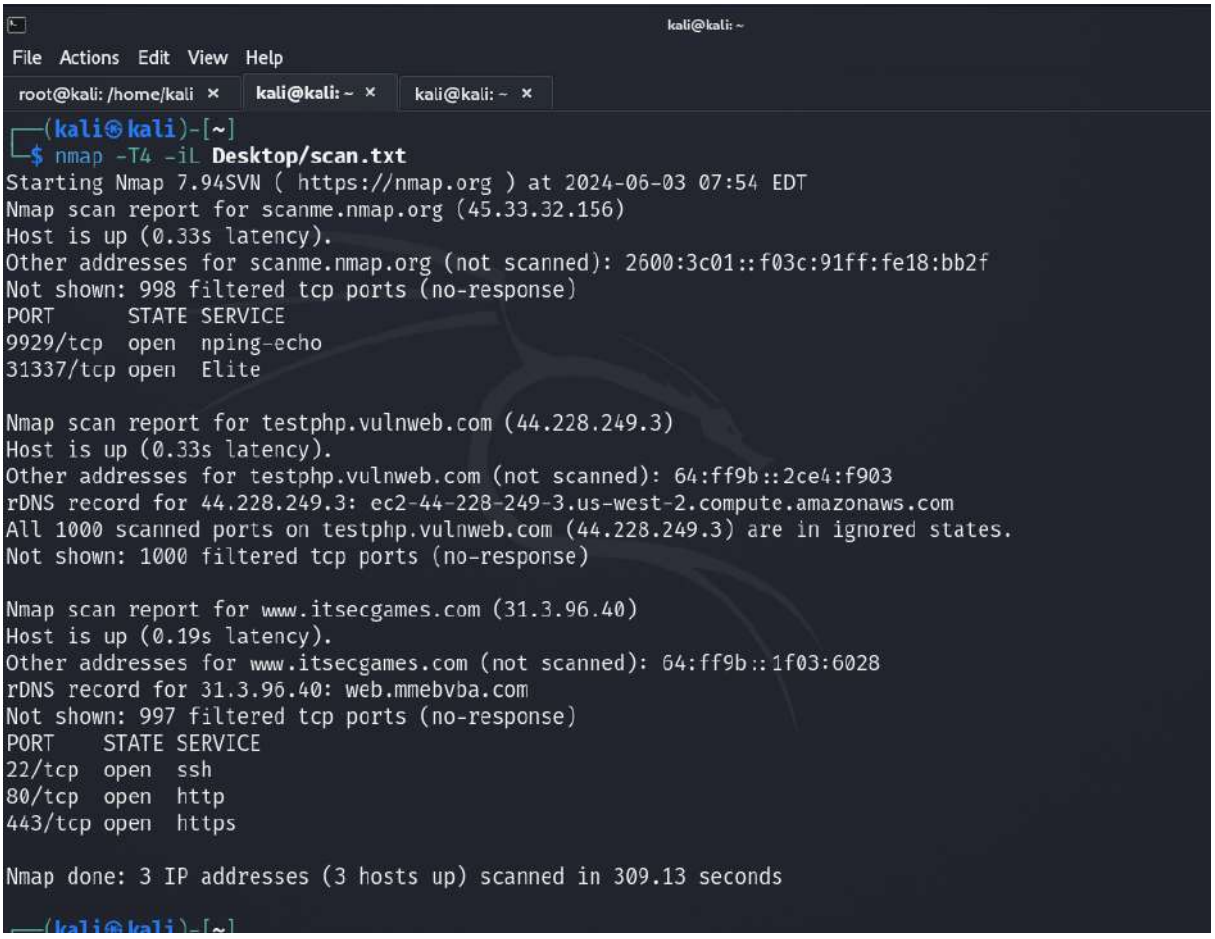
STEP - 3

Open the terminal and Use the following Nmap command to scan all targets listed in the scan.txt file

```
nmap -T4 -iL scan.txt
```

Nmap will scan the most common 1000 ports on each target specified in the scan.txt file and display the open ports, if any. T4 is used for faster outputs.

This is the Generated **Output**-

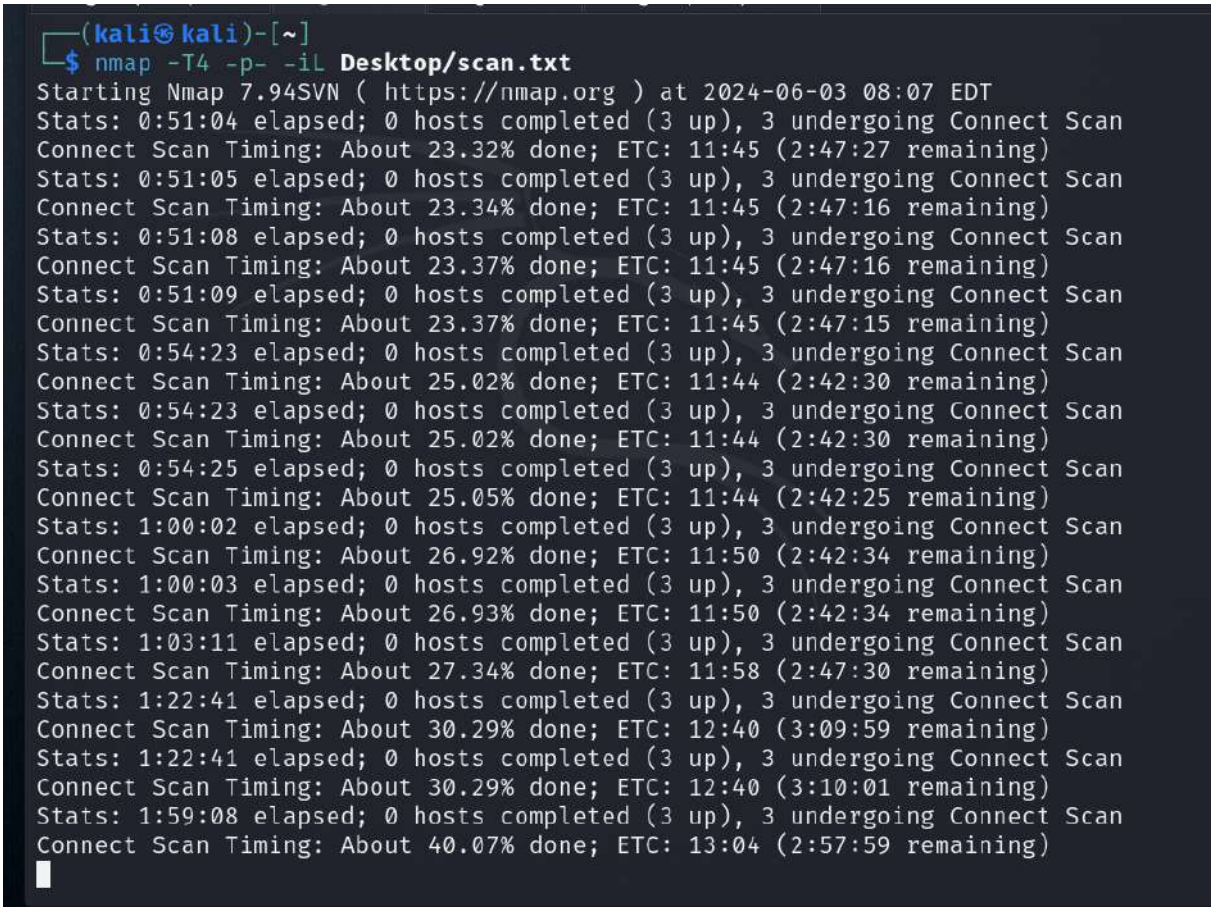


Scanning all 65,535 ports for target websites in a single command.

STEP 1

Use the following Nmap command to scan all 65,535 ports on the three targets:

```
nmap -T4 -p- -iL Desktop/scan.txt
```



B) Perform different types of scanning methods mentioned below on open ports.

- **Service Version Scan**
- **Aggressive Scan**
- **Retrieve Domain of an IP address- 54.251.249.26**

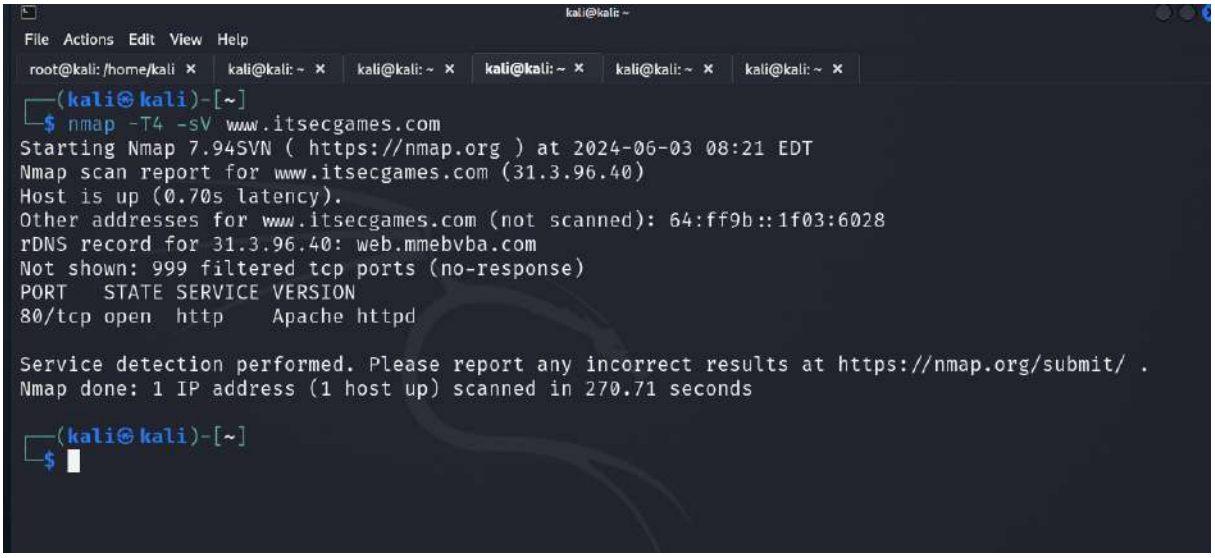
Performing Service Version Scan

Service Version Scan: A service version scan is used to identify the version of the services running on the open ports of a target system. This information can be helpful in identifying potential vulnerabilities associated with specific service versions.

```
nmap -sV [target_ip_or_domain]
```

STEP 1

Enter the above command on the kali linux terminal and hit Enter.



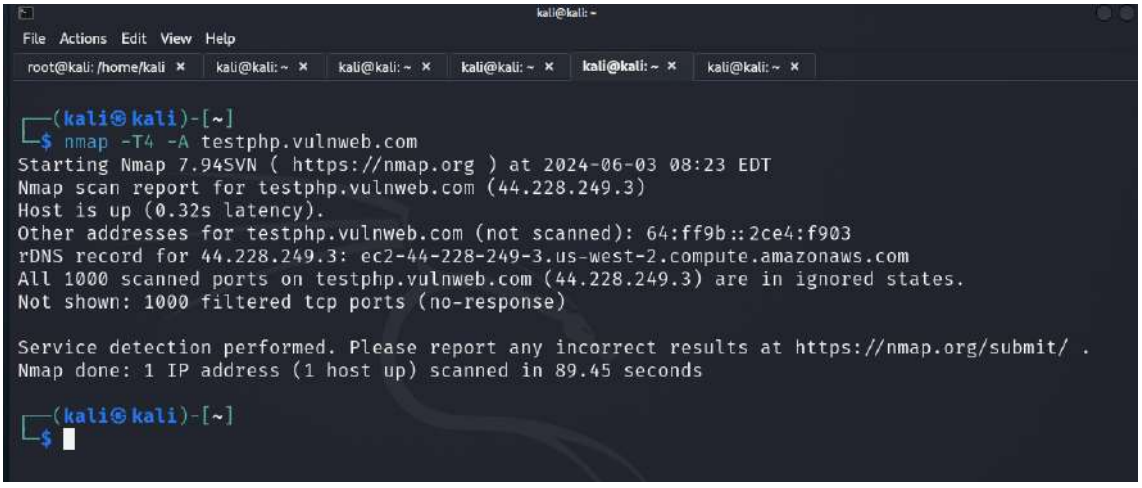
Performing Aggressive Scan

Aggressive Scan: An aggressive scan is a comprehensive scanning technique that combines various Nmap scan types and scripts to gather detailed information about a target system. It can be useful for identifying potential vulnerabilities and gathering reconnaissance data.

```
nmap -A [target_ip_or_domain]
```

STEP 1

Enter the above command on the kali linux terminal and hit Enter.



Retrieving Domain of an IP address- 54.251.249.26

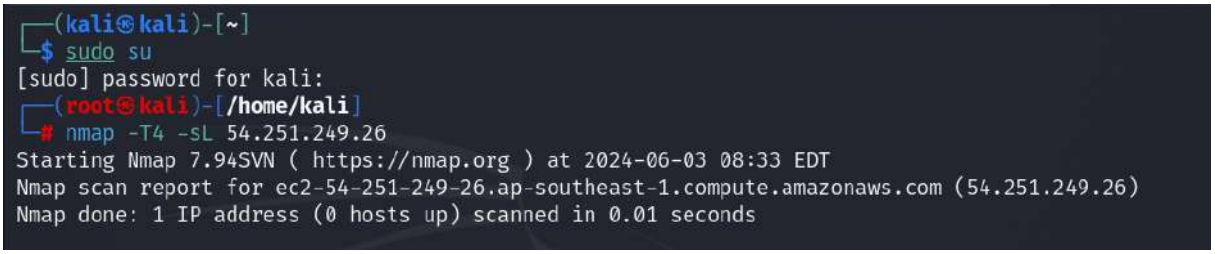
We can retrieve the hostnames/domains of any IP address by using any of the following commands.

```
nmap -T4 54.251.249.26
nmap -T4 -sL 54.251.249.26
```

STEP 1

Get root access in the terminal by entering sudo su in the Linux Terminal.

STEP 2

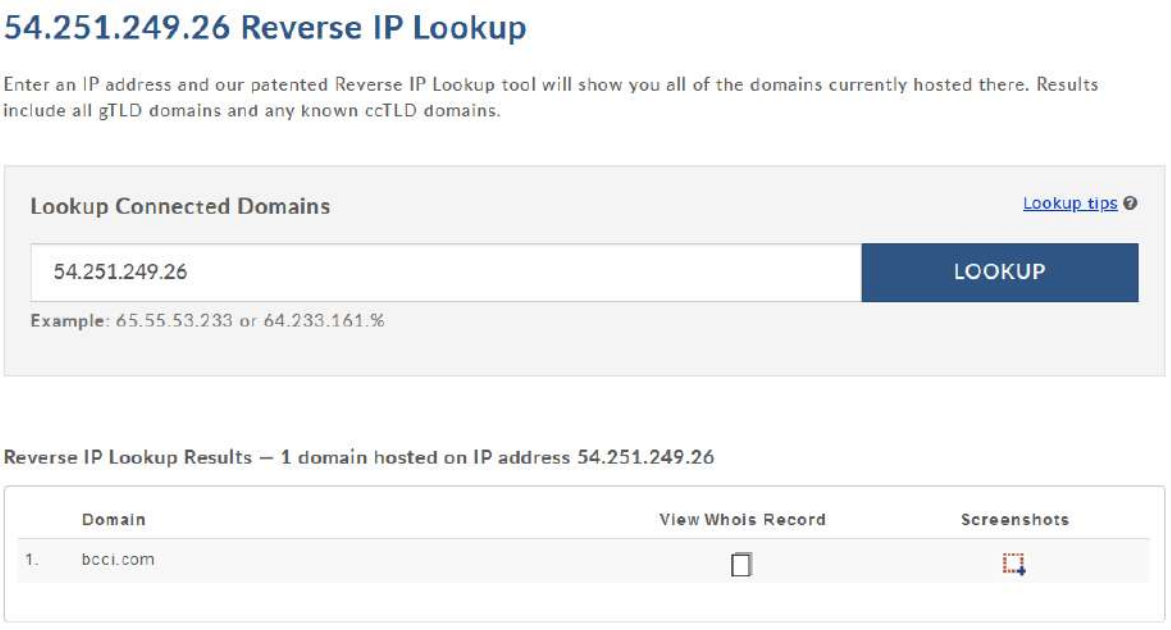


This is the generated output:

```
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-03 08:33 EDT
Nmap scan report for ec2-54-251-249-26.ap-southeast-1.compute.amazonaws.com
(54.251.249.26)
Nmap done: 1 IP address (0 hosts up) scanned in 0.01 seconds
```

STEP 3

For further information serach this ip on Reverse Ip Lookup website. On your favourite brower go to https://reverseip.domaintools.com and enter the ip **54.251.249.26** in the search box.



Observation:

Hence the host name for the ip **54.251.249.26** is **ec2-54-251-249-26.ap-southeast-1.compute.amazonaws.com** which indicates that this IP address belongs to an Amazon Web Services (AWS) EC2 instance in the Asia Pacific (Singapore) region.

The domain name for this website is **bcci.com**

C) Performing Different Enumerations on any Pakistan website

- **FTP Enumeration**
- **HTTP Enumeration**
- **SSL Enumeration**
- **SMTP Enumeration**
-

The target pakistani website we are choosing here is **https://homeshopping.pk/**

(ip - 104.21.34.56)

Enumeration

Enumeration is a critical phase of a cyber attack that involves **extracting more detailed information** about a target environment, such as **user names, machine names, network resources, shares, and services**. This process helps the attacker to get a better understanding of the system architecture and design, and to **identify potential vulnerabilities**.

There are several types of enumeration:

- **FTP Enumeration:** File Transfer Protocol (FTP) enumeration involves identifying FTP servers and attempting to connect to them to gather information about existing directories and files.

```
nmap --script ftp-anon,ftp-bounce,ftp-proftpd-backdoor,ftp-vsftpd-backdoor,
ftp-vuln-cve2010-4221 -p 21 [target]
```

- **HTTP Enumeration:** HyperText Transfer Protocol (HTTP) enumeration involves gathering information about web servers, including server type, cookies set, HTTP headers, and any potential security weaknesses.

```
nmap --script http-enum -p 80 [target]
```

- **SSL Enumeration:** Secure Sockets Layer (SSL) enumeration involves identifying SSL services and gathering information about the SSL certificate and configuration, which may reveal potential security weaknesses.

```
nmap --script ssl-enum-ciphers -p 443 [target]
```

- **SMTP Enumeration:** Simple Mail Transfer Protocol (SMTP) enumeration involves gathering information about the SMTP server. This could include the server banner, the type of mail server, and whether the server allows anonymous logins.

```
nmap --script smtp-commands,smtp-enum-users -p 25 [target]
```

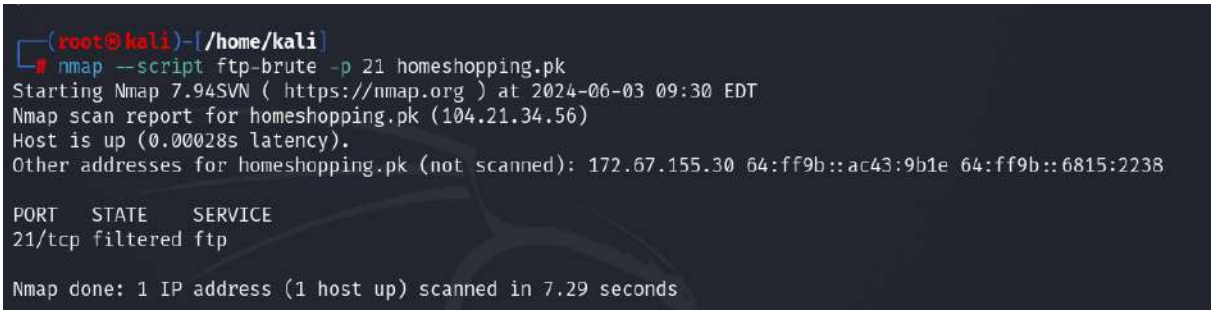
The target pakistani website we are choosing here is **https://homeshopping.pk/**

Performing FTP Enumeration-

STEP 1 Open kali linux terminal and search for open ports in the website by entering the following command.

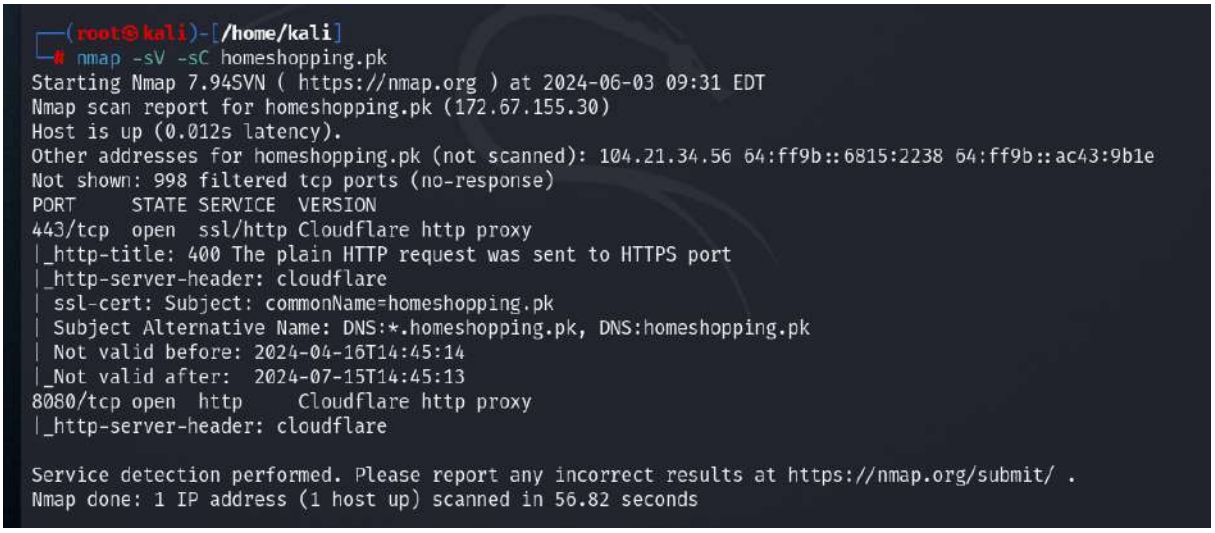
```
nmap --script ftp-brute -p 21 <host>
```

The command `nmap --script ftp-brute -p 21 <host>` performs an FTP brute force attack on the target host. It attempts to connect to the FTP server on port 21 using a list of usernames and passwords to find valid credentials.



To check if an FTP server allows anonymous logins we can enter the following command.

```
nmap -sV -sC <target>
```



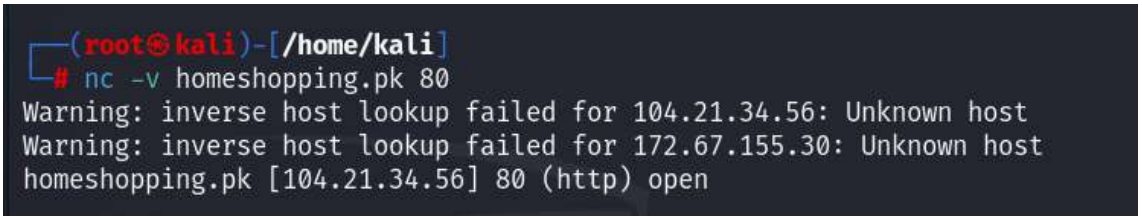
Performing HTTP Enumeration

HTTP enumeration is the process of gathering information about a web server and its associated web applications.

1. **Basic HTTP Banner Grab**

```
nc -v <target IP> 80
```

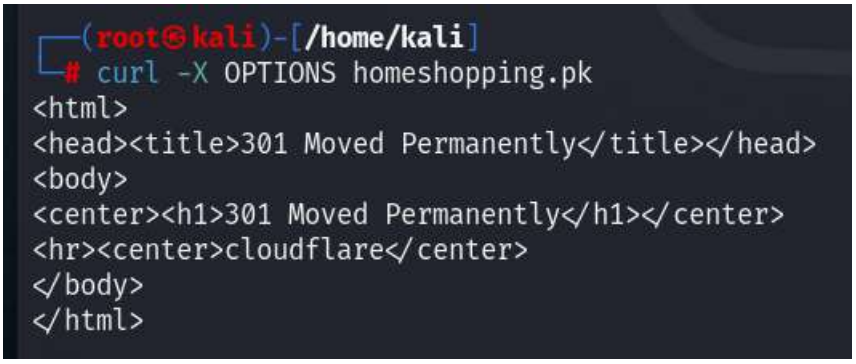
This command connects to port 80 of the target IP address using the nc (netcat) command, and retrieves the HTTP banner for the web server.



2. **HTTP options:**

```
curl-XOPTIONS<targetURL>
```

This command sends an HTTP OPTIONS request to the target URL using the curl command, and retrieves information about the server's supported HTTP methods and other capabilities.



3. **HTTP headers:**

```
curl-I<targetURL>
```

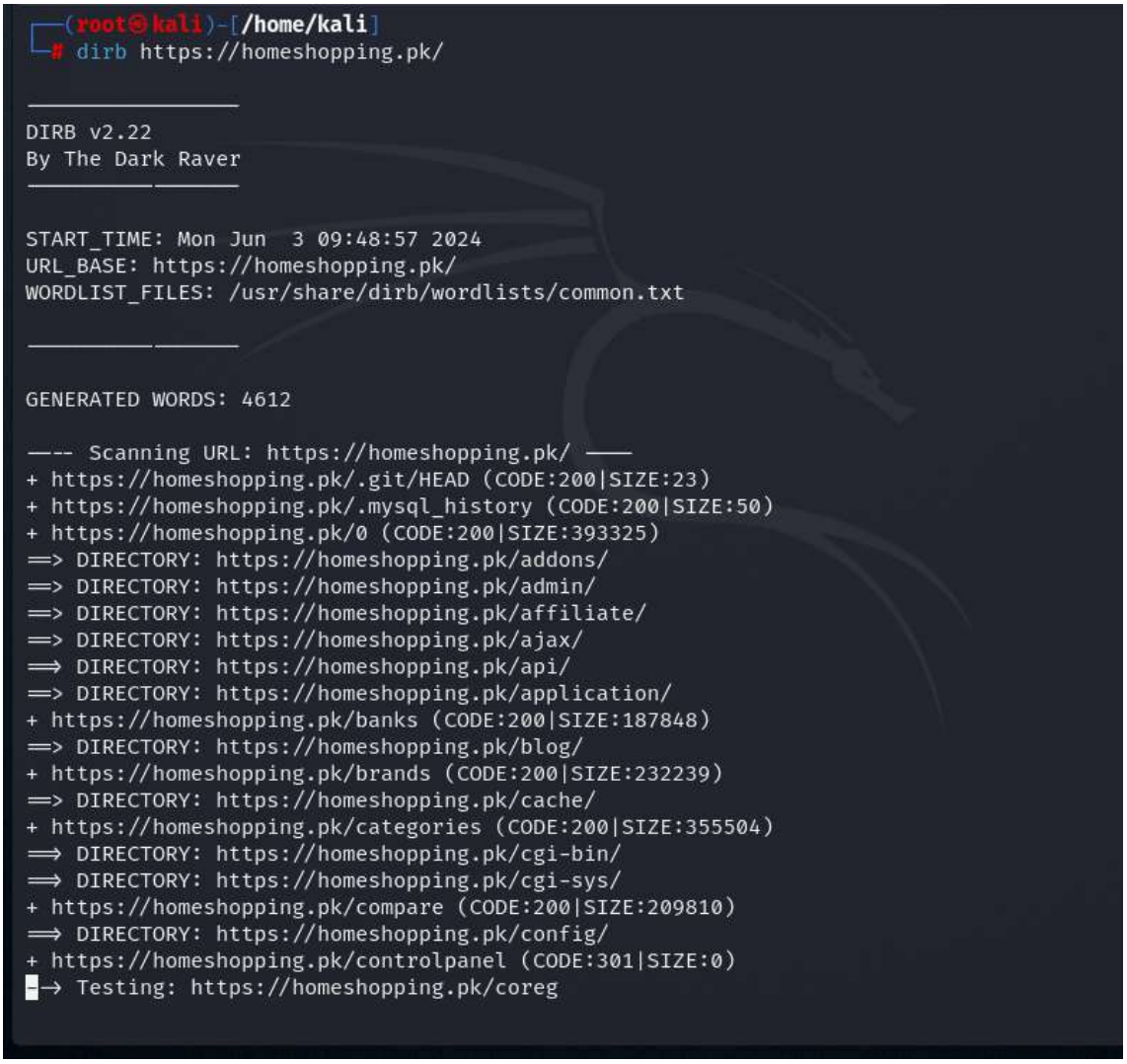
This command sends an HTTP HEAD request to the target URL using the curl command, and retrieves the server's response headers. This can include information such as the server type, supported content types, and other details.

4. **Directory enumeration:**

```
dirb <targetURL>
```

This command performs a directory brute-force attack against the target URL using the dirb tool, and attempts to enumerate all available directories and files on the web server.

Here is the list of directories scanned by directory enumeration on the target site.



5. **Content discovery:**

```
gobuster -u <target URL> -w <wordlist file> -x <fileextensions>
```

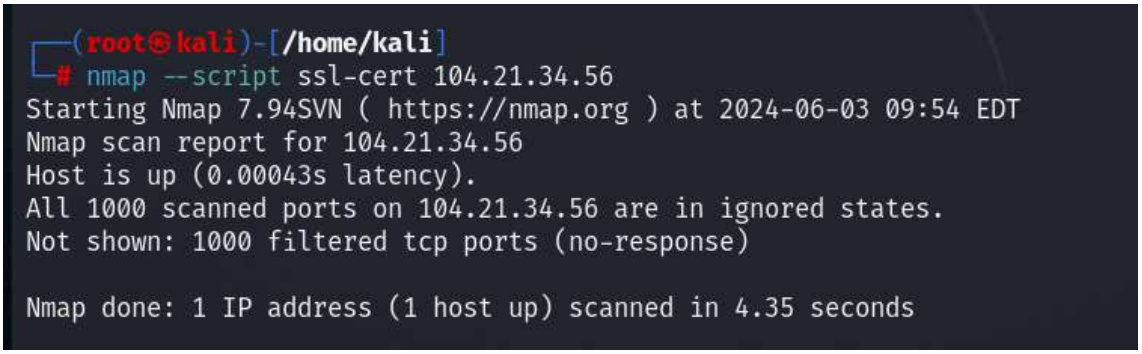
This command performs a content discovery attack against the target URL using the gobuster tool, and attempts to discover hidden files and directories on the web server. The -u option specifies the target URL, the -w option specifies the wordlist file to use for the attack, and the -x option specifies the file extensions to search for (e.g. php, html, txt, etc.).

Performing SSL Enumeration

```
nmap –script ssl-cert ip_address
```

The command `nmap –script ssl-cert ip_address` is used to obtain the SSL certificate of a target system. The `ssl-cert` script is part of Nmap's scripting engine, which allows Nmap to perform a variety of additional functions beyond basic scanning. In this case, the `ssl-cert` script retrieves the SSL certificate from the target system,

providing information such as the issuing authority, the validity period, and any subject alternative names (SANs) present in the certificate.



Performing SMTP Enumeration

```
sudo nmap --script smtp-ntlm-info <target>
```

The command `sudo nmap --script smtp-ntlm-info <target>` is used to gather information about the Simple Mail Transfer Protocol (SMTP) server of the target system, specifically the server's NTLM information. NTLM (New Technology LAN Manager) is a suite of Microsoft security protocols that provides authentication, integrity, and confidentiality to users.

END OF ASSIGNMENT

23EO5-ST#IS#6653-Task-3

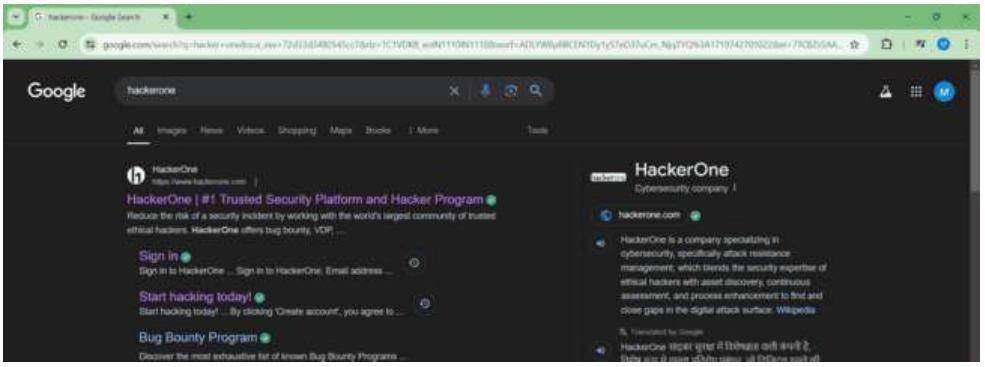
Objectives - Implement Subdomain Enumeration, Social Engineering and Email Spoofing.

A) Find Subdomains for Any 3 targets that you need to select targets from the HackerOne platform.

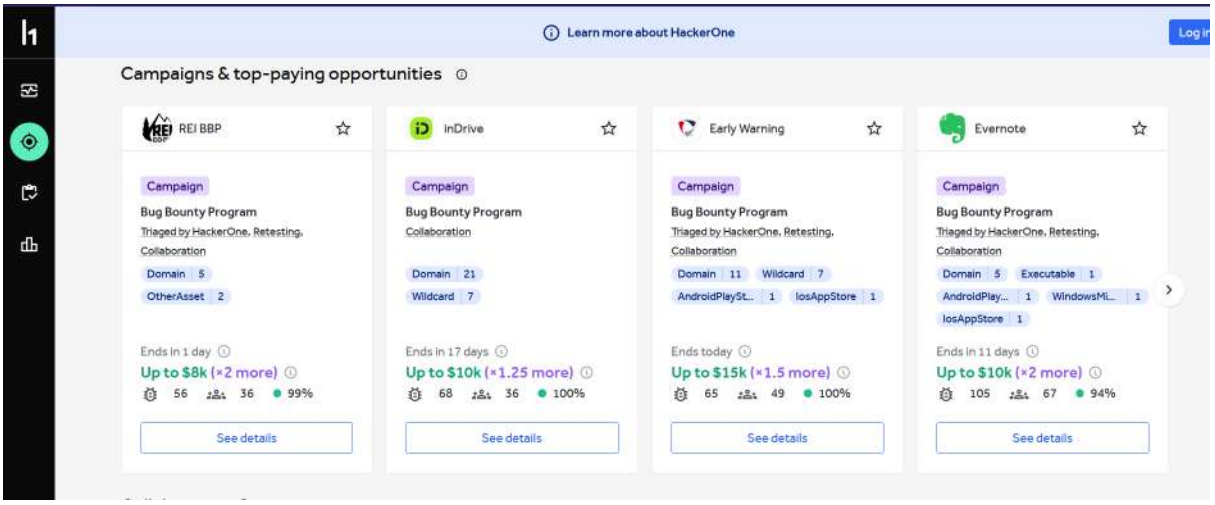
Subdomains are subdivisions of your website's main domain. They're used to organize and navigate to different sections of your site. Each subdomain can contain its own distinct content, such as blog.yourwebsite.com or shop.yourwebsite.com.

Amass is a tool that is used for in-depth attack surface mapping and asset discovery. It uses open source intelligence (OSINT) to help discover network infrastructure, including **subdomains**, domains, IP addresses, and more. This information is crucial for hackers and cybersecurity professionals alike to understand the layout of a network before initiating an attack or a security audit.

1. Open any Browser and Search HackerOne.



2. Open the first link and search for targets



Here we are trying to find the subdomains of inDrive, Evernote and Grammarly.

3. Now Open Oracle VM Manager

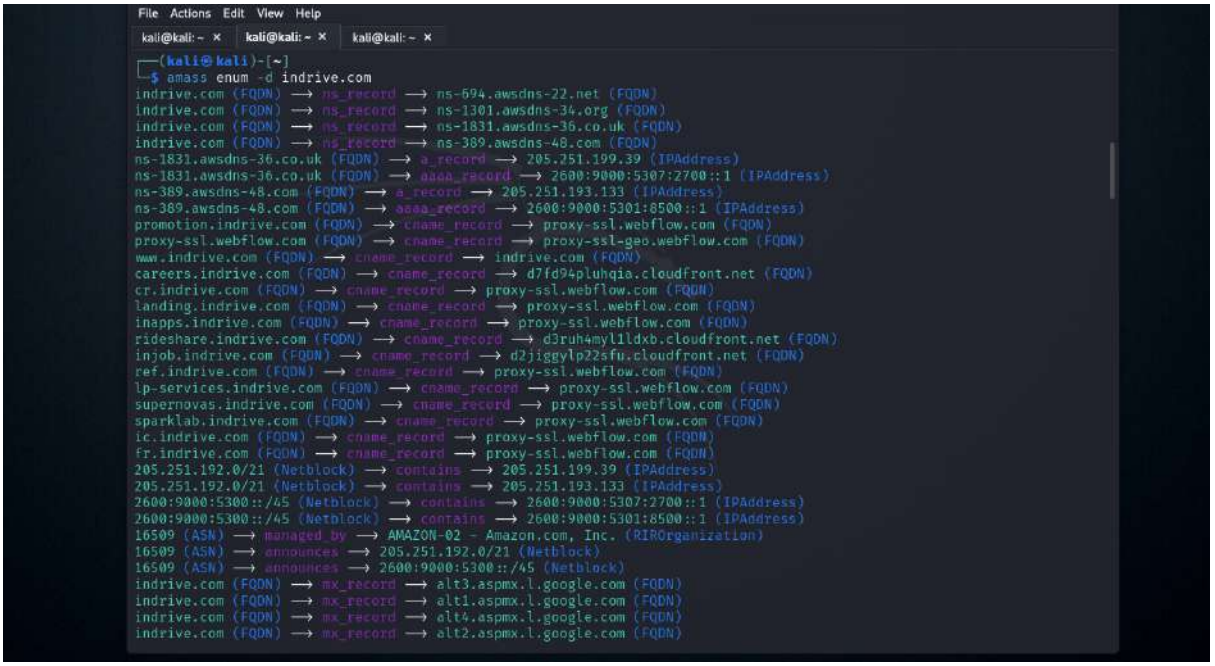
4. Start Kali Linux.

5. Open Terminal in Kali

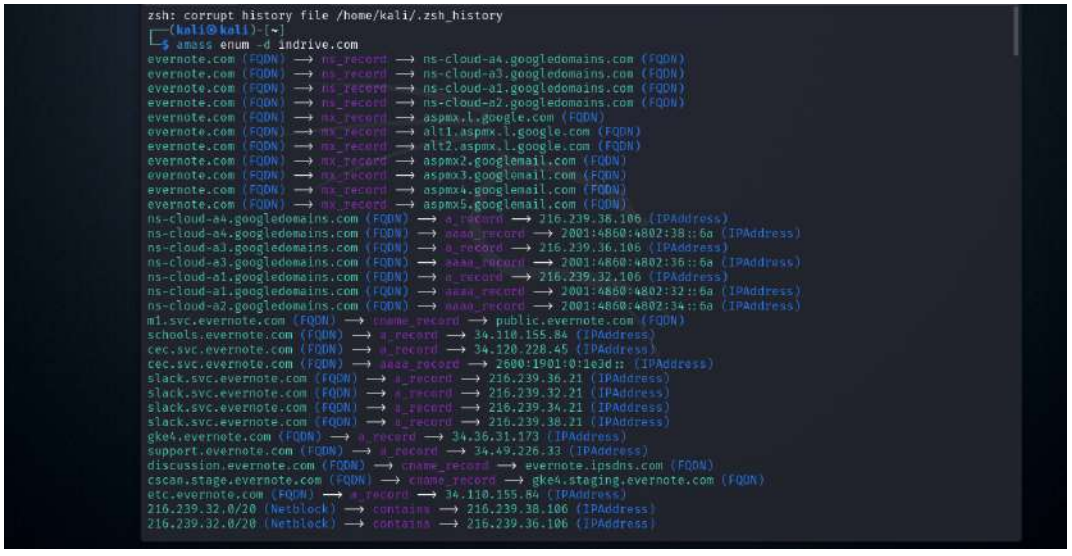
6. Install amass package by `sudo apt-get install amass.`

7. Amass package will be installed instantly.

8. Now type `amass enum -d indrive.com` to scan the first site. Here are the results.



9. Similiarly search for Evernote.com by the command `amass enum -d Evernote.com`



10. Search the subdomains for the finaly site. `amass enum -d grammarly.com`

B. Create a fake login page for 3 social media handles given below and detect if the link is malicious or not, using tools like Virustotal or Netcraft extension.

¬ LinkedIn

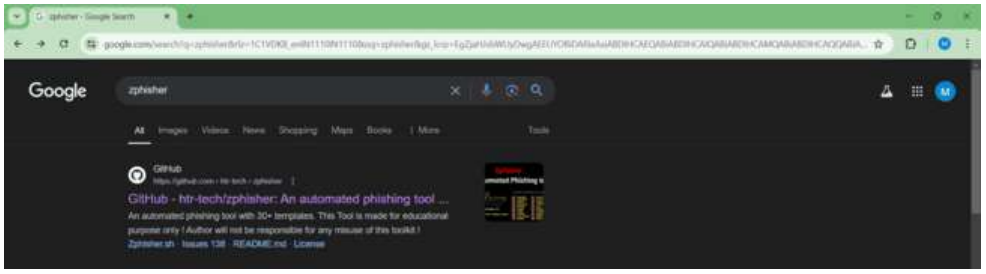
¬ Microsoft 365

¬ Gmail

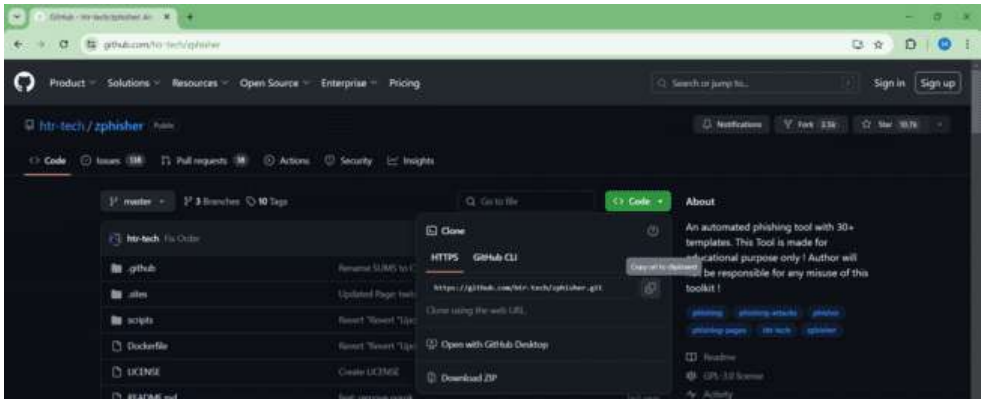
Zphisher is an advanced phishing toolkit that is designed for penetration testing. It provides the capabilities to quickly and easily create phishing pages for various sites, including social media platforms like LinkedIn, Microsoft 365, and Gmail. This tool is primarily used in the cybersecurity field for educational purposes, allowing cybersecurity professionals to understand phishing techniques and improve their defenses against such attacks.

STEP 1. Open Google Chrome

STEP 2. Search zphisher



STEP 3. Open the first link and Copy the URL



STEP 4. Open oracle Virtual manager

STEP 5. Start Kali Linux

STEP 6. Open terminal in Kali

STEP 7. Write “ git clone https://github.com/htr-tech/zphisher.git”

STEP 8. Write “ls” command

STEP 9. Write cd zphisher

STEP10. Write “ls -l”

STEP 11. Write “./zphisher.sh”

For LinkedIn

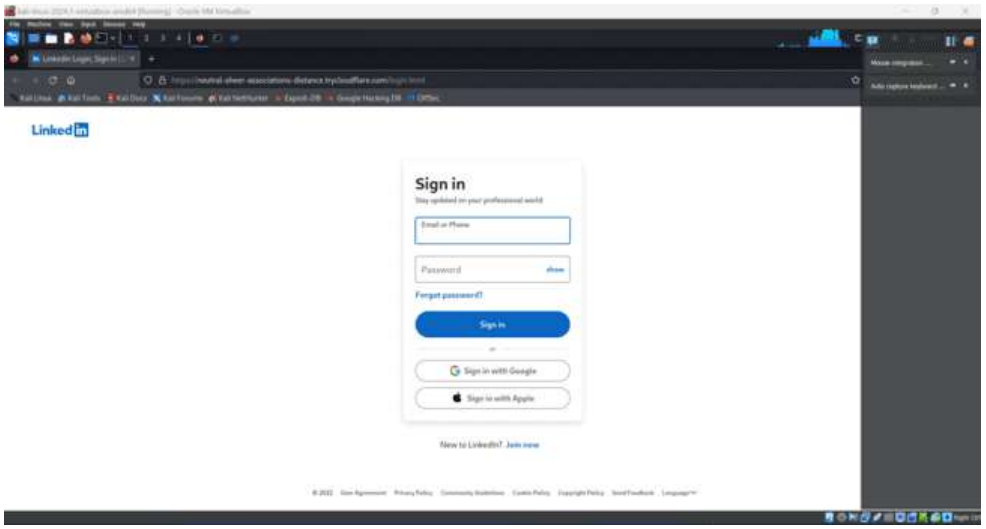
STEP 12. Select option 14 for Linkedin

STEP 13. Select option 2 Cloudfared

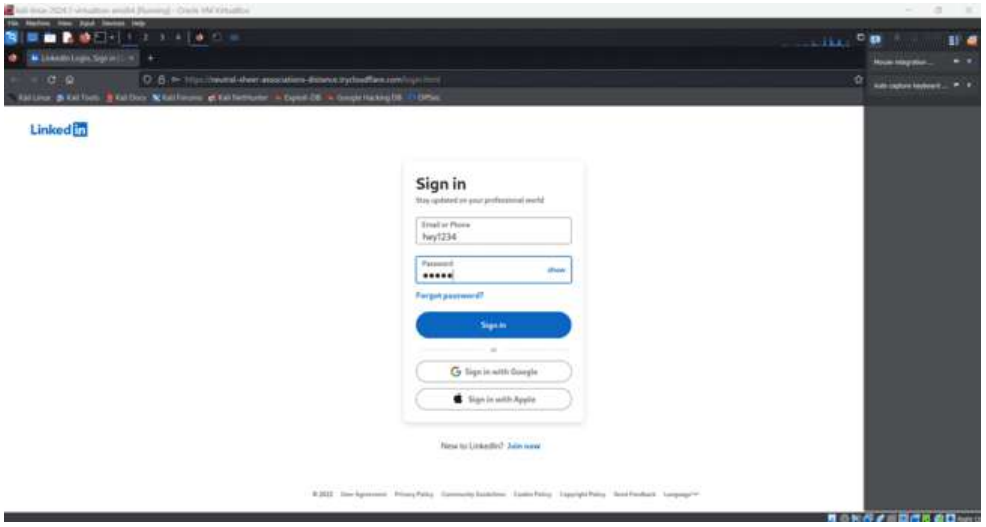
STEP 14. Click N

STEP 15. We got 3 urls

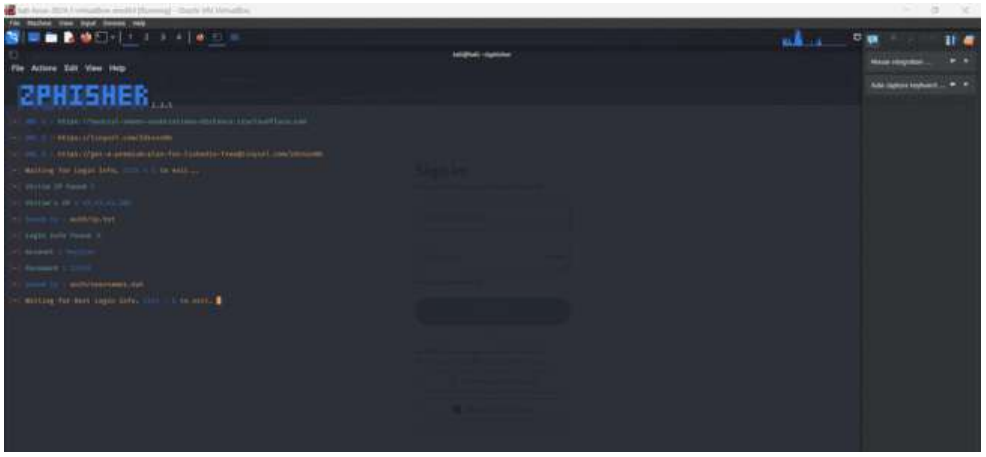
STEP 16. Right Click on the first url and click on open link



STEP 17. Enter your username and password

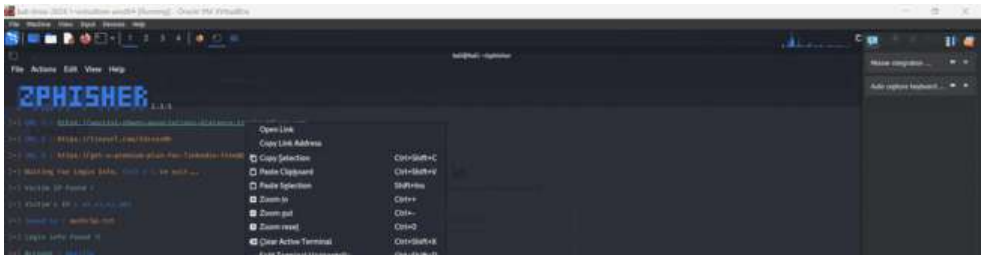


STEP 18. Open kali linux and see details of the user



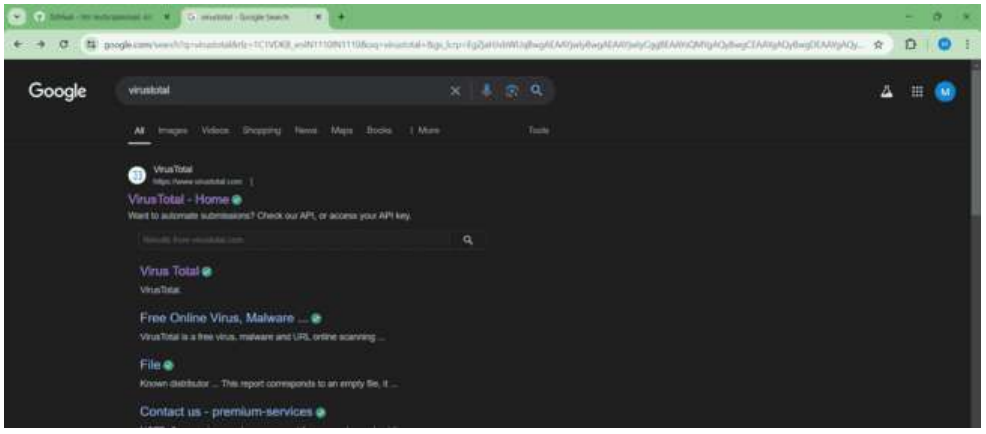
Details of Victim user found

STEP 19. Copy the cloudfared url



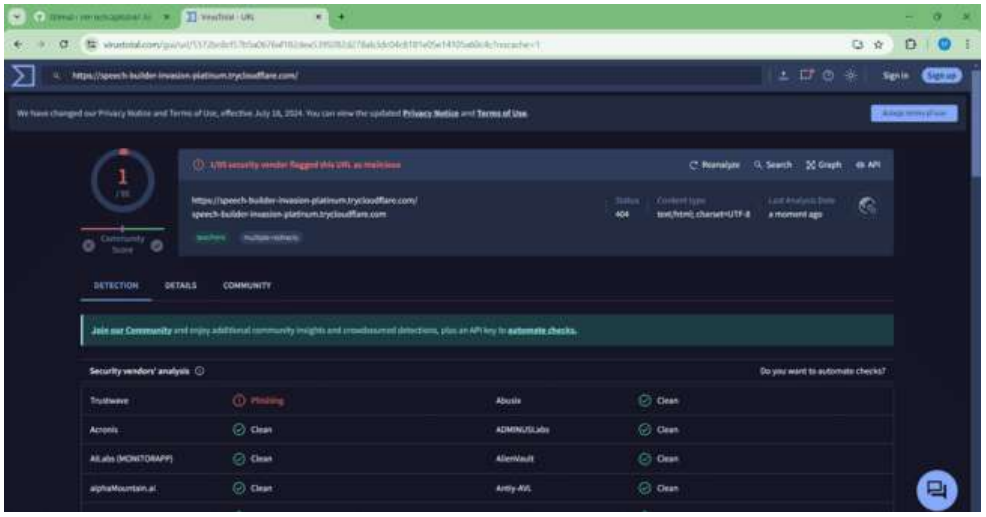
STEP 20. Open chrome browser

STEP 21. Search “Virustotal ”



STEP 22. Open virustotal

STEP 23. Enter the link in search bar and press enter



Here is the result

For Microsoft 365

STEP 24. Open terminal in Kali

STEP 25. Write " git clone https://github.com/htr-tech/zphisher.git"

STEP 26. Write “ls” command

STEP 27. Write cd zphisher

STEP 28. Write “ls -l”

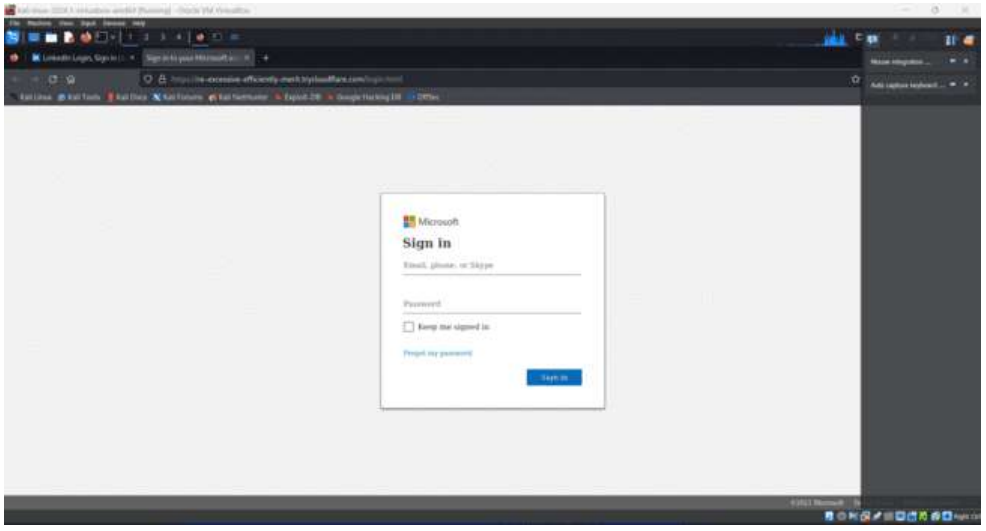
STEP 29. Write “./zphisher.sh”

STEP 30. Select option 4 for Microsoft 365 and press enter

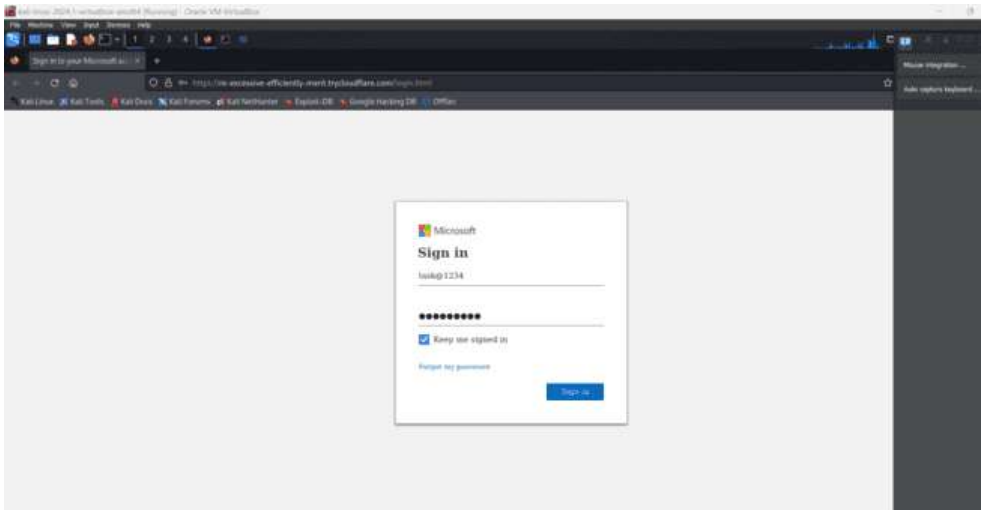
STEP 31. Select cloudfared option and press enter

STEP 32. Press N and enter

STEP 33. Right Click on the first url and click on open link



STEP 34. Enter username and password

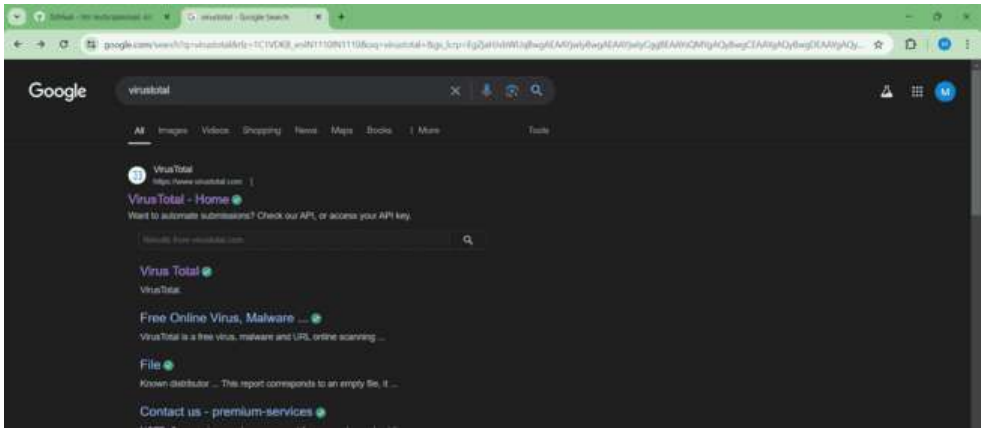


STEP 35. Open kali terminal again and see the details of the user

Here is the result

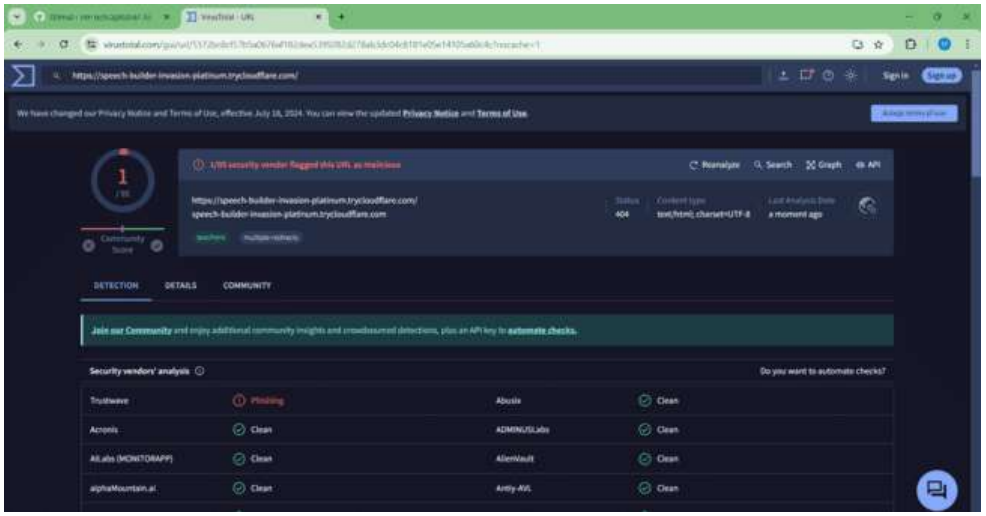
STEP 36. Open chrome browser

STEP 37. Search “Virustotal ”



STEP 38. Open virustotal

STEP 39. Enter the link in search bar and press enter



Here is the result

For Gmail

STEP 40. Open terminal in Kali

STEP 41. Write " git clone

https://github.com/htr-tech/zphisher.git

"

STEP 41. Write "ls" command

STEP 43. Write cd zphisher

STEP 44. Write “ls -l”

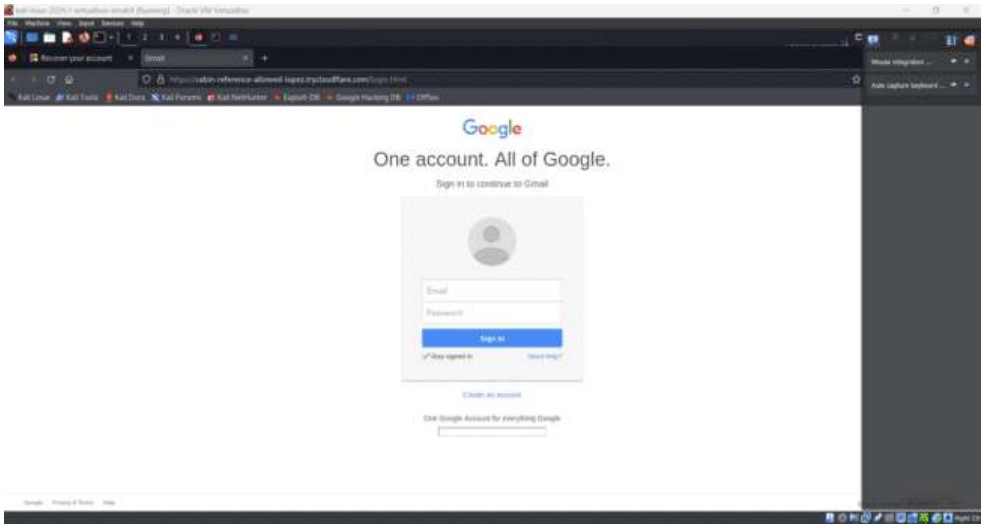
STEP 45. Write “./zphisher.sh”

STEP 46. Select option 4 for Microsoft 365 and press enter

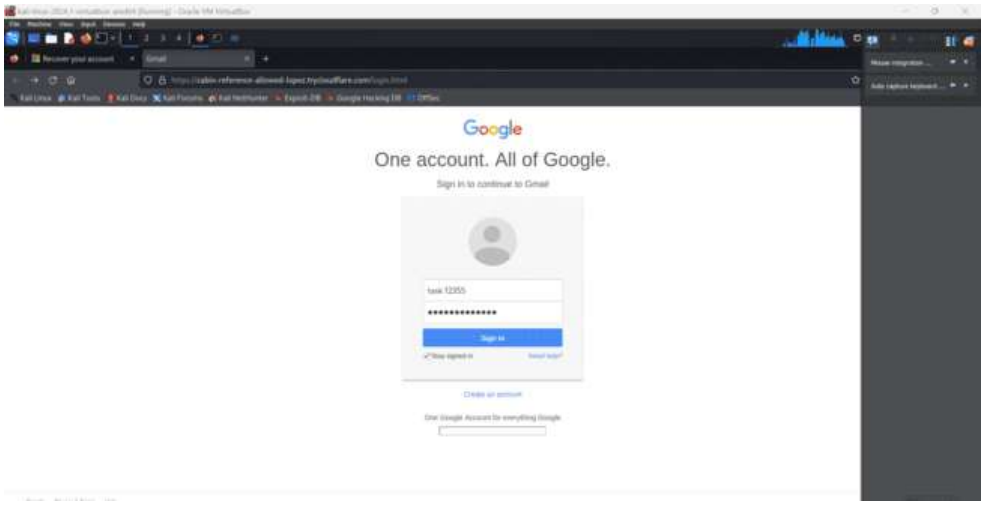
STEP 47. Select cloudfared option and press enter

STEP 48. Press N and enter

STEP 49. Right Click on the first url and click on open link



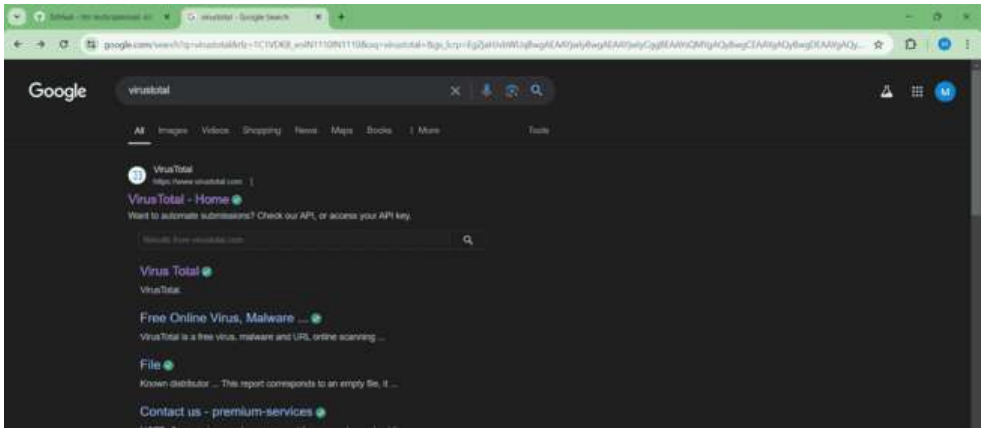
STEP 50. enter username and pass word



STEP 51. Open kali terminal again and see the details of the user

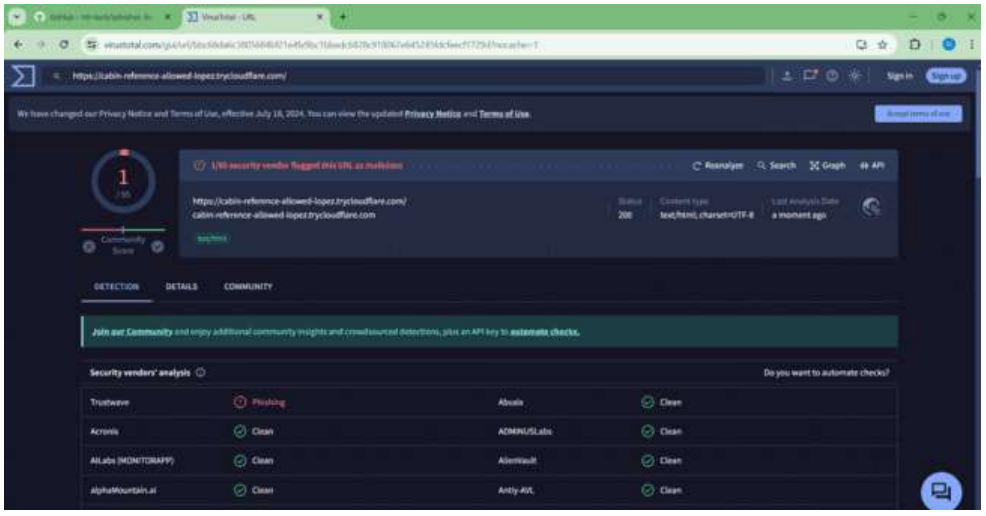
STEP 52. Open chrome browser

STEP 53. Search “Virustotal ”



STEP 54. Open virustotal

STEP 55. Enter the link in search bar and press enter

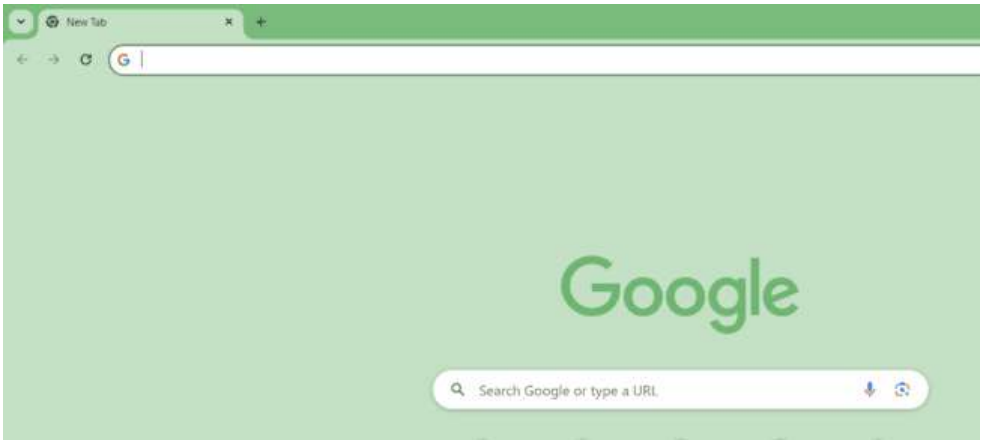


Here is the result

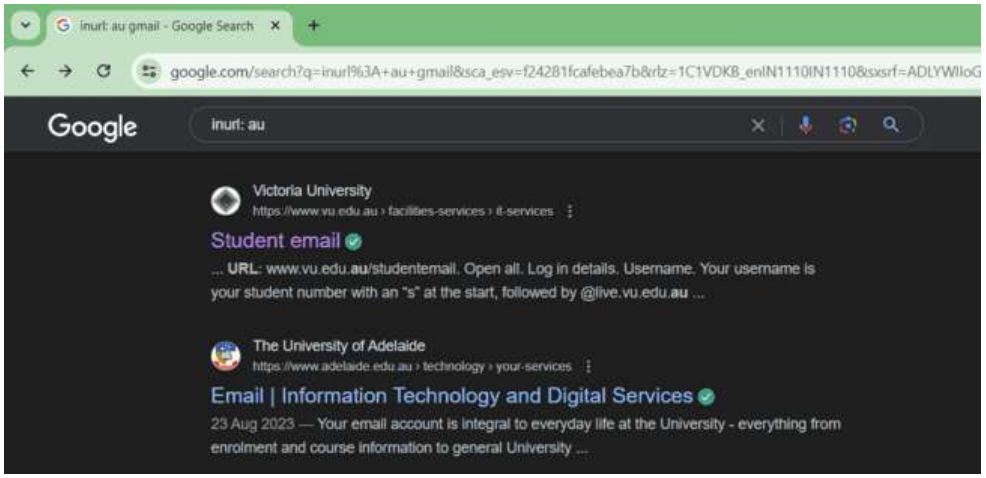
C) Identify any 3 Business email IDs where you can do an email spoofing attack.

>. Identify any 3 Business email IDs

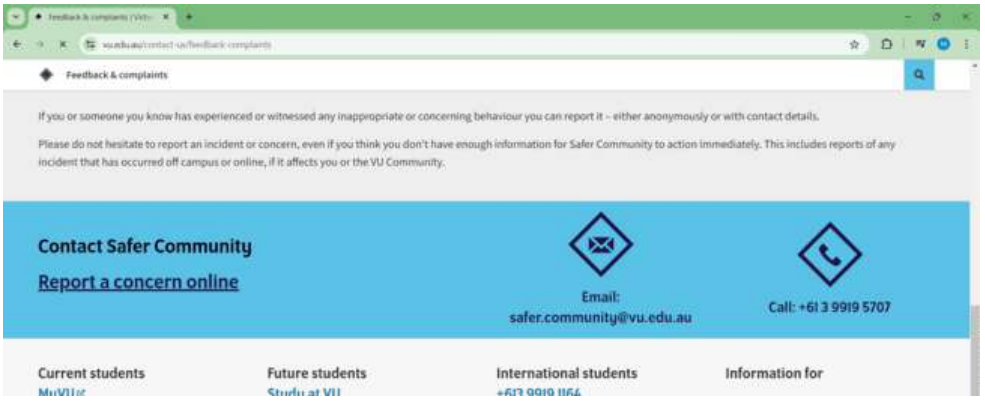
1. Open Chrome Browser



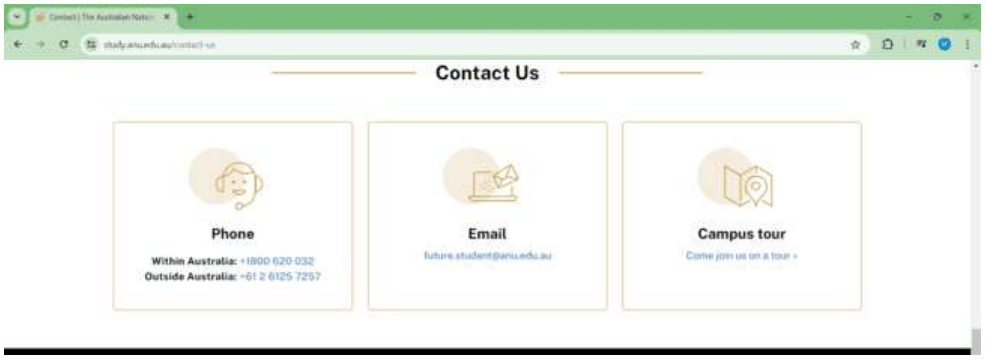
2. Type “inurl: au”

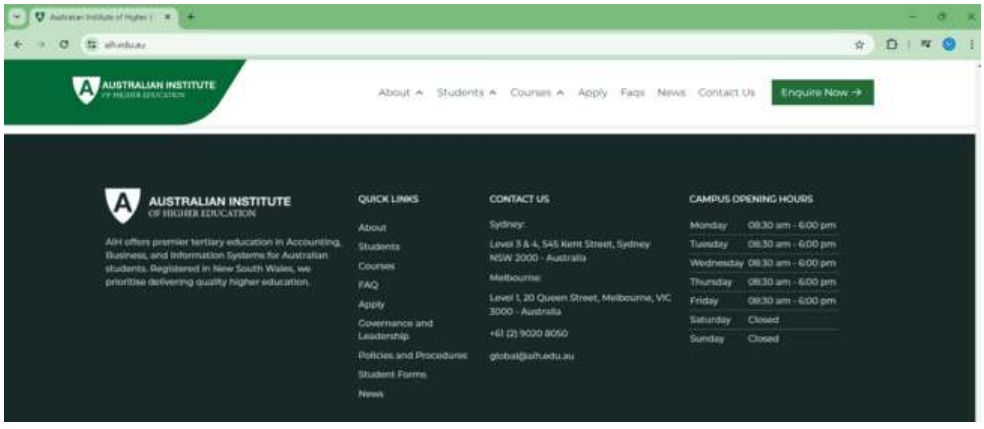


1. Open the website and search for a business email



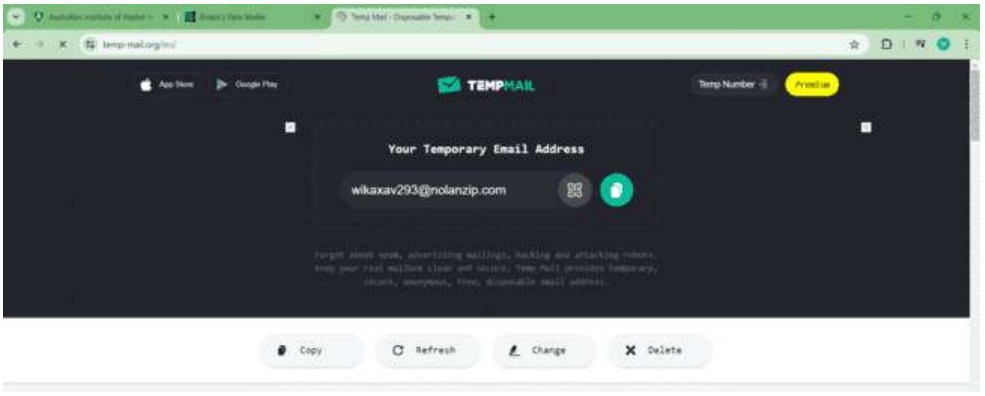
2. Similarly do the same for 2 more websites



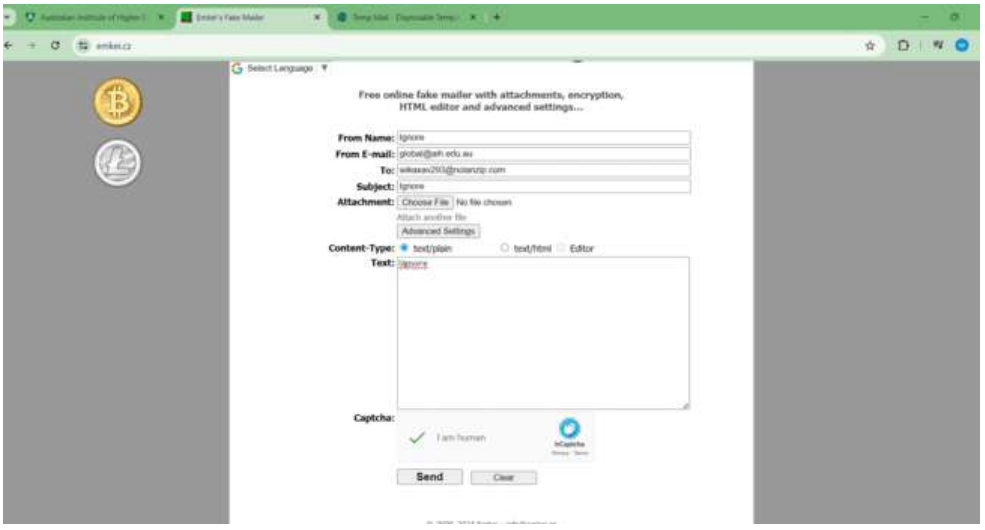


1. Now open emkies fake mailer

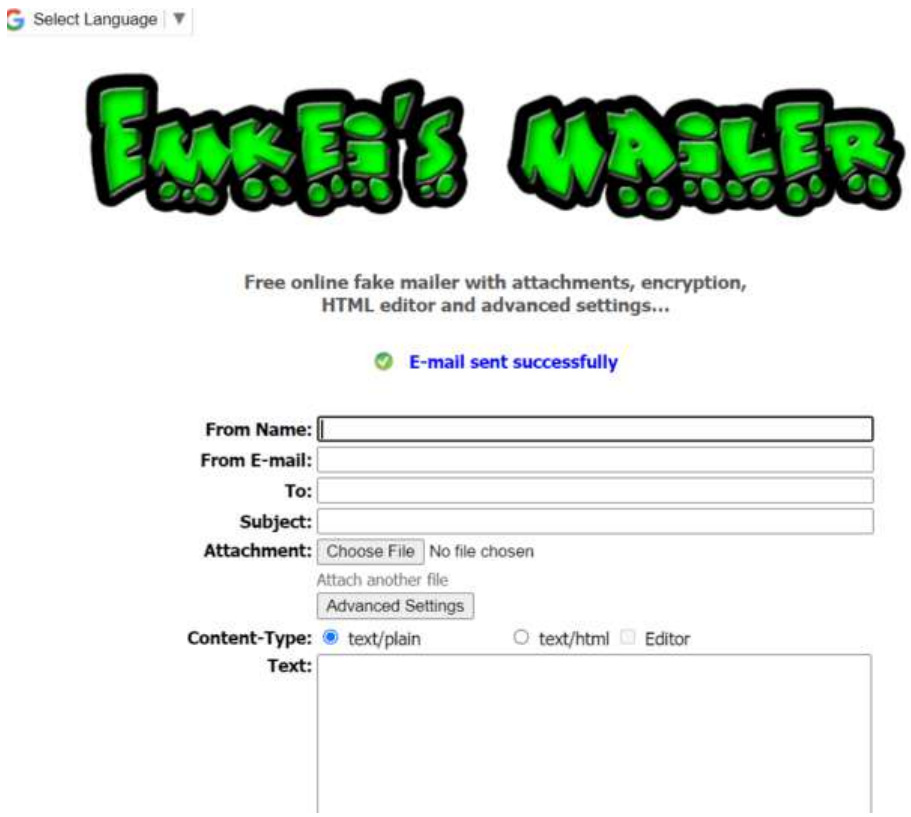
2. Now generate a temperorary email id form temp mail to receive the mail



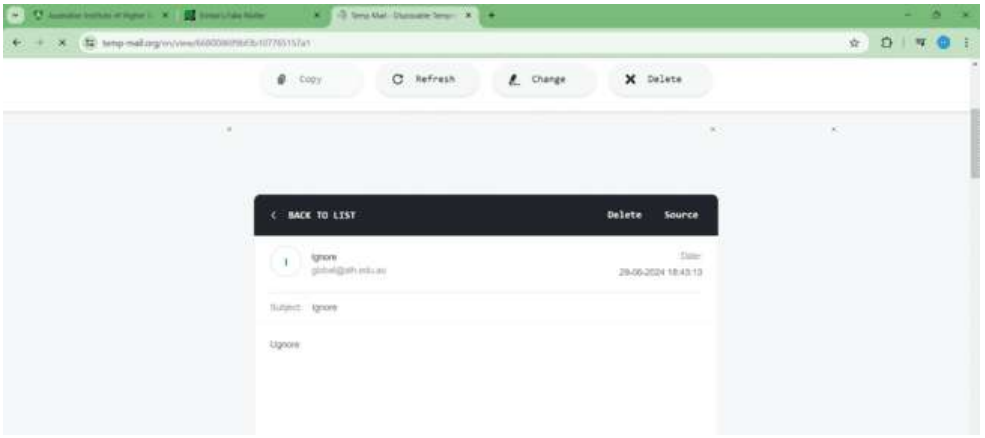
3. Now fill the details in emkies form



4. Clcik on send

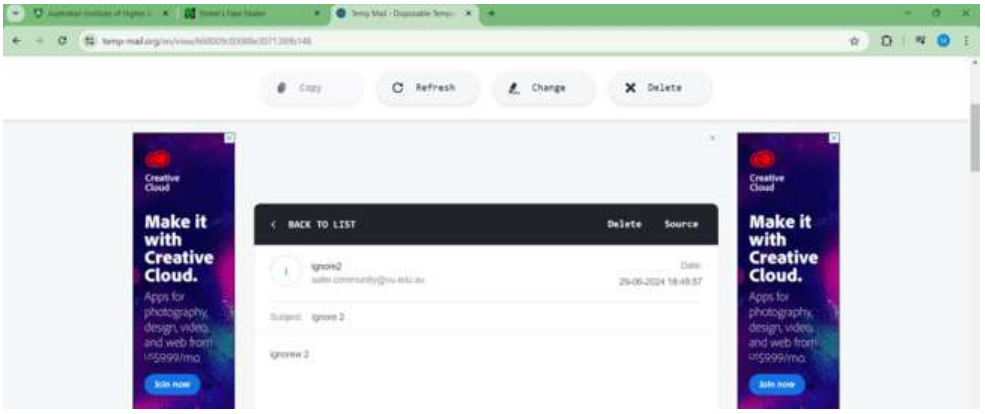


5. Now open temp mail and check inbox , we received a mail

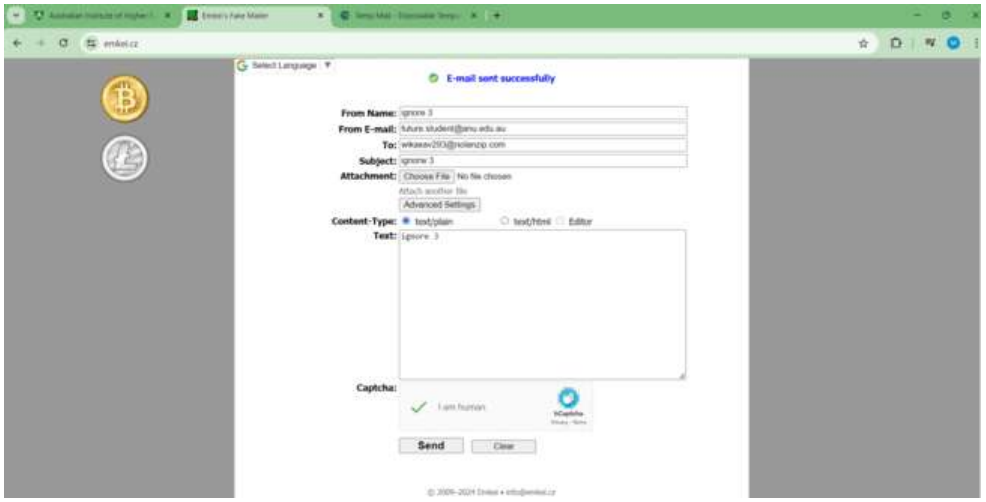


1. For next website safer.community@vu.edu.au

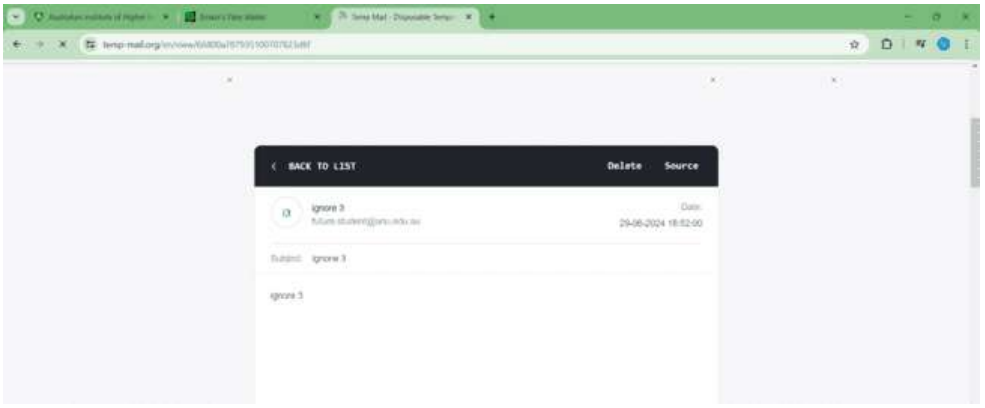
1. Click on send and check inbox



2. Now for our 3rd mail future.student@anu.edu.au



1. Click on send and check inbox



Summary-

This document provides step-by-step instructions for conducting various technological operations, most of which involve terminal commands in Kali.

We performed how to find subdomains of any website. The operations detailed include cloning a GitHub repository zPhisher, executing commands in a cloned repository, running a phishing script, and analyzing potential phishing URLs in VirusTotal. The document also outlines procedures for identifying business email IDs for potential email spoofing attacks.

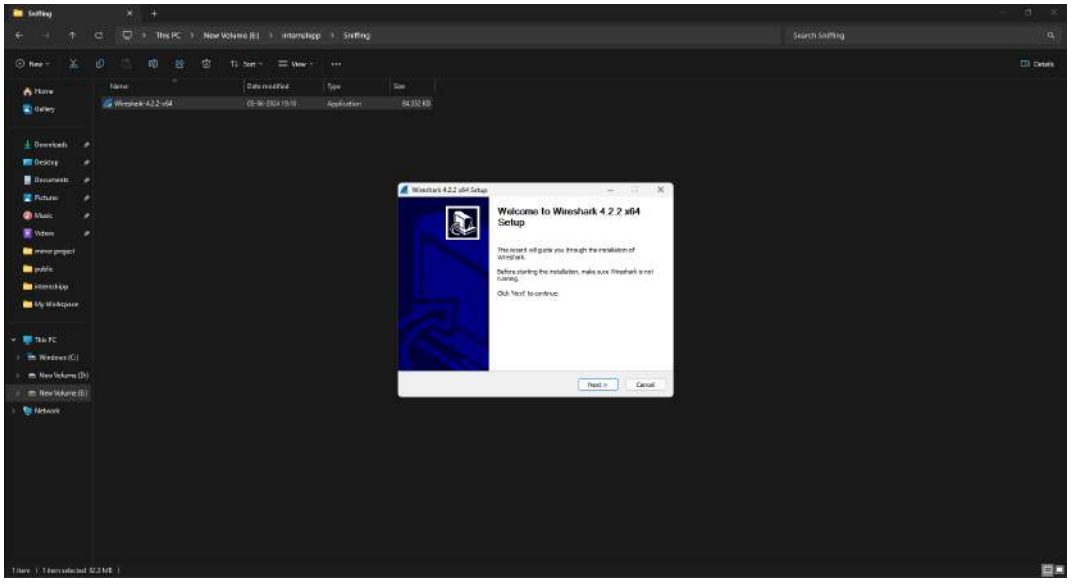
23EO5-ST#IS#6653-Task-4

Objective-01: Sniffing - Identify the websites that have vulnerable protocols to sniff.

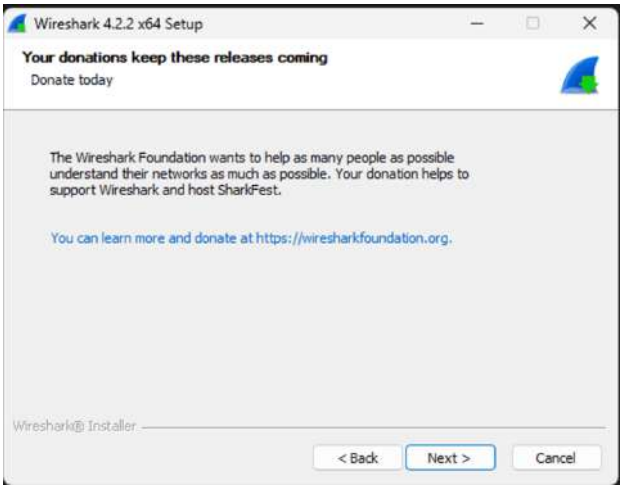
- HTTP
- FTP
- POP

Sniffing - Sniffing refers to the process where a malicious user captures and inspects data as it travels over a network. This data could include details such as usernames, passwords, credit card numbers, or other sensitive information. Sniffing can be used both for malicious purposes, such as stealing information, or for network troubleshooting and optimization.

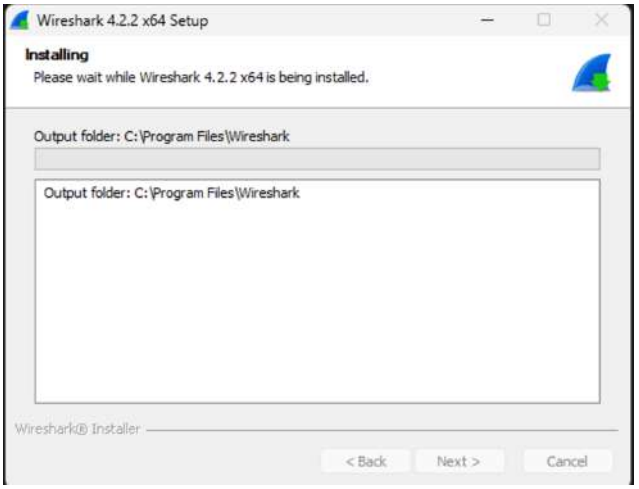
STEP-1: Open Wireshark-4.2.2-x64 Setup exe file.



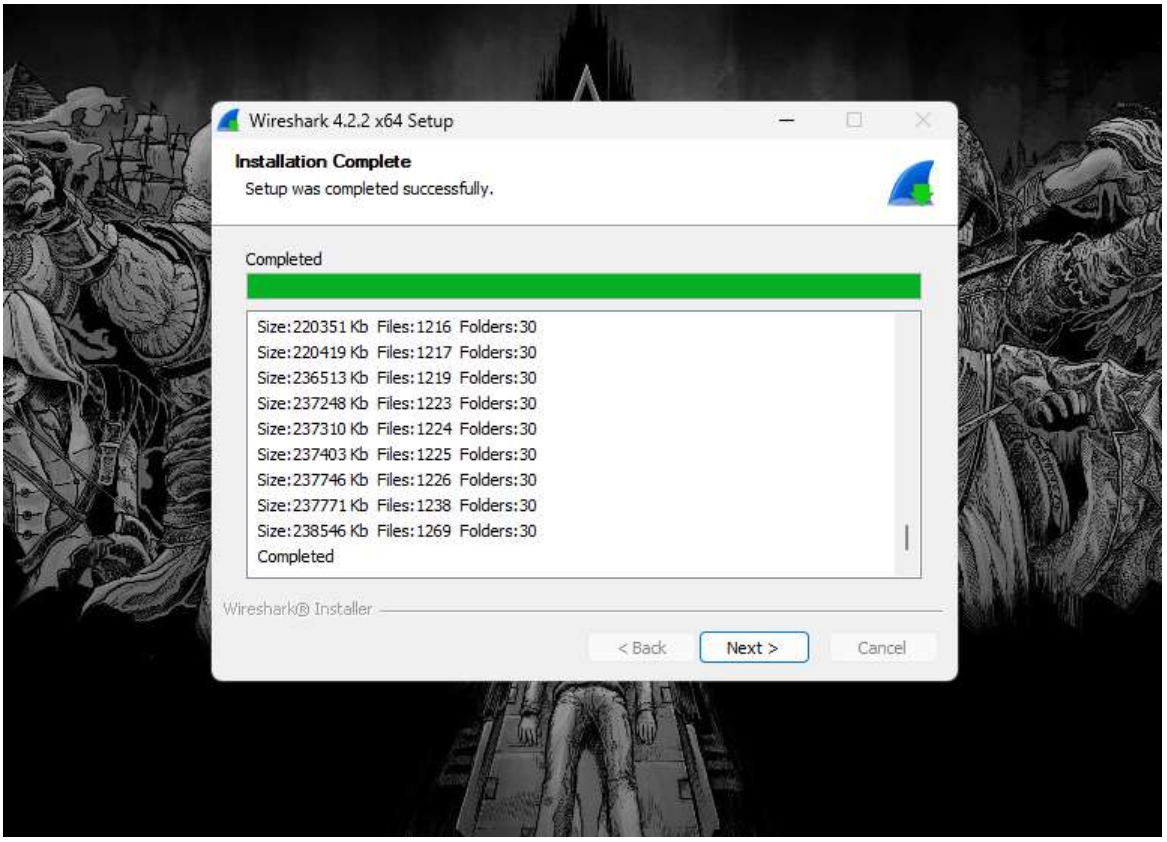
STEP-2: Click on next.



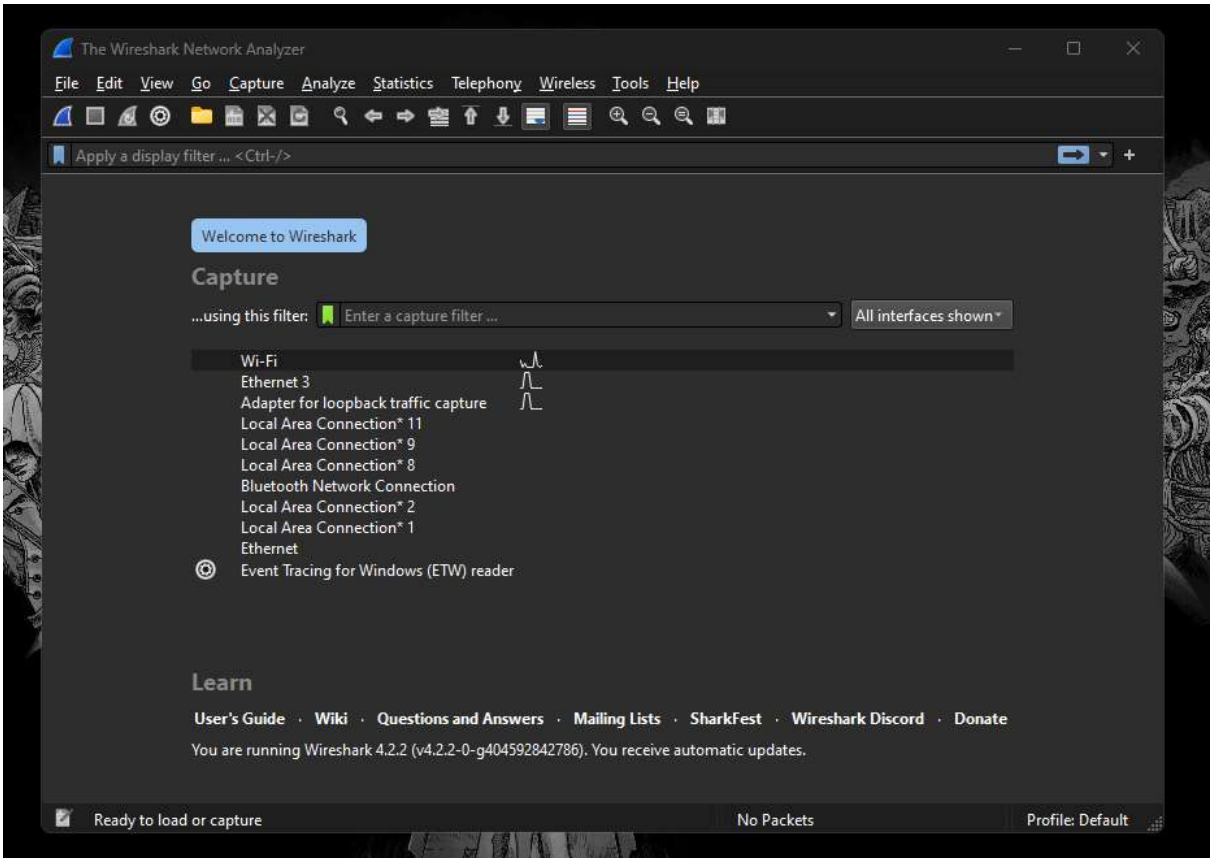
STEP-3: **Continue the process and select installation directory.**



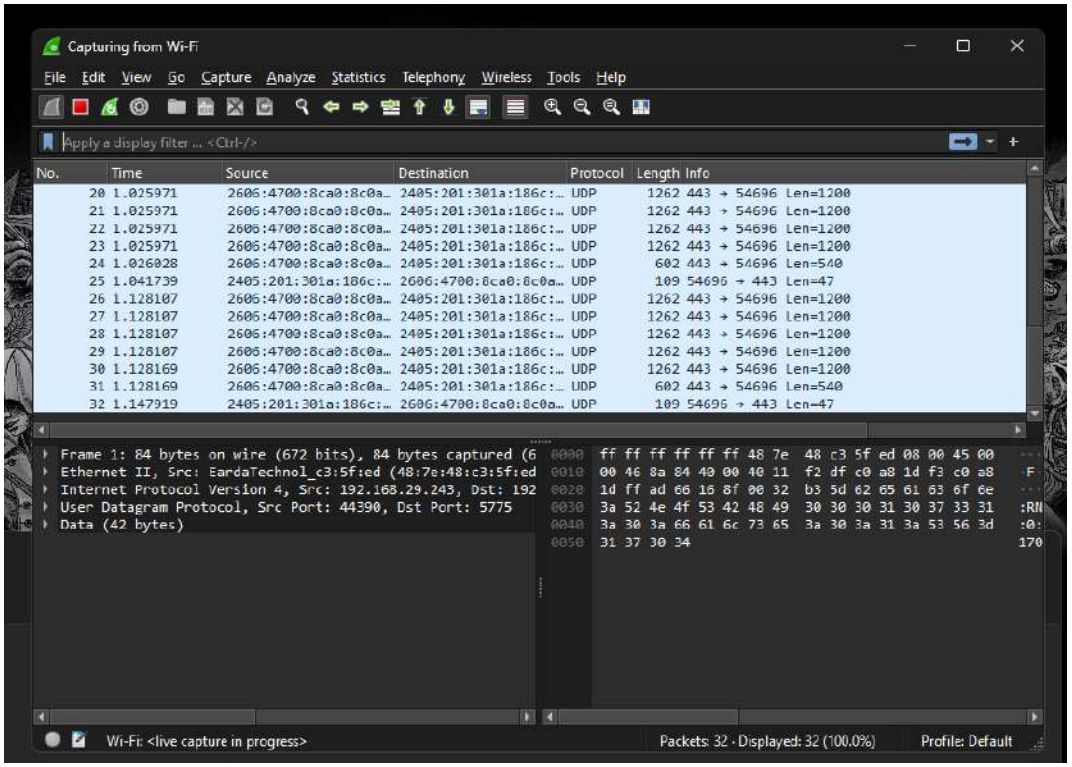
STEP-4: **Click on next.**

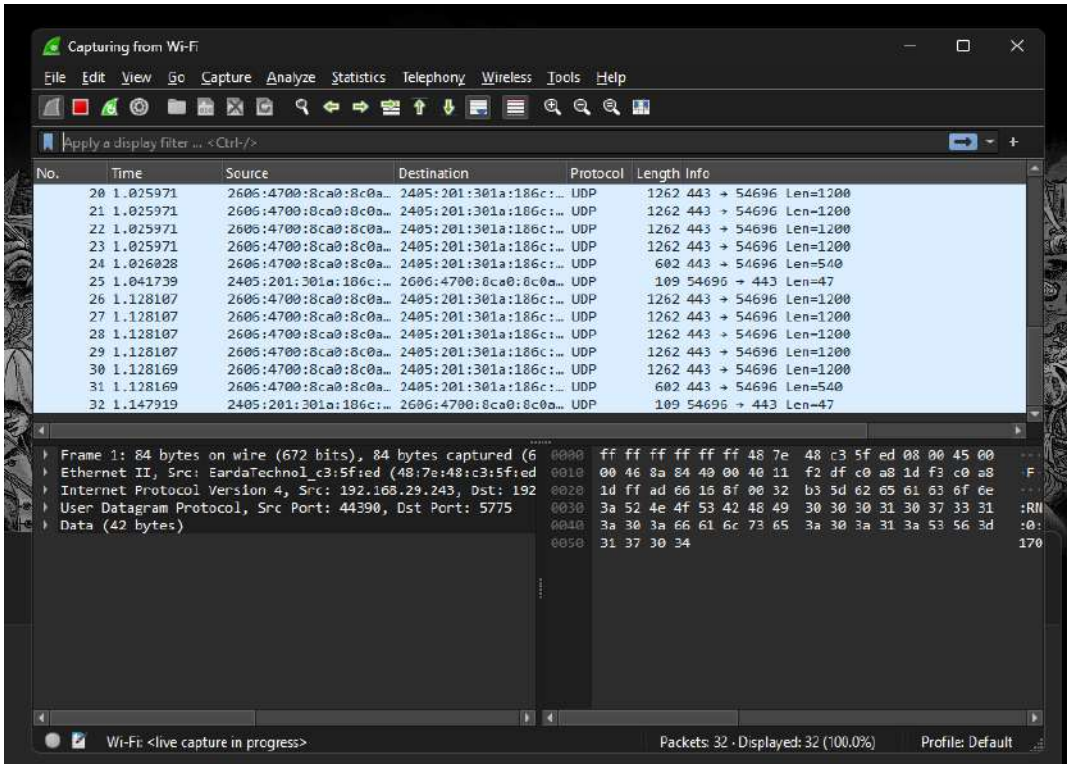


STEP-5: Open wireshark tool.

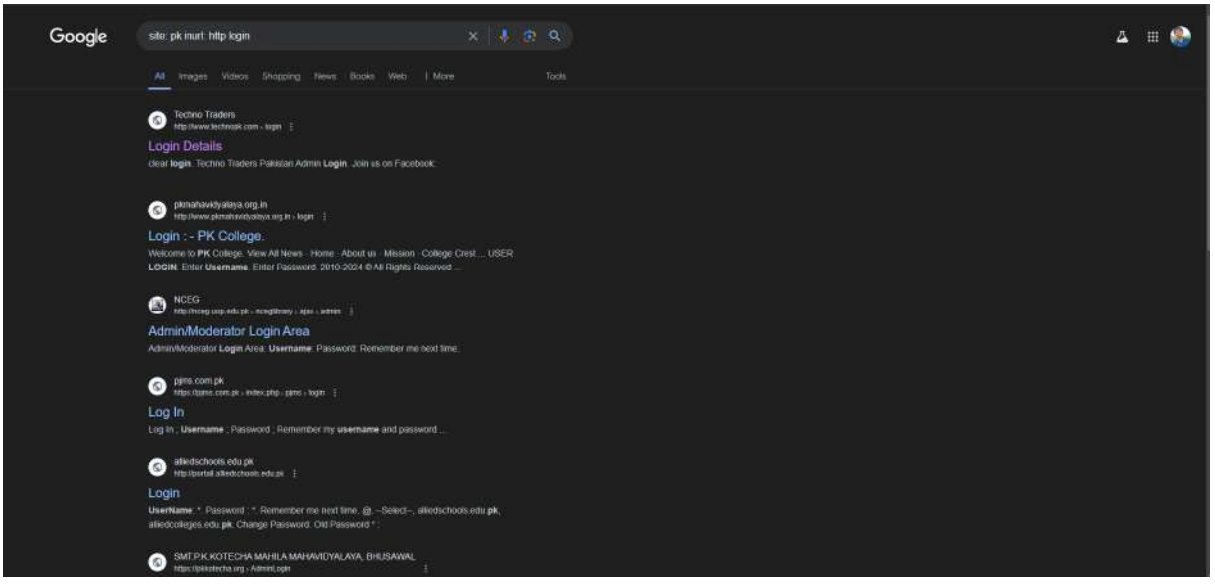


STEP-7: Open the WIFi adaptor.

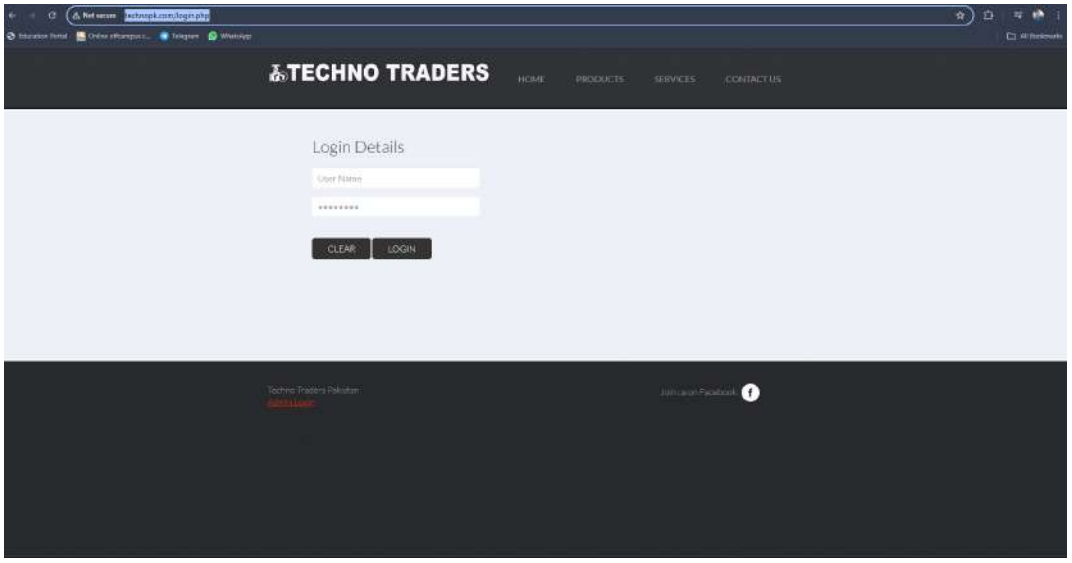




STEP-8: Open browser and search the following `site: pk inurl: http login`

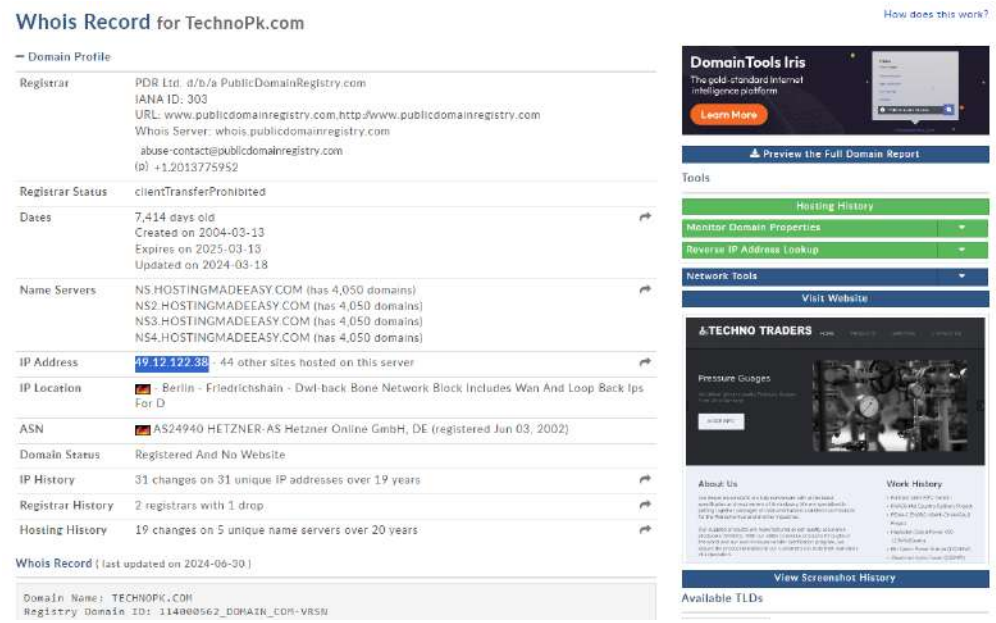


STEP-09: now copy the website url.

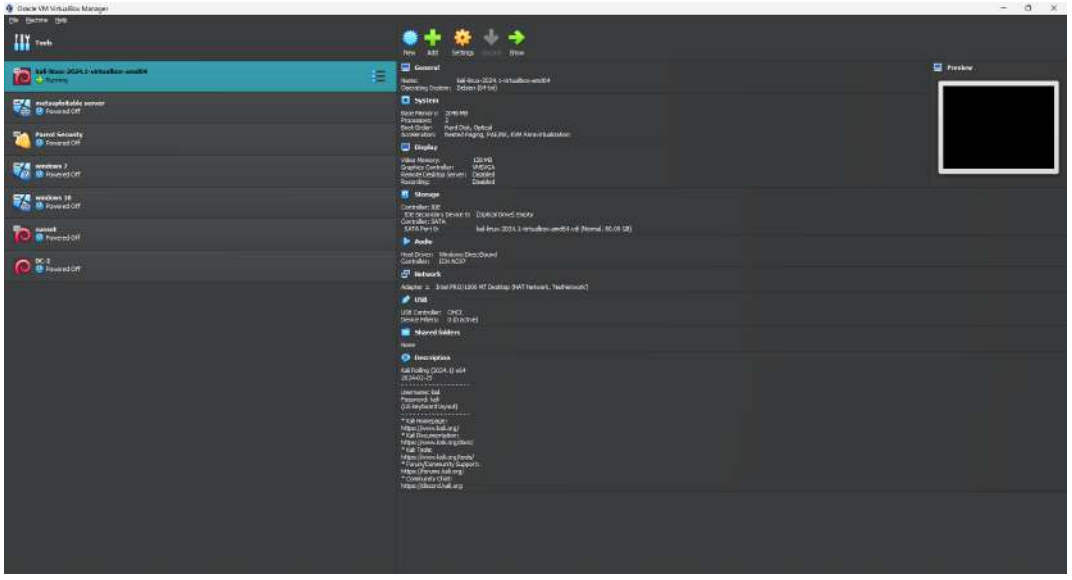


STEP-10: now go to https://whois.domaintools.com/ and paste the copied website url and click on search.

STEP-11: Note down the ip address.

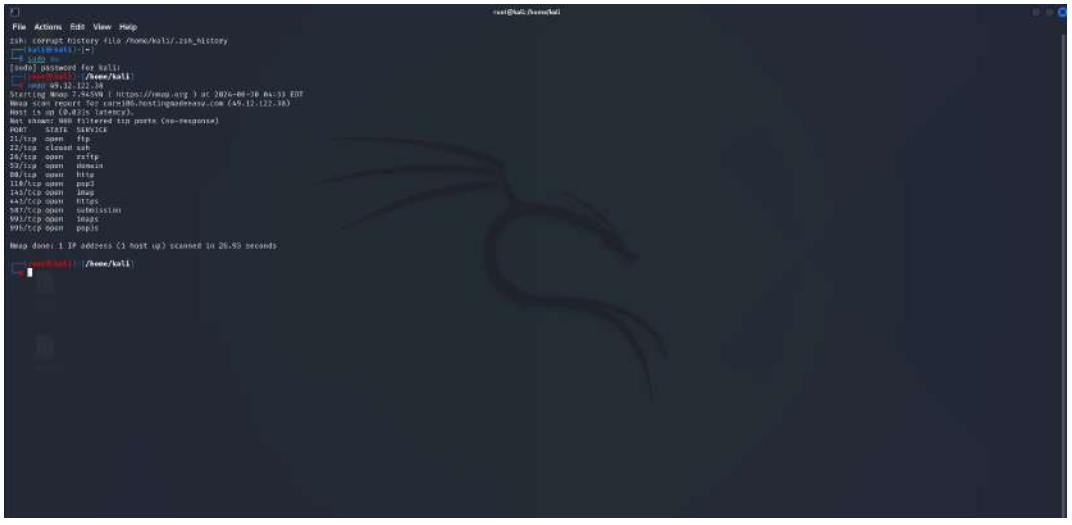


STEP-12: now open kai-linux virtual machine



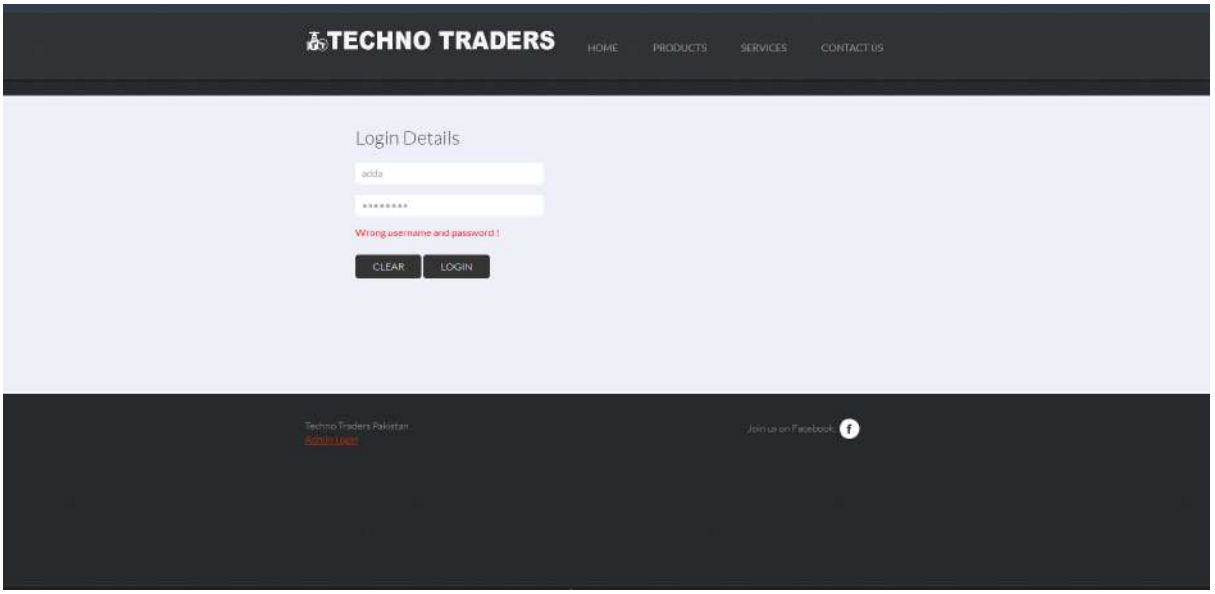
STEP-13: open terminal and go root user.

STEP-14: Run the command `nmap <target_ip>` to get open ports

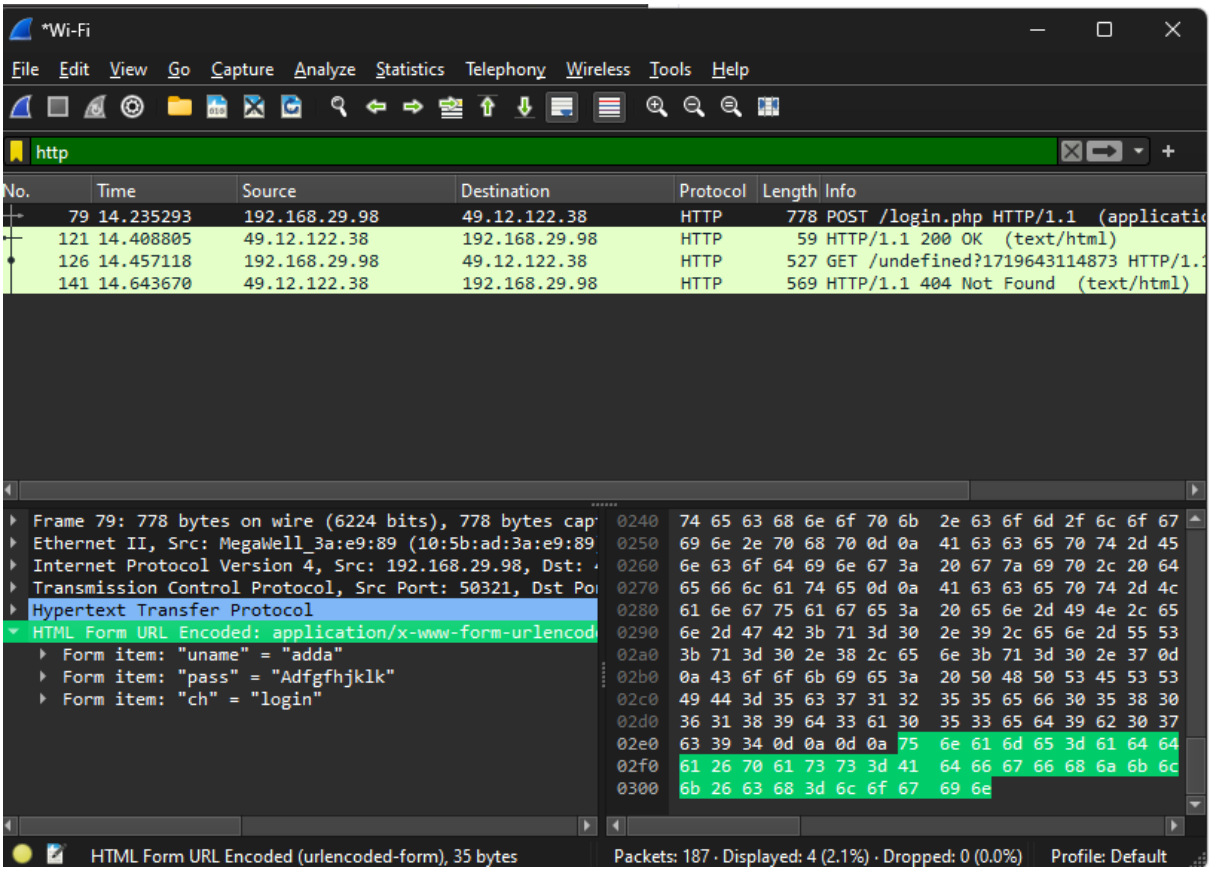


Since the website is vulnerable to all thre protocols we will use it to sniff the data.

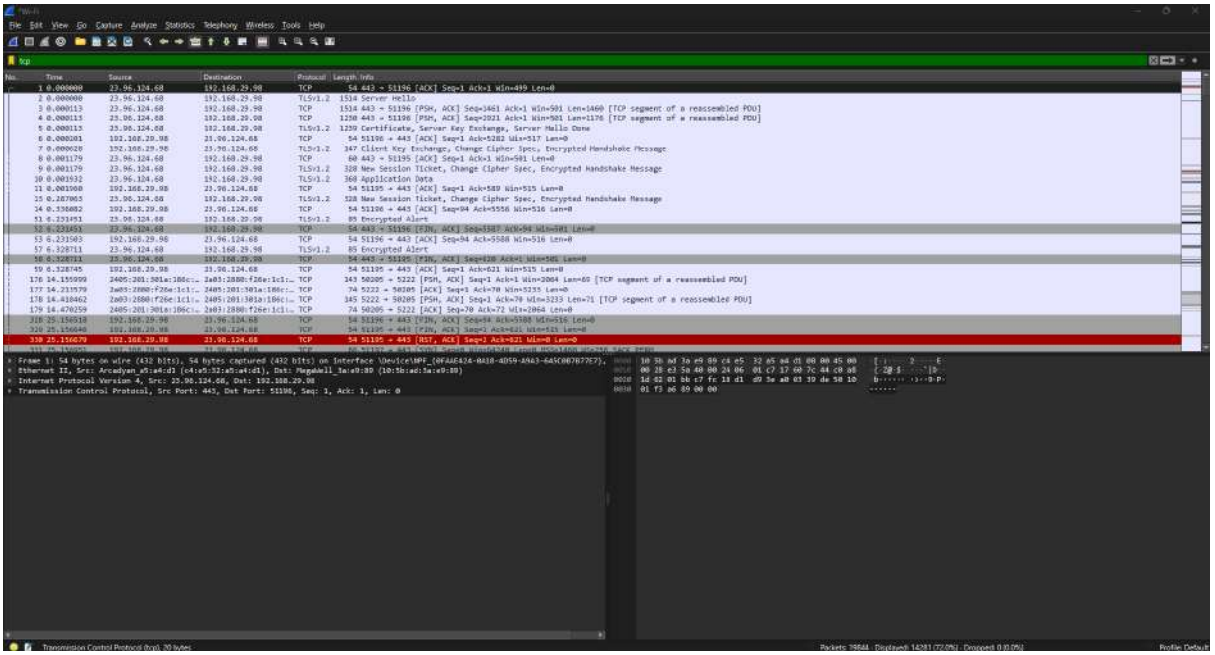
STEP-15: Now open the website and enter random login credentials.



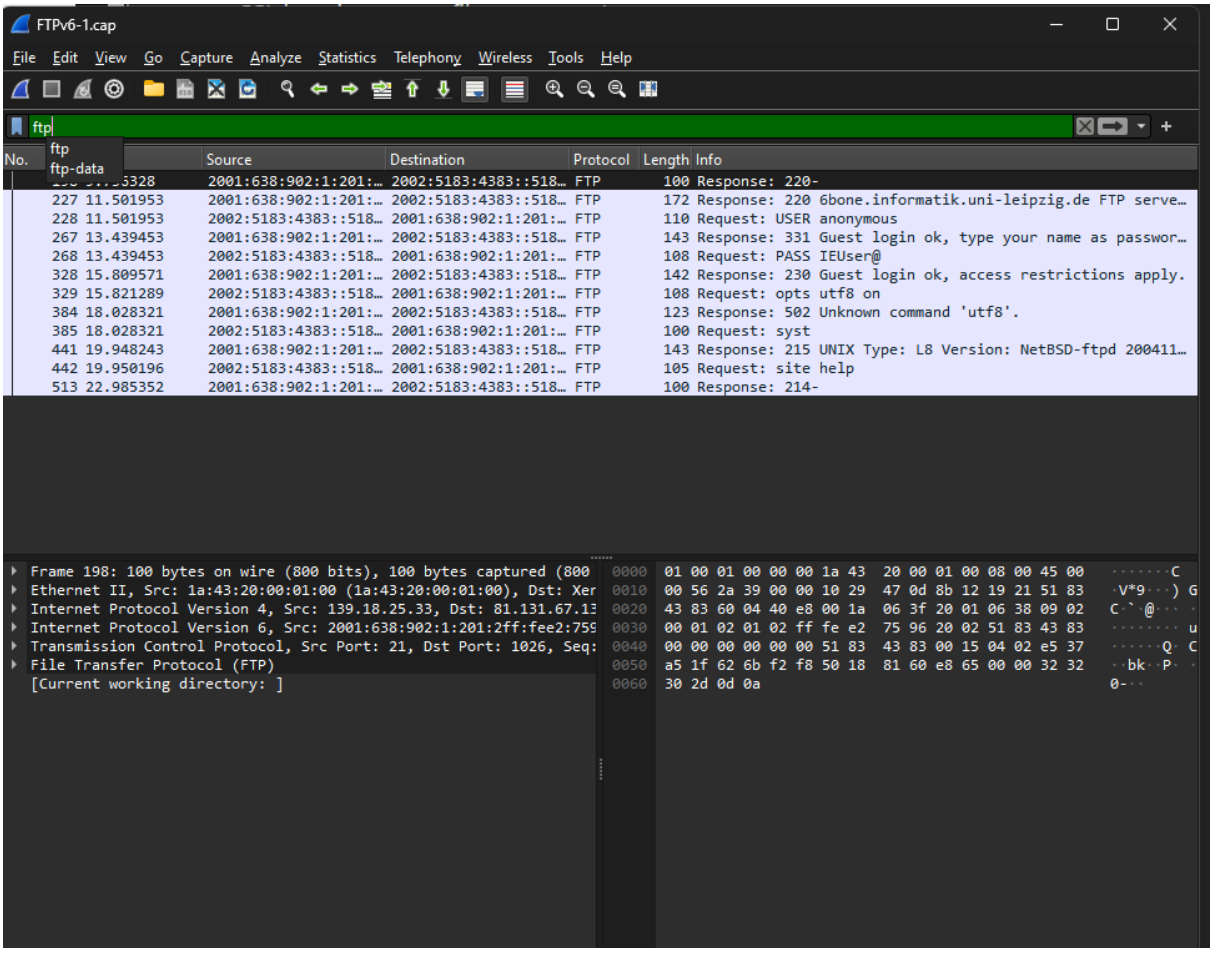
STEP-11: Simulatneosuly check wirehshark tool and search for `http` in the search box.



As we can see the credentials we used in the website is shown at the bottom. The website is vulnerable to http protocol.

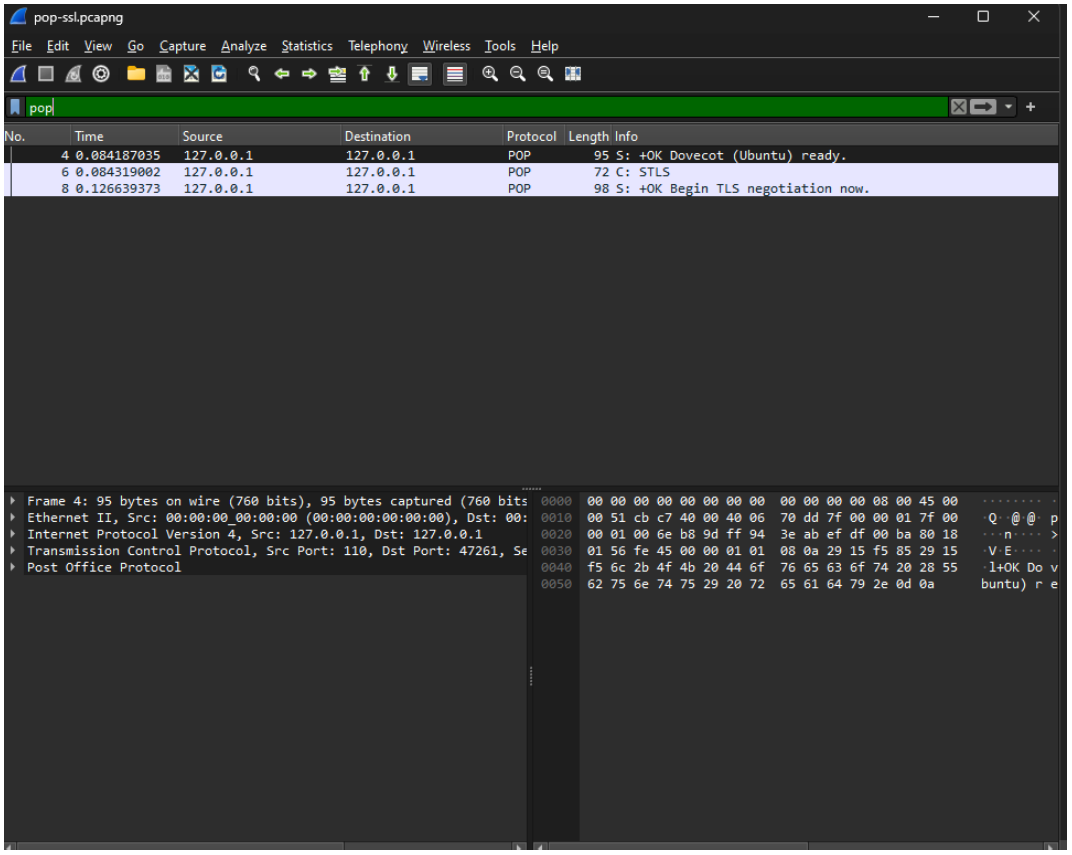


STEP-12: Simultaneously open wireshark tool and search for `ftp` in the search bar.



Wireshark has captured ftp protocol.

STEP-13: search `pop` in search bar to sniff data



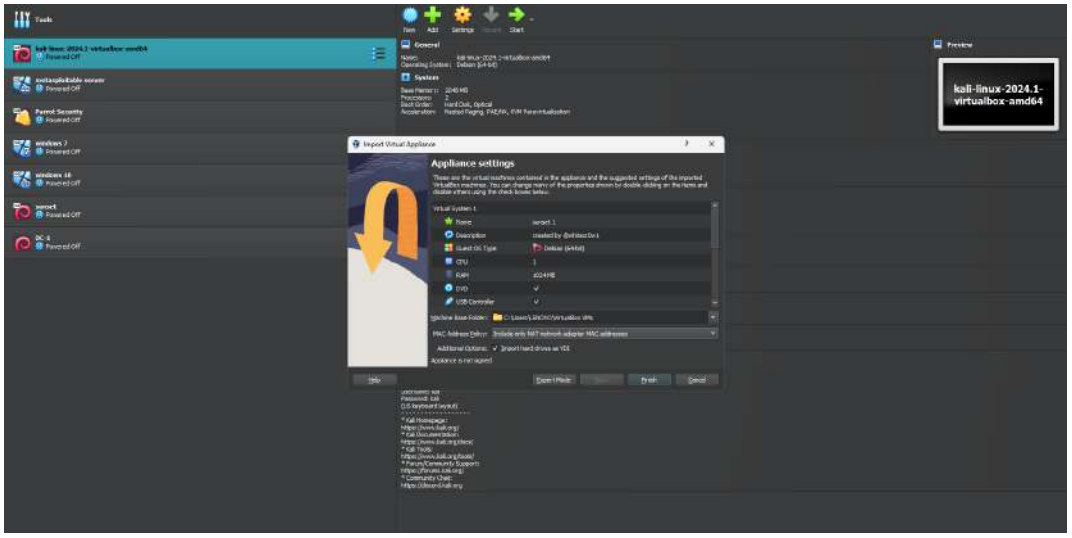
Conclusion:

Wireshark is powerful network protocol analyzer. It is used to capture and interactively browse the traffic running on the computer. here we used it to sniff data from a website capturing ftp,http and pop protocol. Wireshark is often used for network troubleshooting, analysis, software and protocol development, and education.

Objective-02: Server Hacking – Crack the servers and find the flags

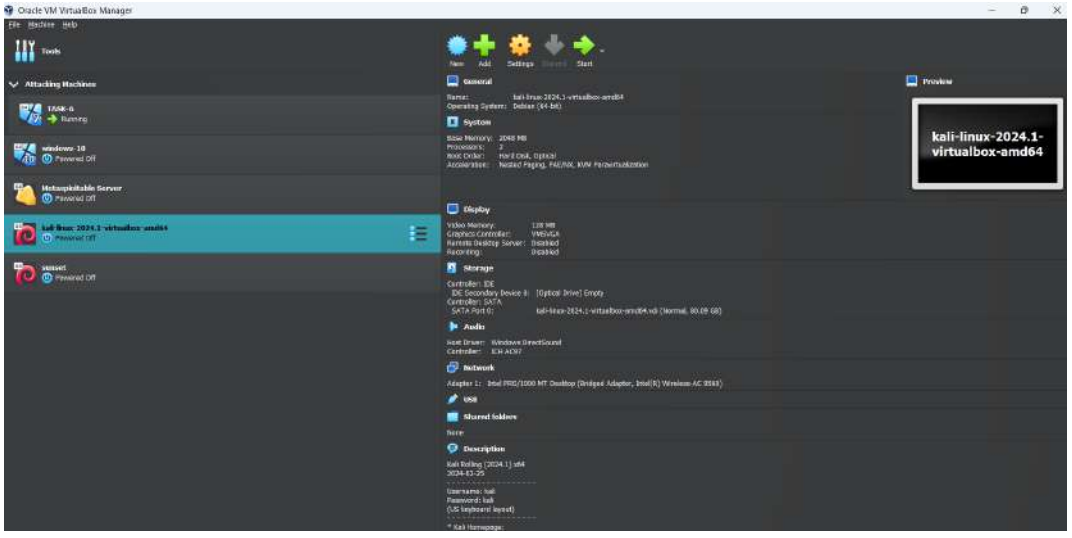
- Exploit the SUNSET server
- Exploit the DC-1 server

STEP-1: Import both sunset server and Dc-1 server in the virtual machine box.

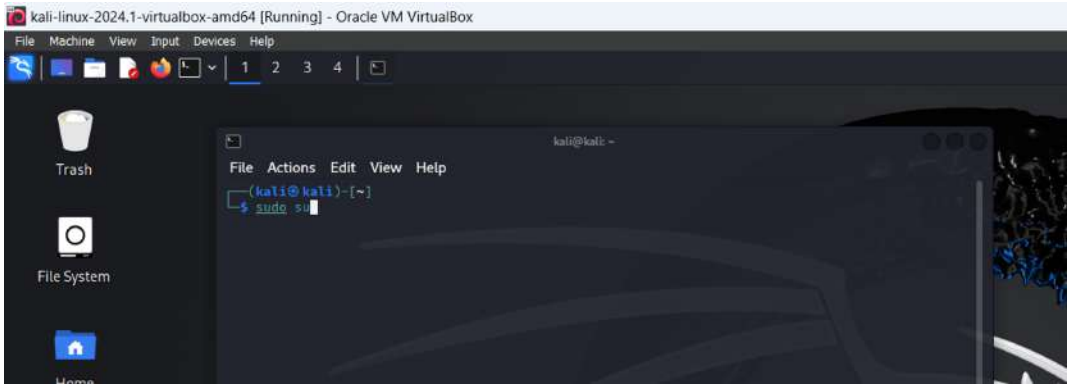


STEP-2.1: Open sunset server.

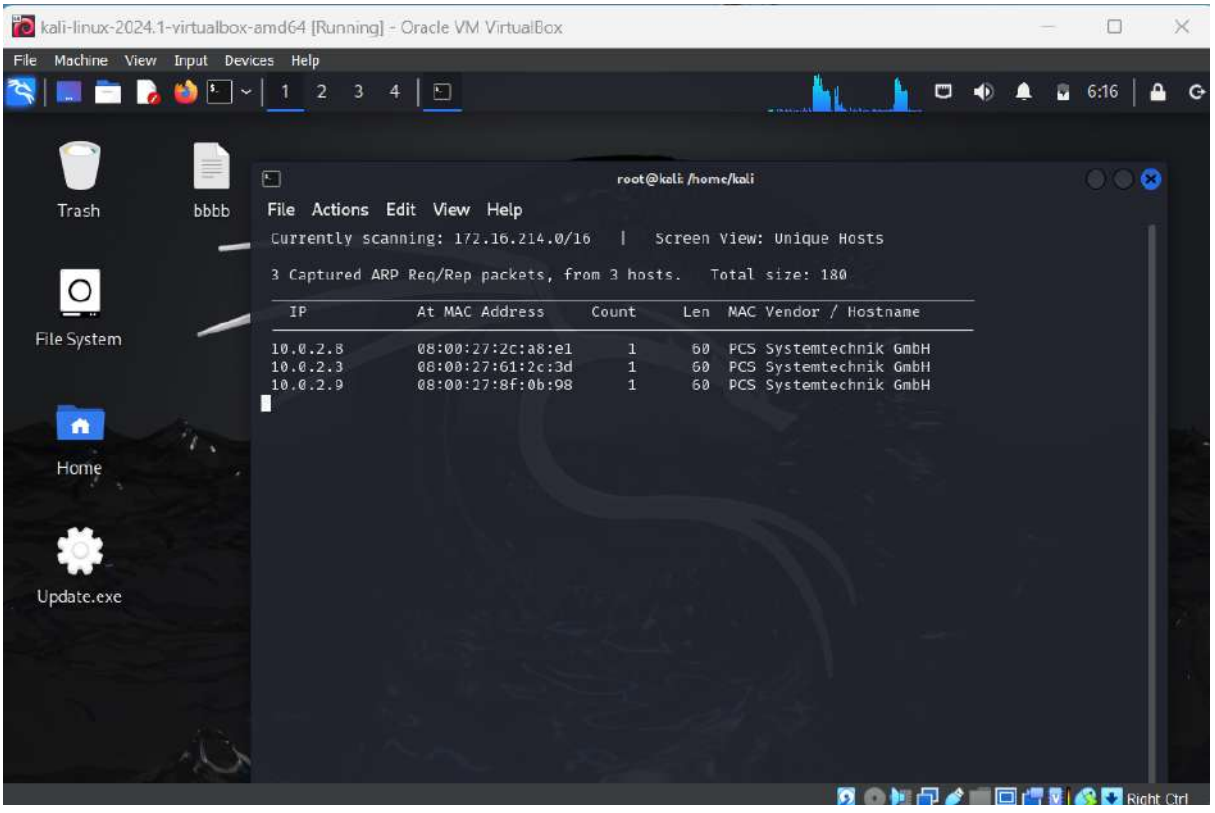
STEP-2.2: Open kali-linux virtual machine(please make sure both sunset server and kali linux are on same network)



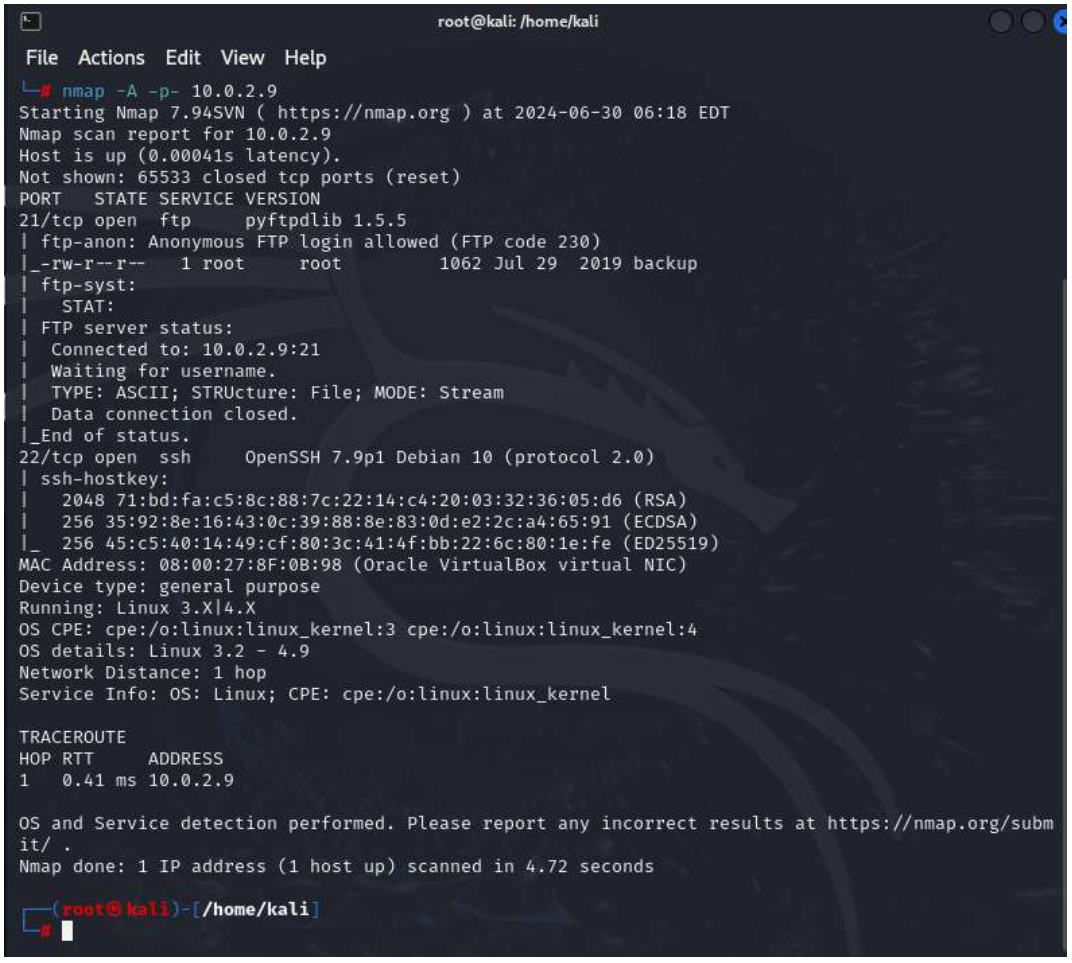
STEP-2.3: Now open a terminal in kali linux and enter root user.



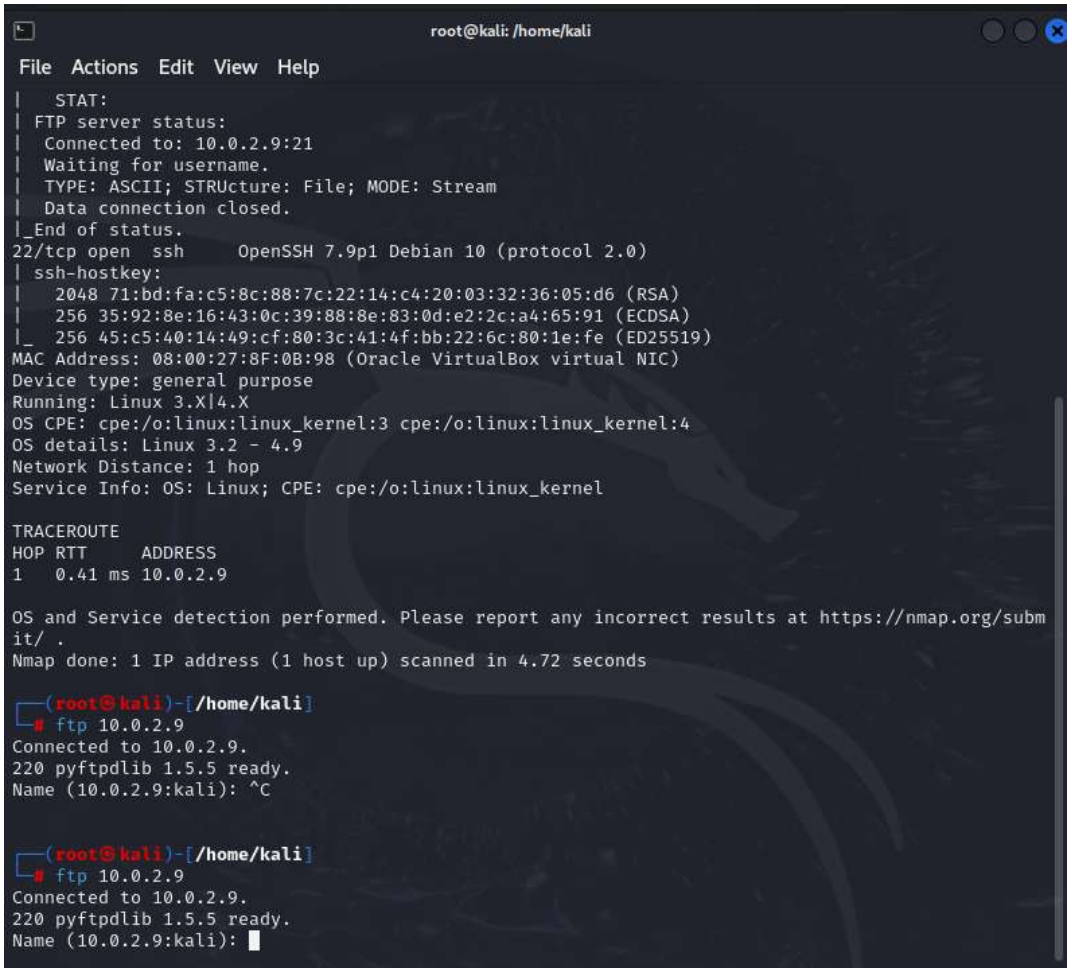
STEP-2.4: once in root user run `netdiscover` to get the ip address of the sunset server.



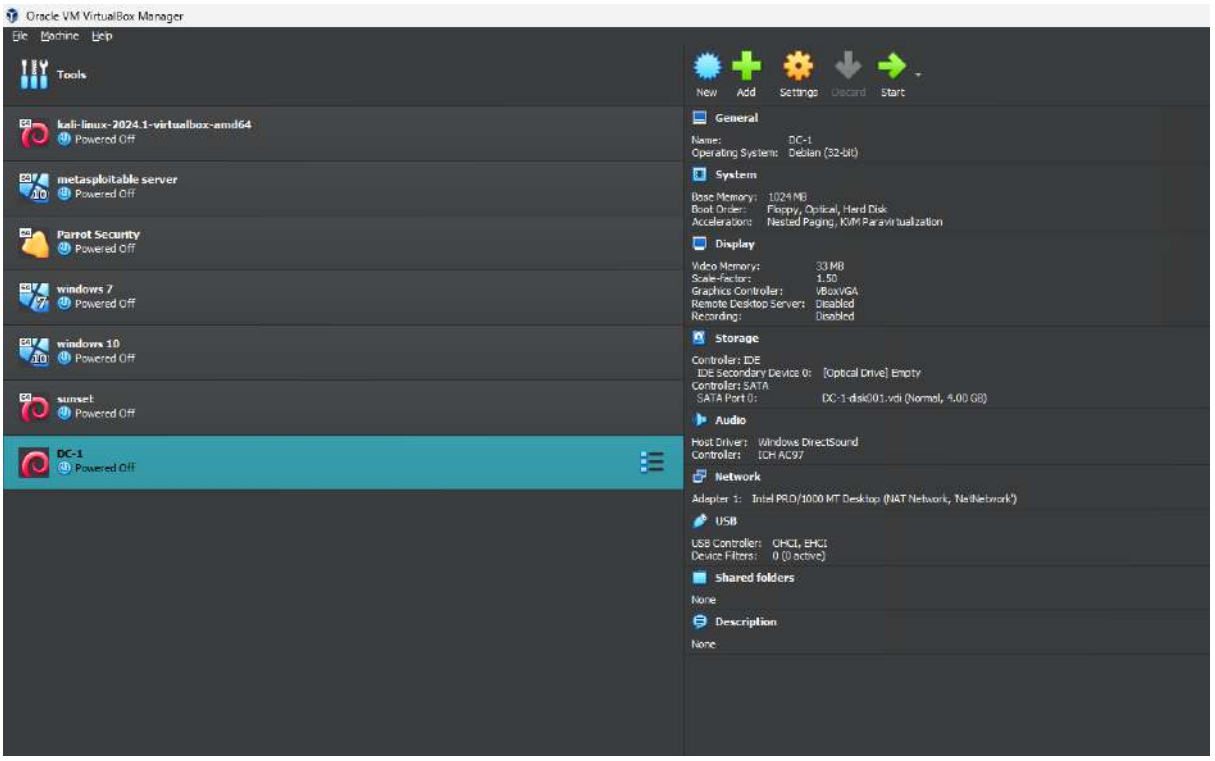
STEP-2.5: Once ip is gained, run `nmap -A -p- <target_ip>` to get the open ports in the particular server.

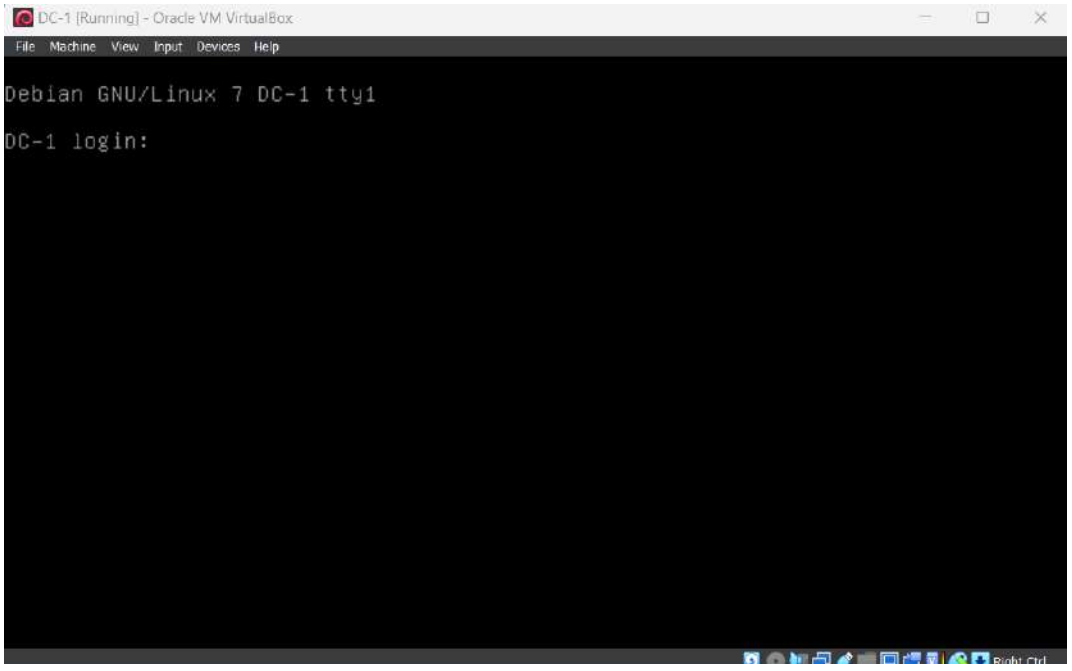


STEP-2.6: Run command `ftp<target_ip>` to gain insights.

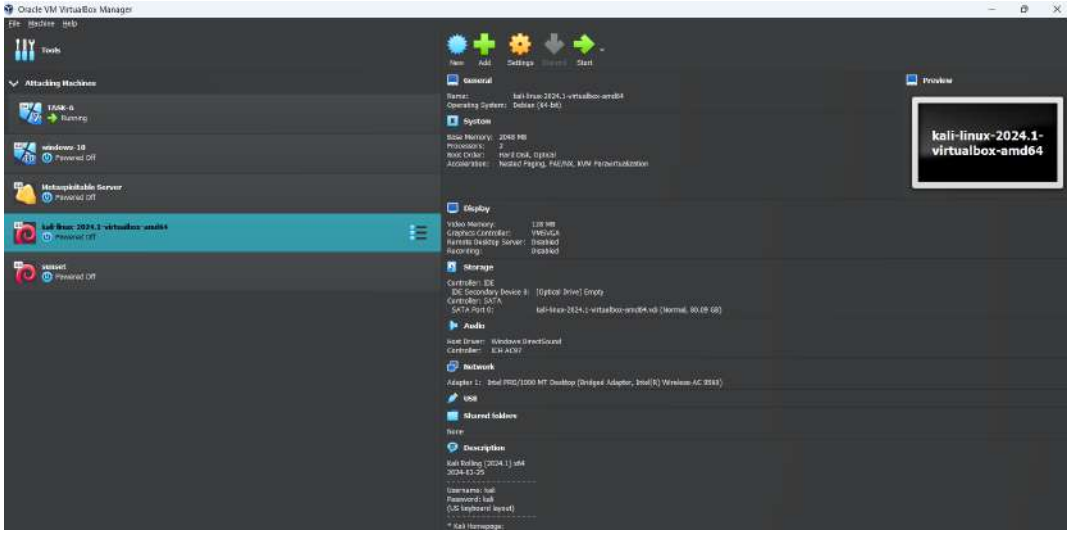


STEP-3.1: Open DC-1 server.

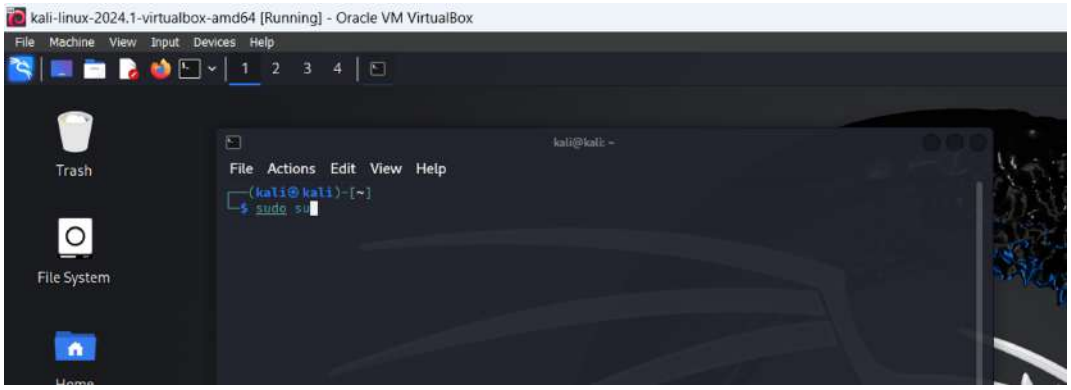




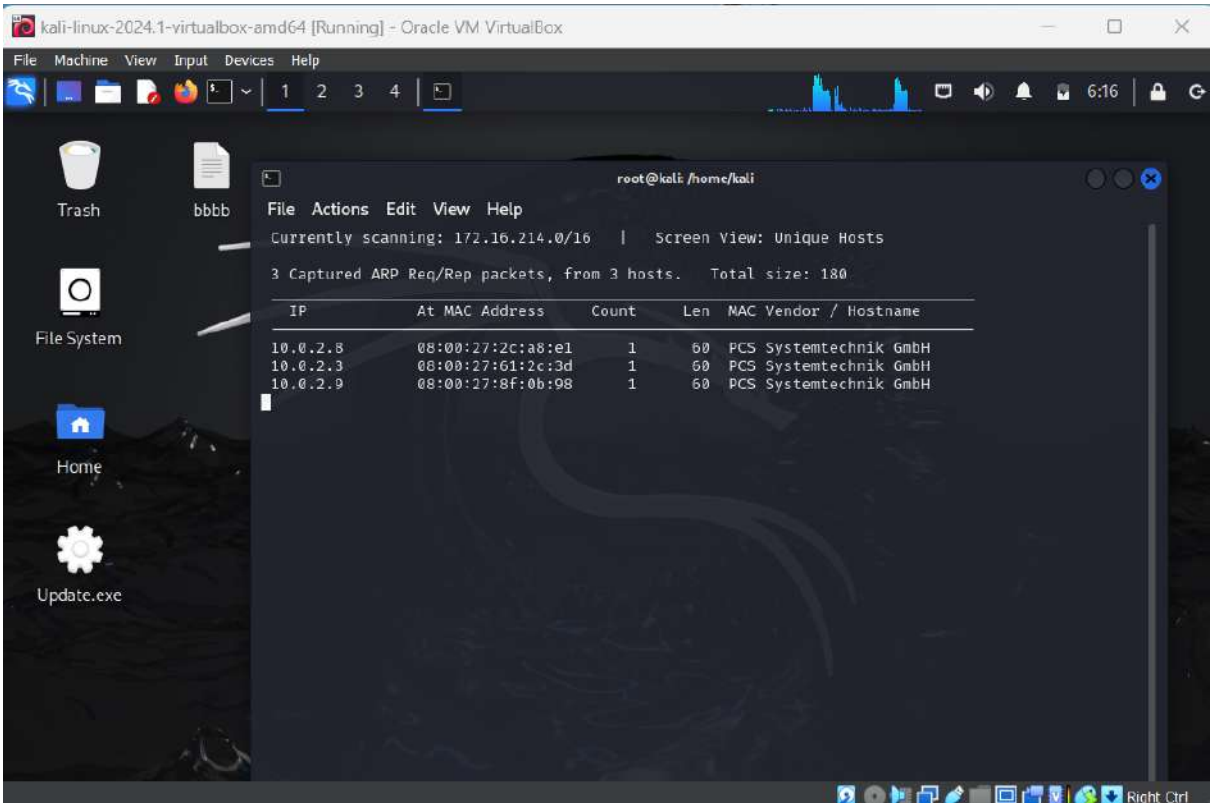
STEP-3.2: Open kali-linux virtul machine(please make sure both sunset server and kali linux are on same network)



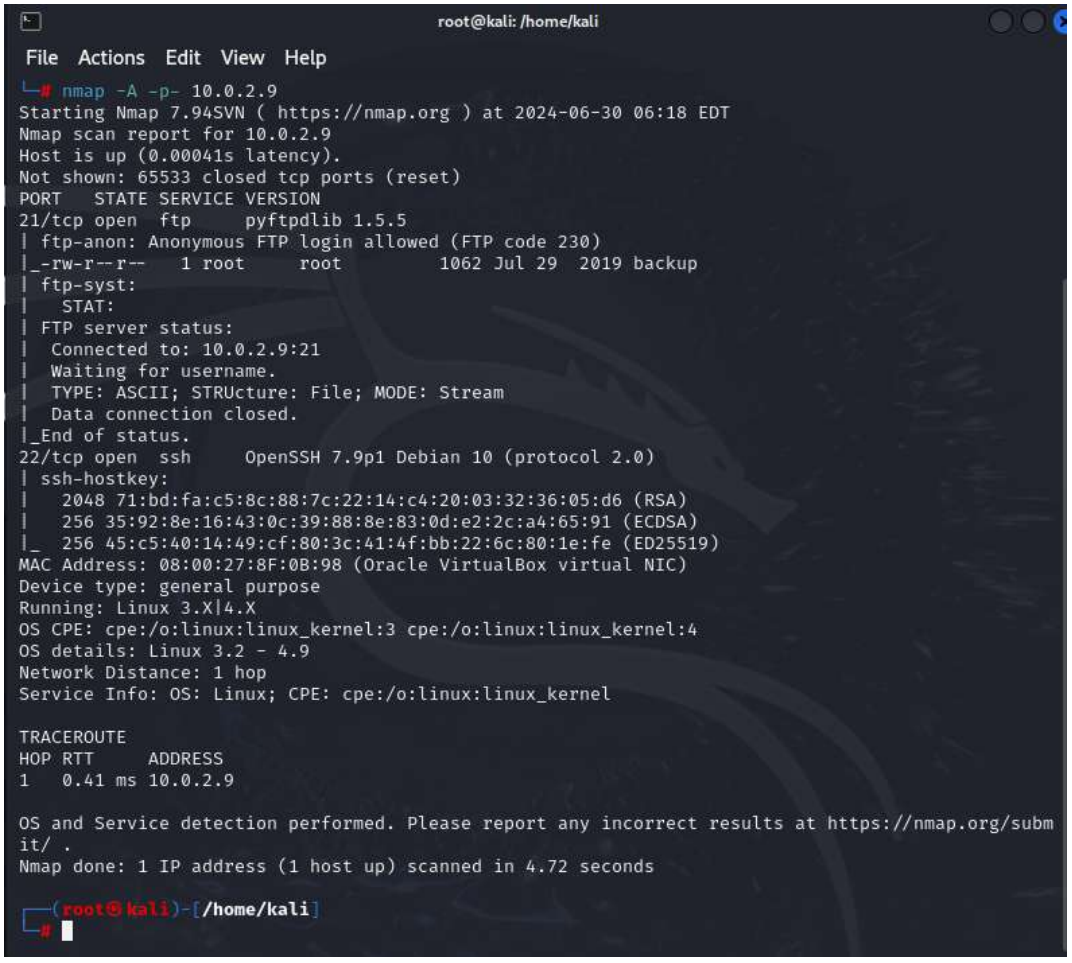
STEP-3.3: Now open a terminal in kali linux and enter root user.



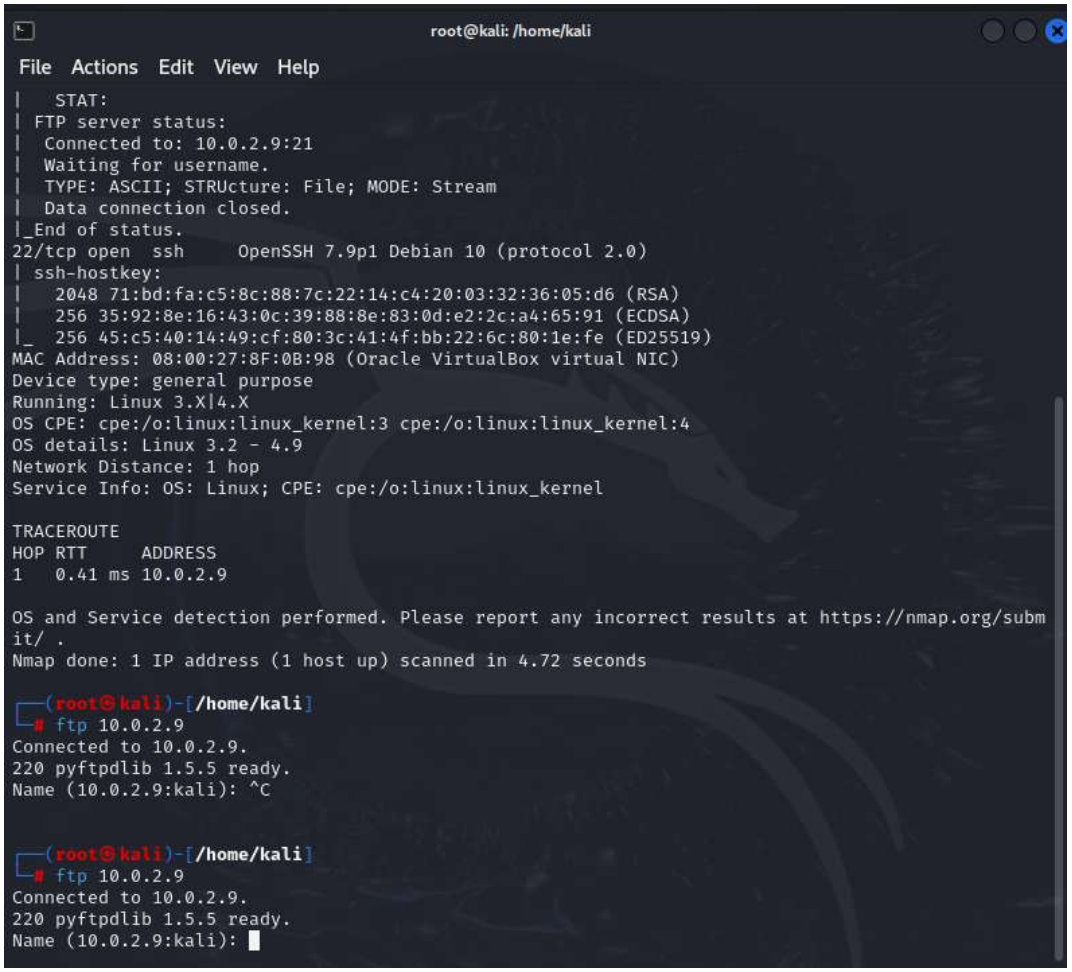
STEP-3.4: once in root user run `netdiscover` to get the ip address of the DC-1 server



STEP-3.5: Once ip is gained, run `nmap -A -p- <target_ip>` to get the open ports in the particular server.



STEP-3.6: Run command `ftp<target_ip>` to gain insights.

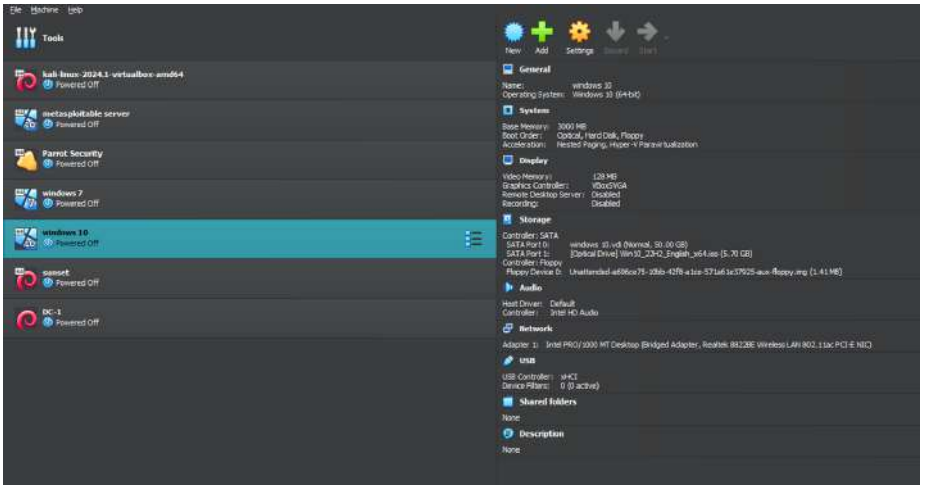


Conclusion:

Both Sunset and DC-1 server are explotable server where linux can be easily used wtih nmap scripts to gain information.

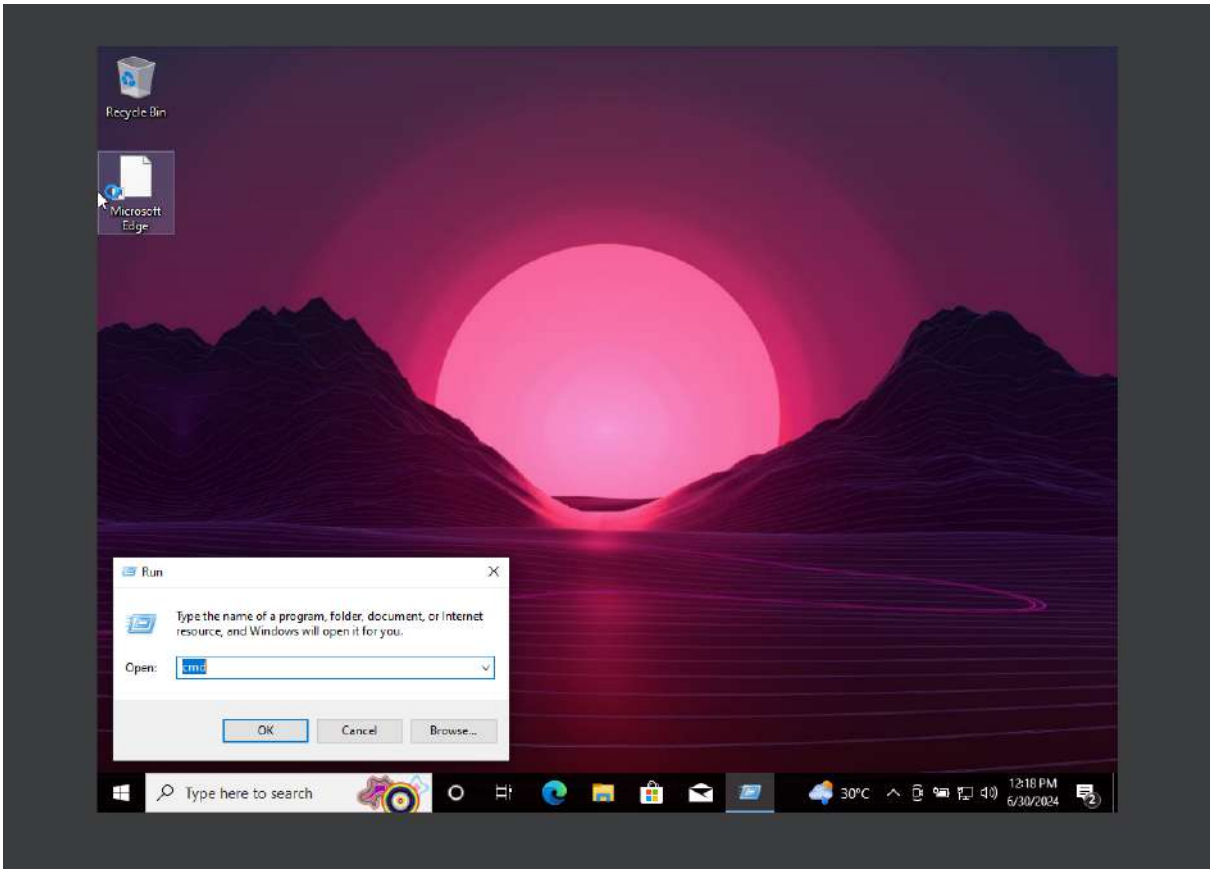
Objective-03: Perform a DOS attack on the windows 10 virtual machine and check the performance.

STEP-1: Open the virtual machine box and launch windows 10 machine



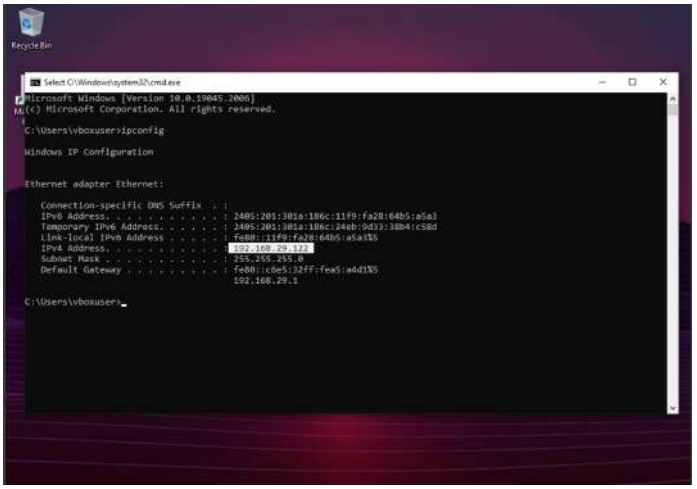
To launch the machine click on start.

STEP-2: Open command prompt using run window



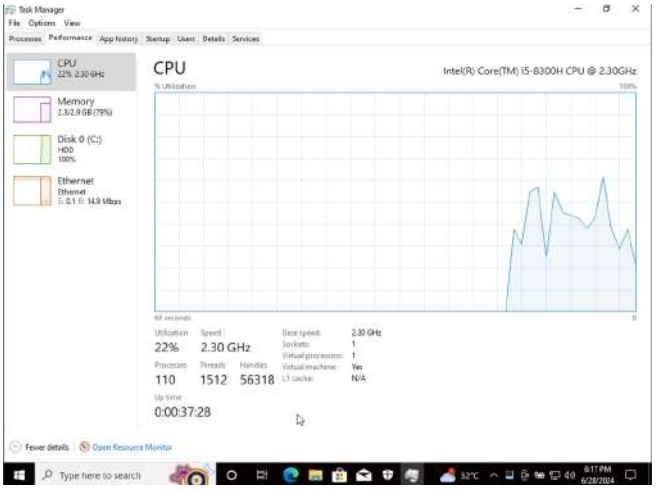
type “cmd” in the search box click on ok.

STEP-3: Now type “ipconfig” in the command prompt to gain IP address of the windows operating system.



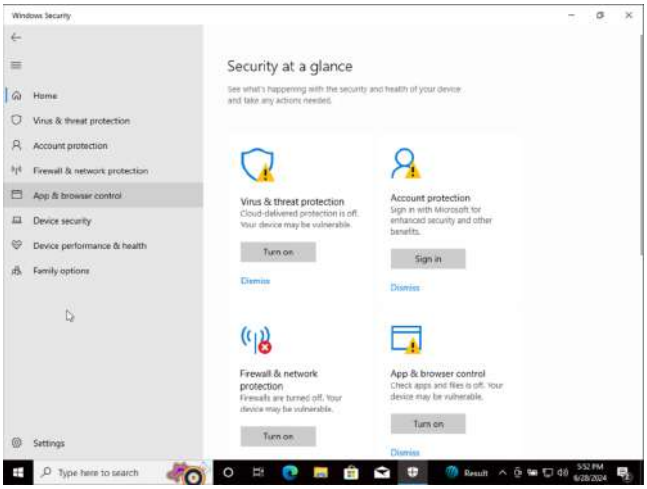
note the IPv4 address.

STEP-4: Now check the performance in the task manager and note it down.

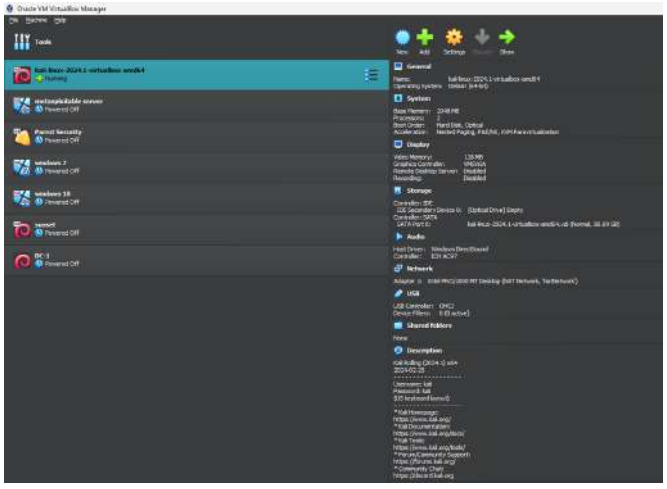


STEP-5: Disable all the security parameters of the windows 10 VM. This includes the following settings:-

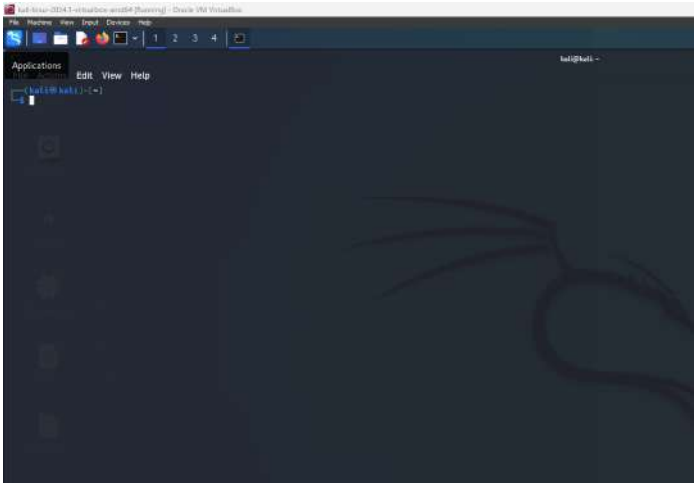
- Virus and threat protection.
- Account protection
- Firewall and network protection.
- App & browser control.



STEP-6: launch Kali-linux virtual machine in VM Box.

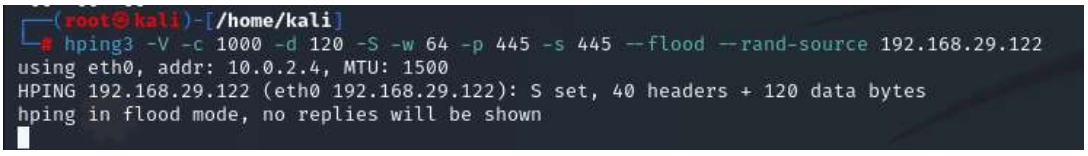


STEP-7: Open new terminal in the linux machine.



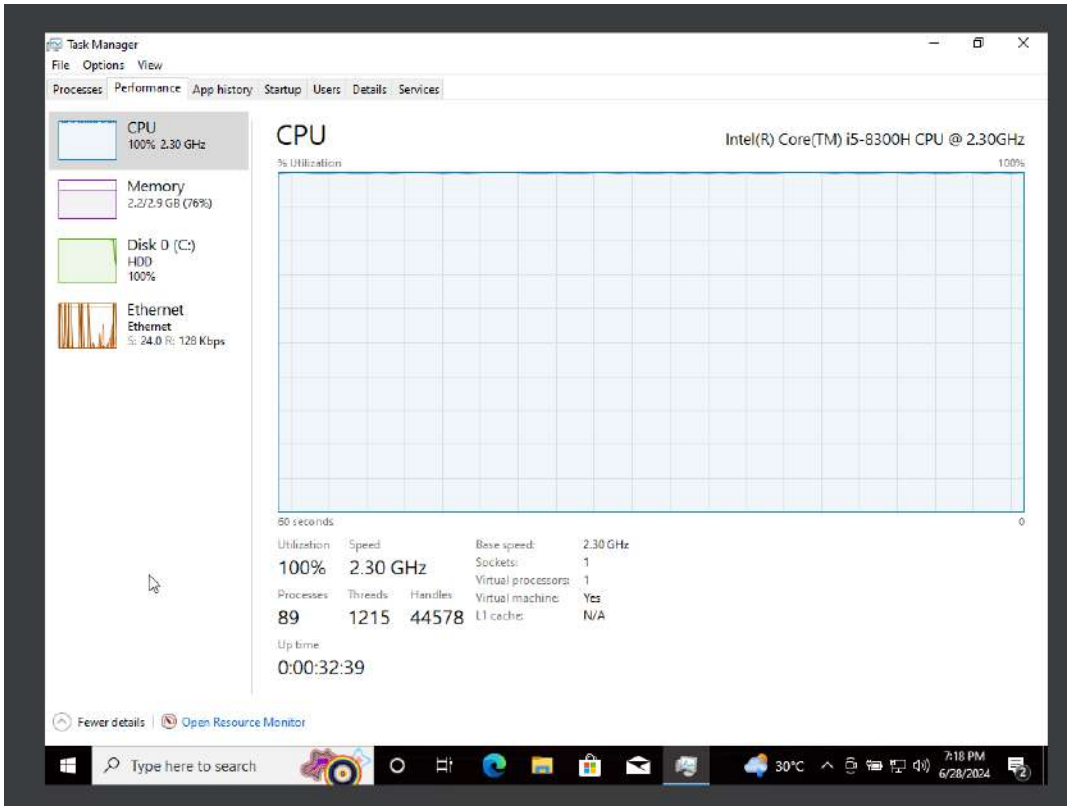
STEP-8: Enter root user by using command `sudo su`

STEP-9: Now to perform DOS attack run the following command `hping3 -V -c 1000 -d 120 -S -w 64 -p 445 -s 445 –flood –rand-source <target_IP>`



This command will start sending flooding the windows virtual machine.

STEP-10: Open windows 10(again) and check the performance using task manager.



As we can see as soon as the Dos attack is started the utilization of all resource in the windows 10 VM clocked to 100% making the machine unusable until attack is stopped.

Conclusion:

- A Denial of Service (DoS) attack aims to leave a machine or network unavailable to its users by overwhelming it with a flood of illegitimate request. This is occurred when the security aspects of any machines are disabled.
- Linux is a powerful tool tool to perform Dos attack on virtual machine.
- To do DOS attack on a website LOIC(low orbit ion cannon) can be used.

Summary-

The report covers three main objectives.

Firstly, it demonstrates how to use **Wireshark**, a network protocol analyzer, to sniff data from a website capturing FTP, HTTP, and POP protocols.

Secondly, it explains how to exploit the Sunset and DC-1 servers using Linux and Nmap scripts.

Lastly, it provides a step-by-step guide on performing a **Denial of Service (DoS)** attack on a Windows 10 virtual machine using Linux, leading to 100% resource utilization and rendering the machine unusable.

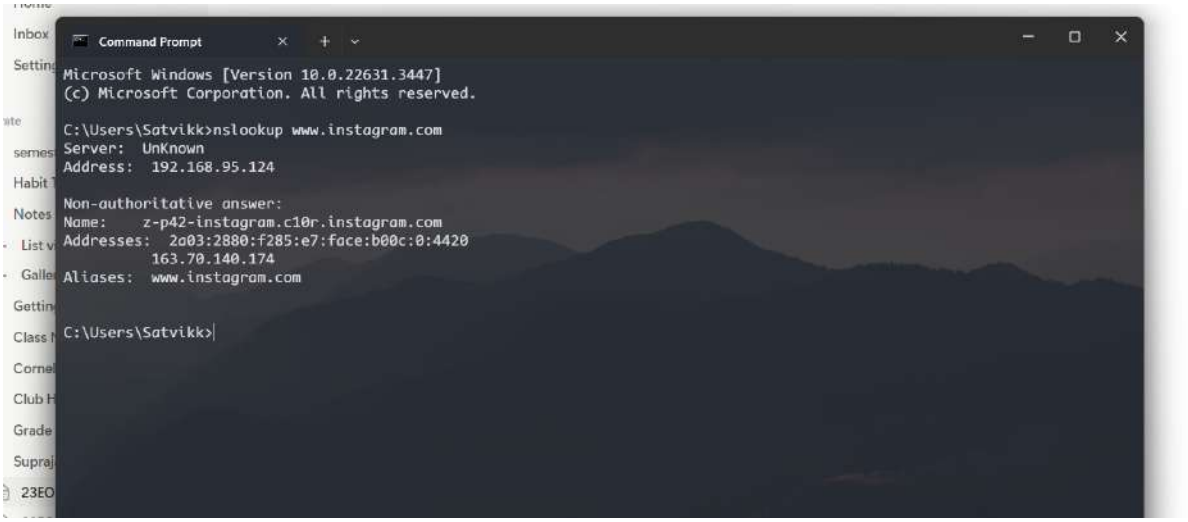
23EO5-ST#IS#6653-Task-5

Objective -

Changing Firewall Rules, DoS using Goldeneye and Wireshark, Backdoor using Metaspoilt framework

A) Turn off the antivirus and block the Instagram web application and a Standalone application by changing the rules of the firewall.

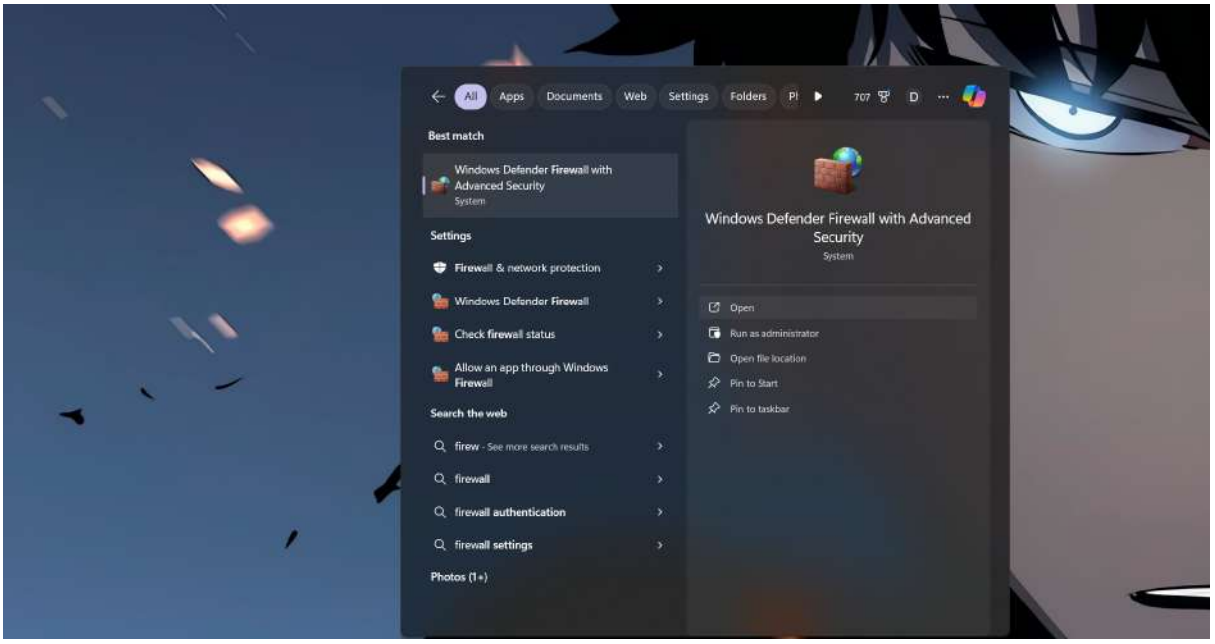
STEP 1 - Find the IP address of Instagram using any tool. Here we have used nslookup.



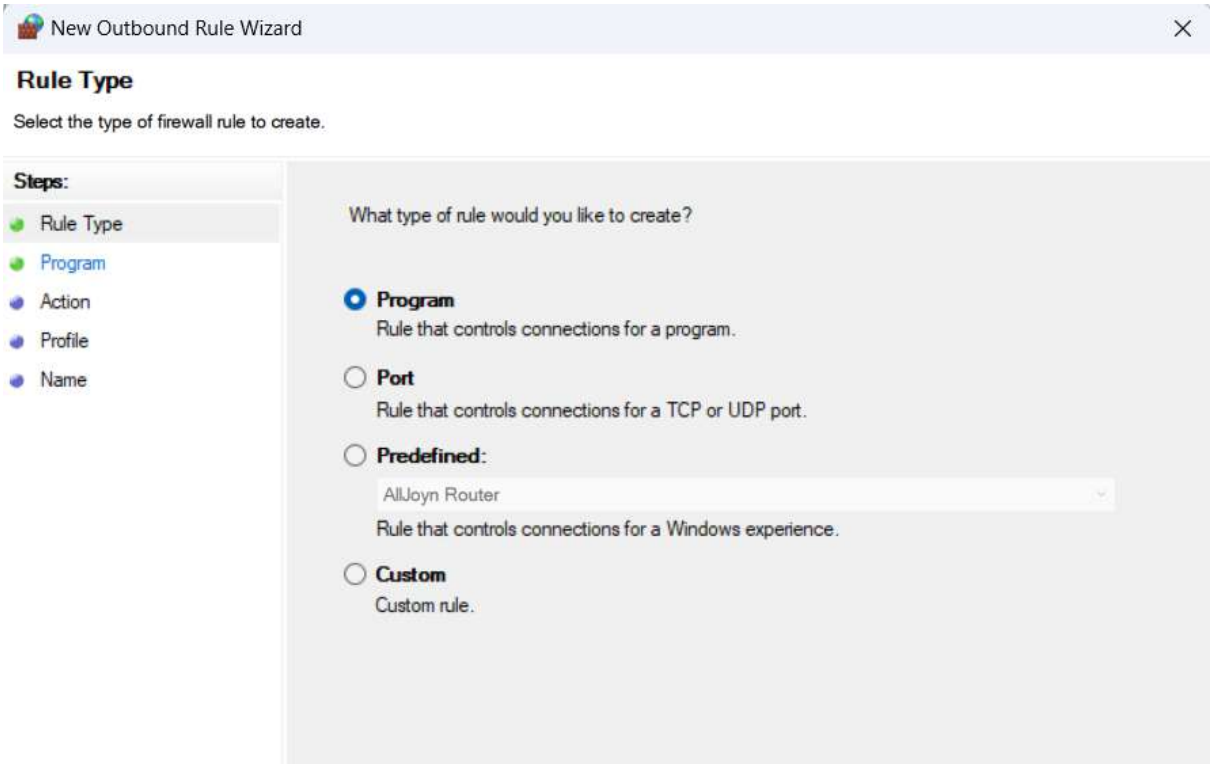
We found that the ip address instagram site uses is

2a03:2880:f285:e7:face:b00c:0:4420
163.70.140.174

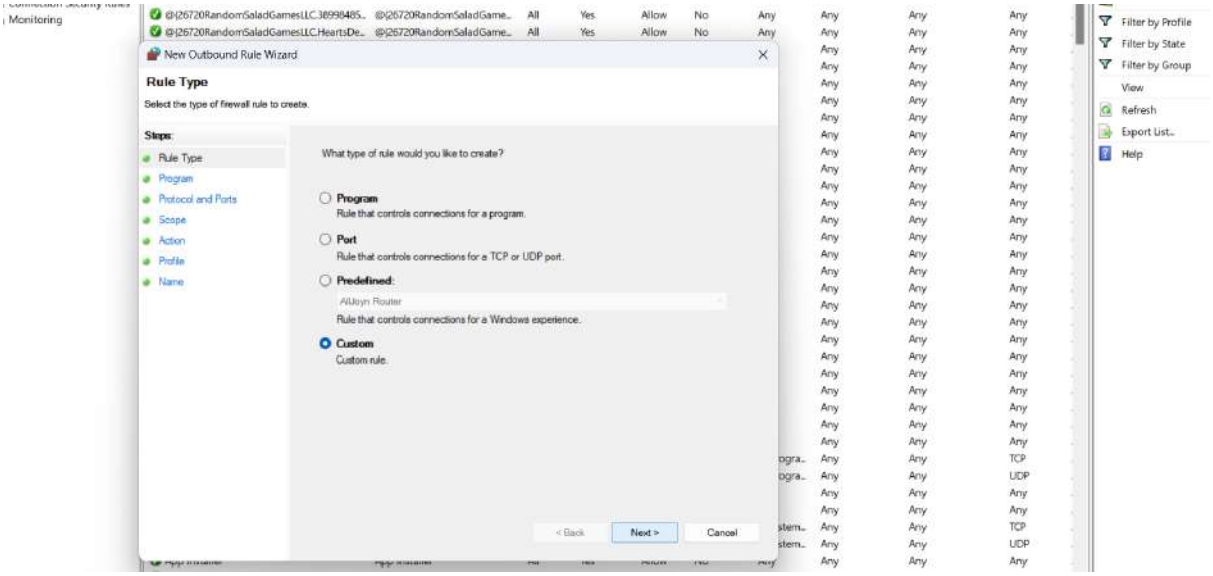
STEP 2 - Open Firewall on your Windows Machine



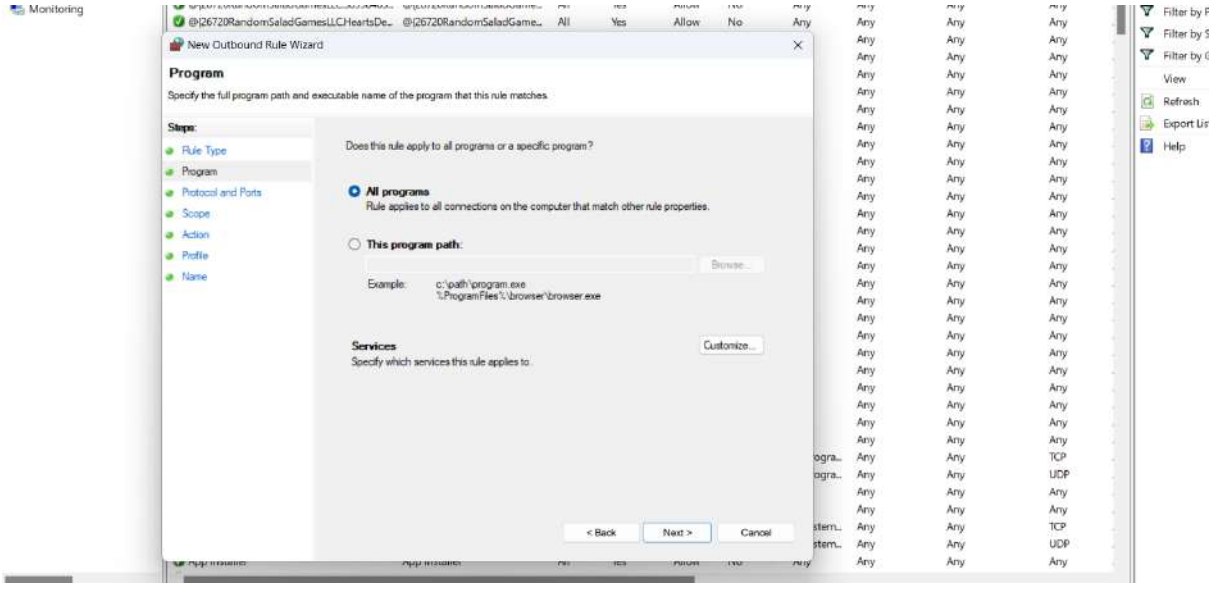
STEP 3 - Under Outbound Rules Section Proceed to add New Rule...



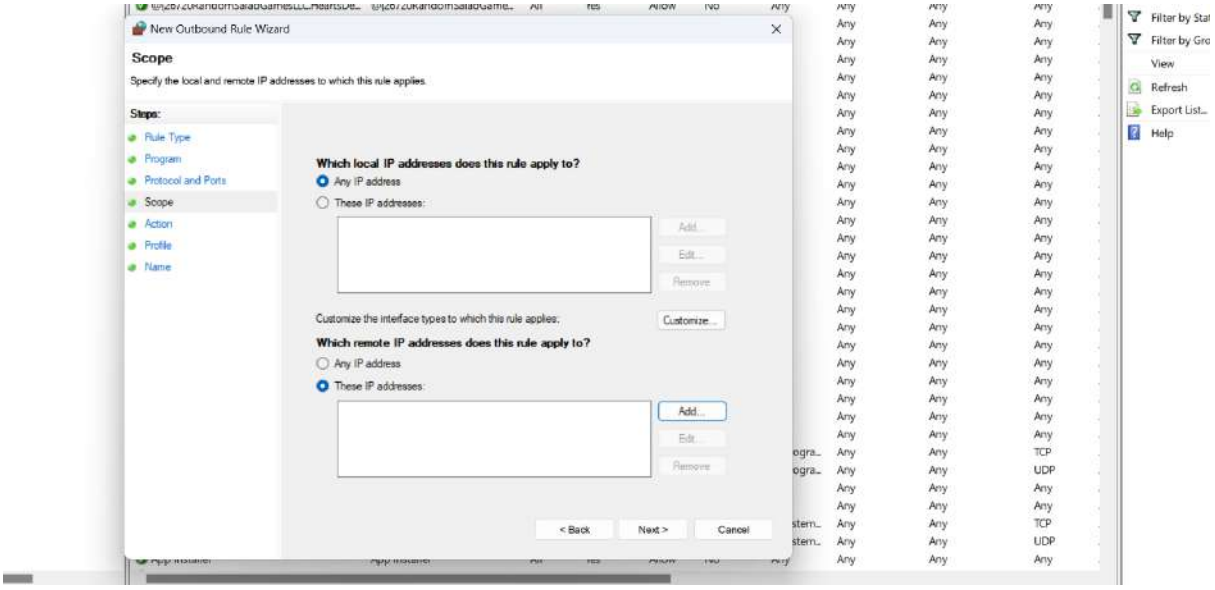
STEP 4 - Select Custom Rule and Click Next



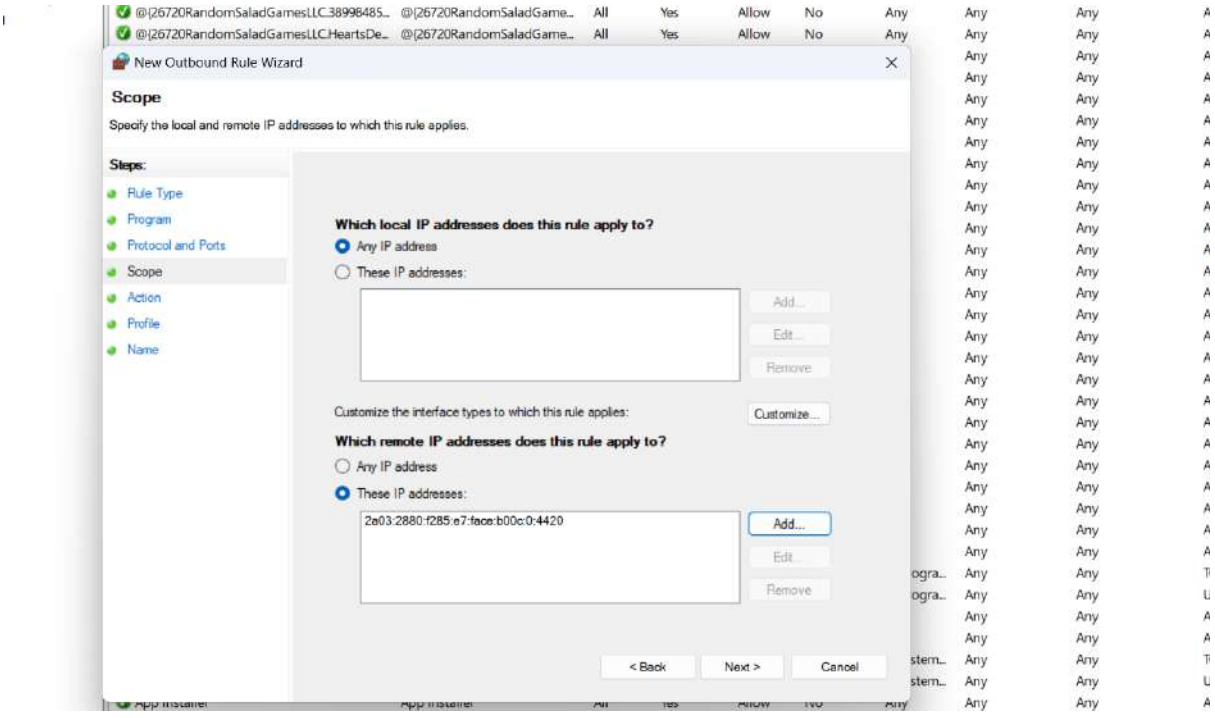
STEP 5 - Select all programs and click next.



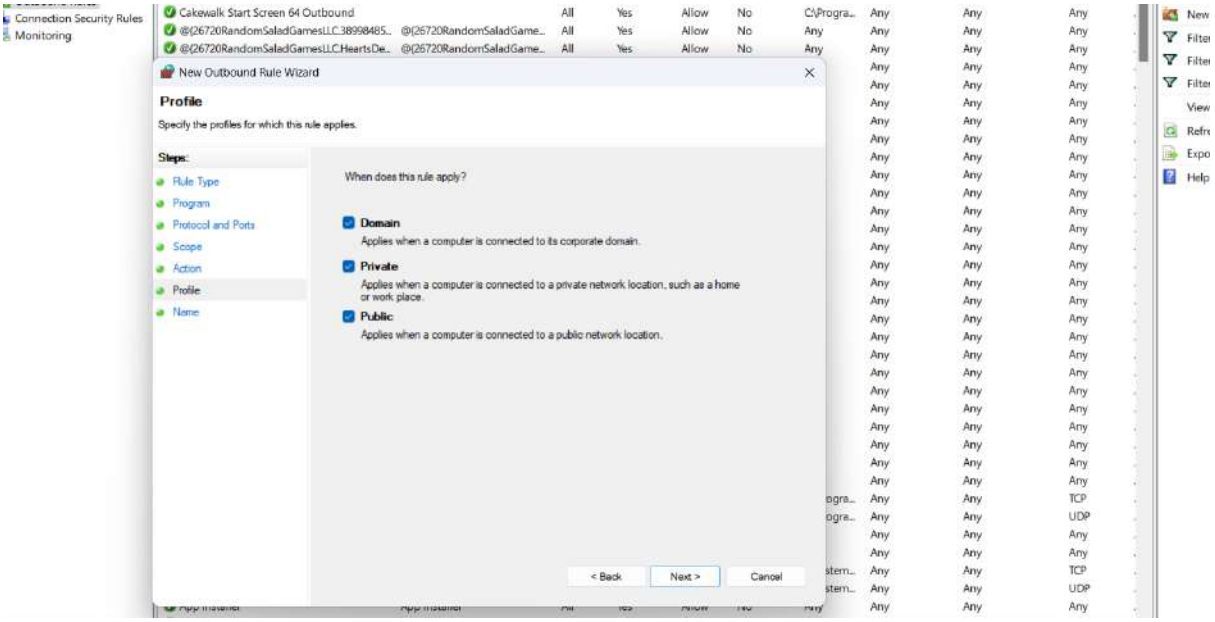
STEP 6 - In Scope section, proceed to add the IP address that we found in the Add IP field.



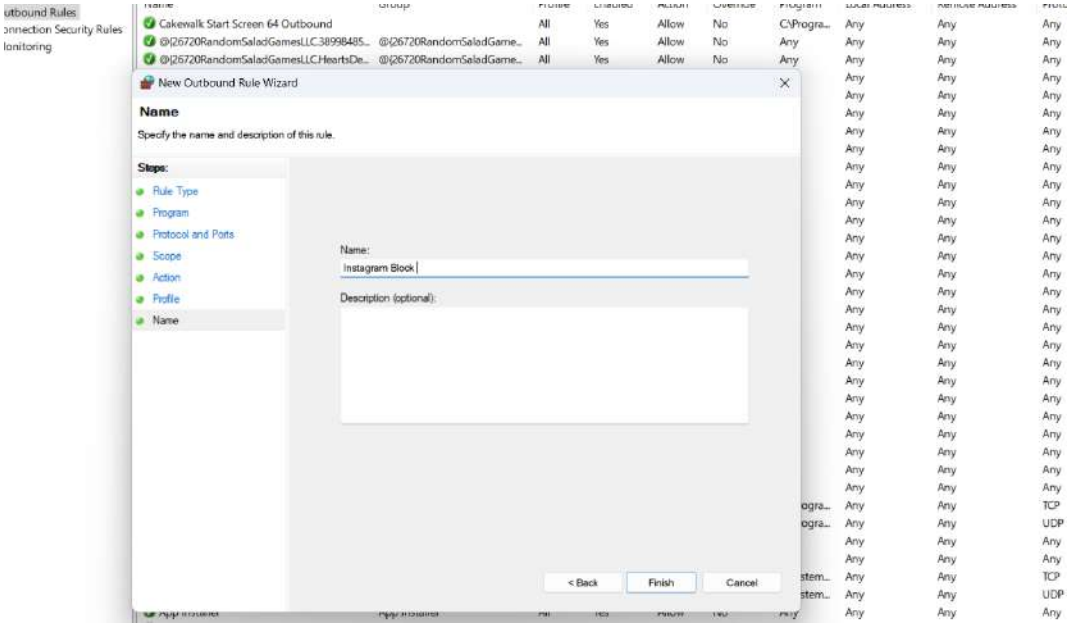
STEP 7 - Add IP addresses and Click Next.



STEP 8 - Leave all the other sections default.

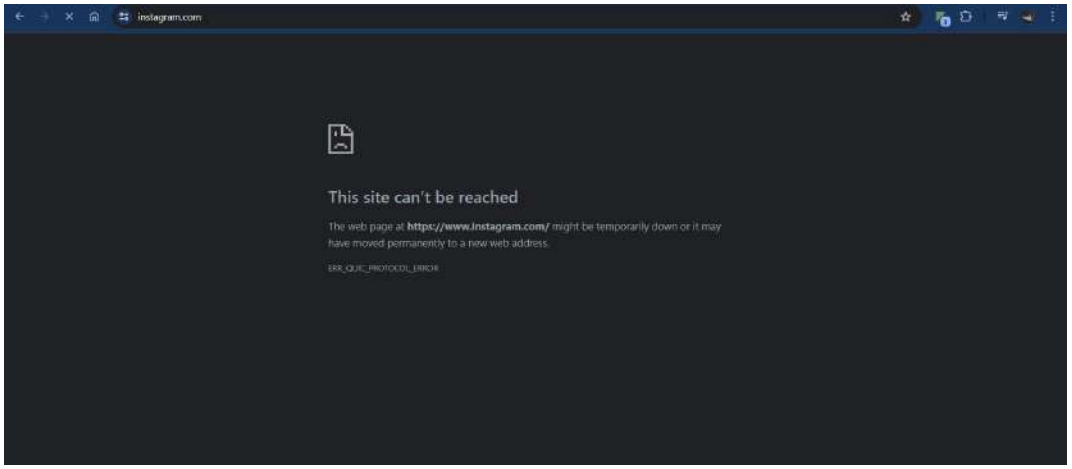


STEP 9 - Give a suitable name to the Rule and Click Finish



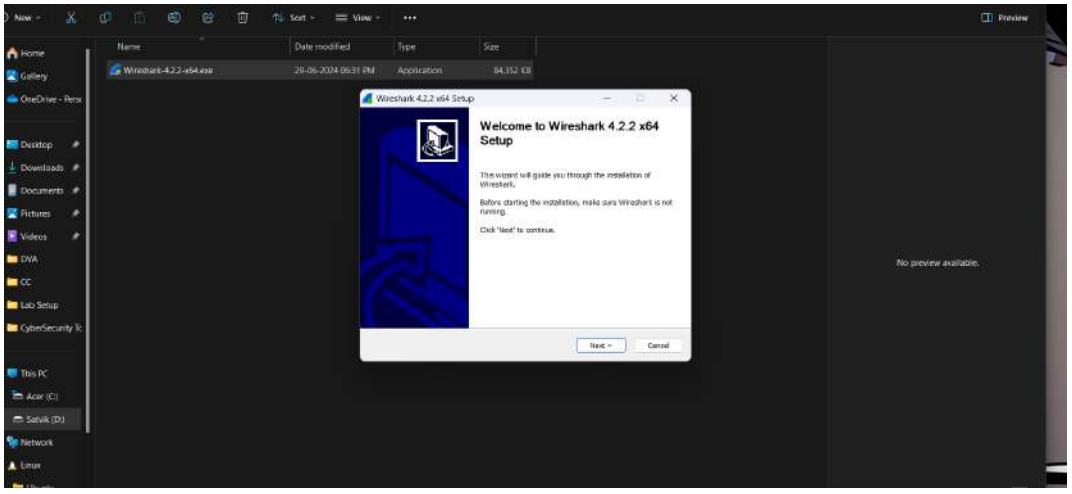
Conclusion

After performing the above steps you can see instagram.com is not accessible anymore.

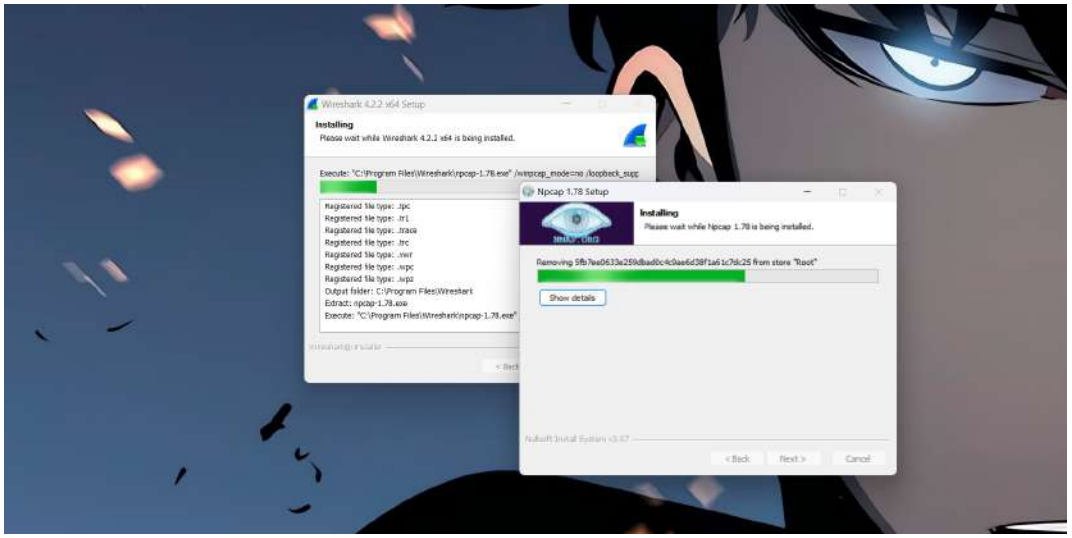


Task B) Perform Dos Attack using the goldeneye tool on any 2 non-Indian websites and Observe the traffic in the Wireshark.

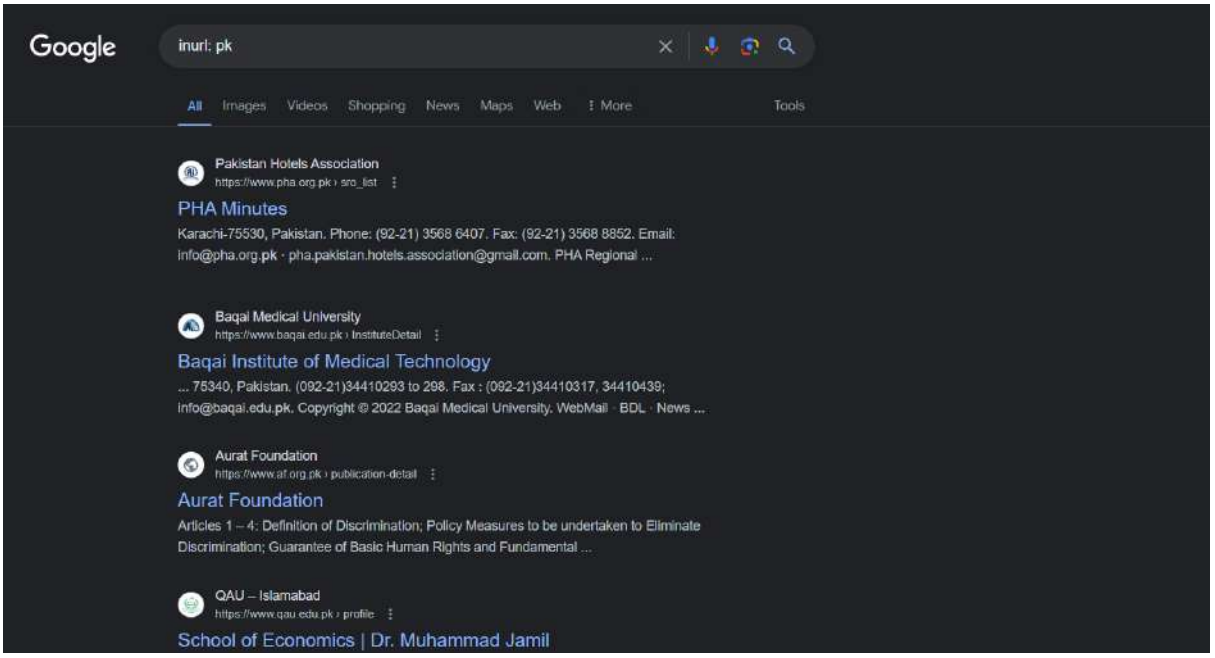
STEP 1 - Install WireShark tool in your machine by running wireshark.exe



STEP 2 - Wait for all the dependencies to install completely.

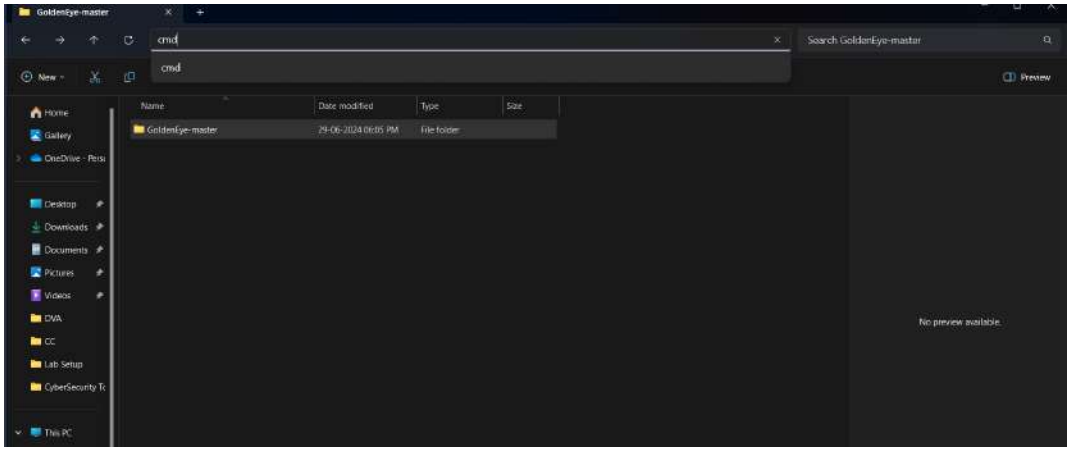


STEP 3 - Find Two Websites to attack using Google Dorks.

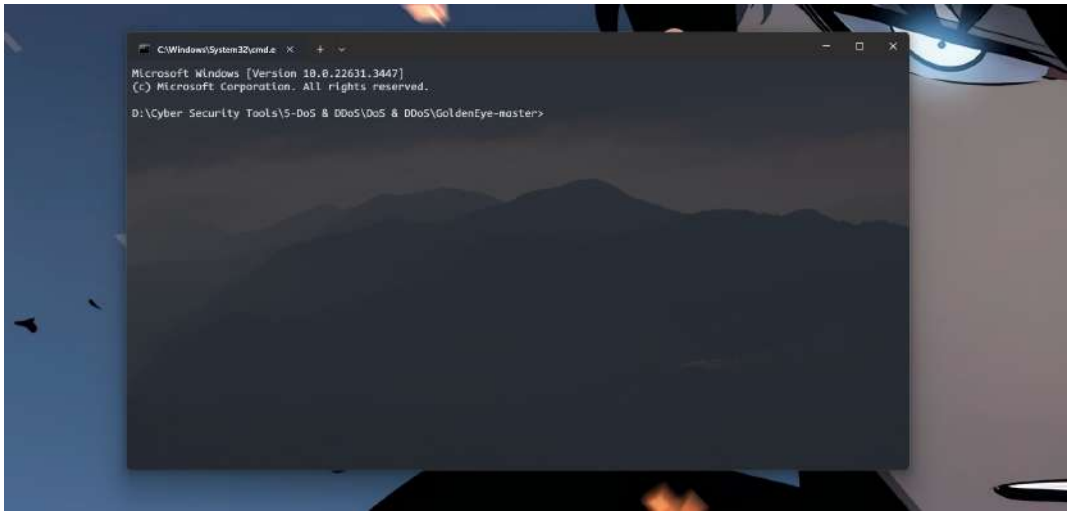


Here we are using szic.pk and panki.it .

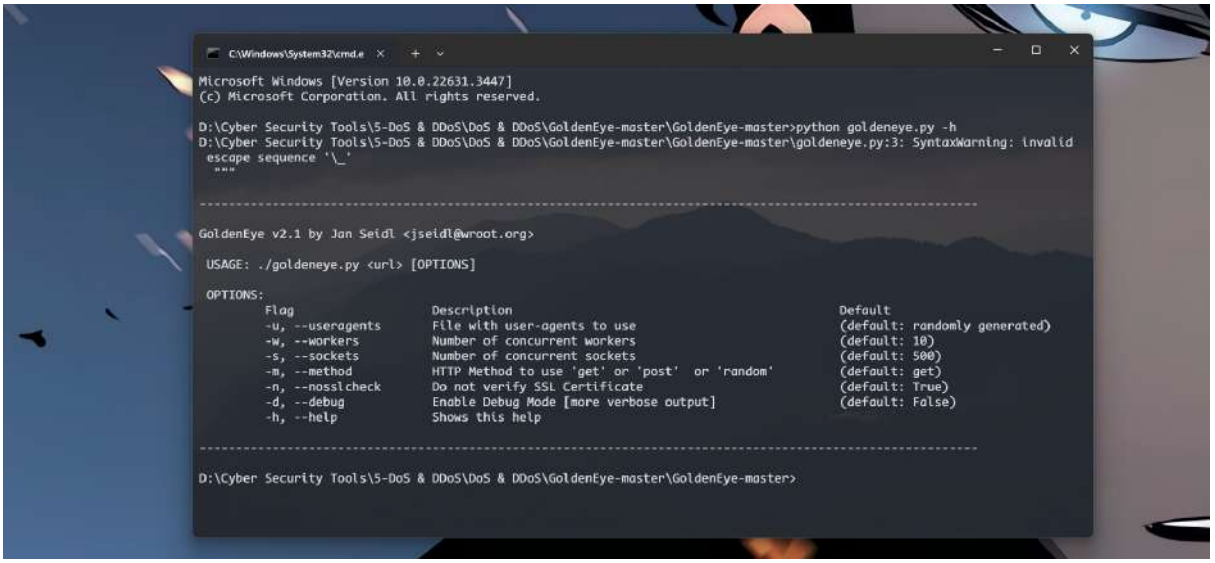
STEP 4 - Now open Golden Eye Tool in cmd (Command Prompt) .



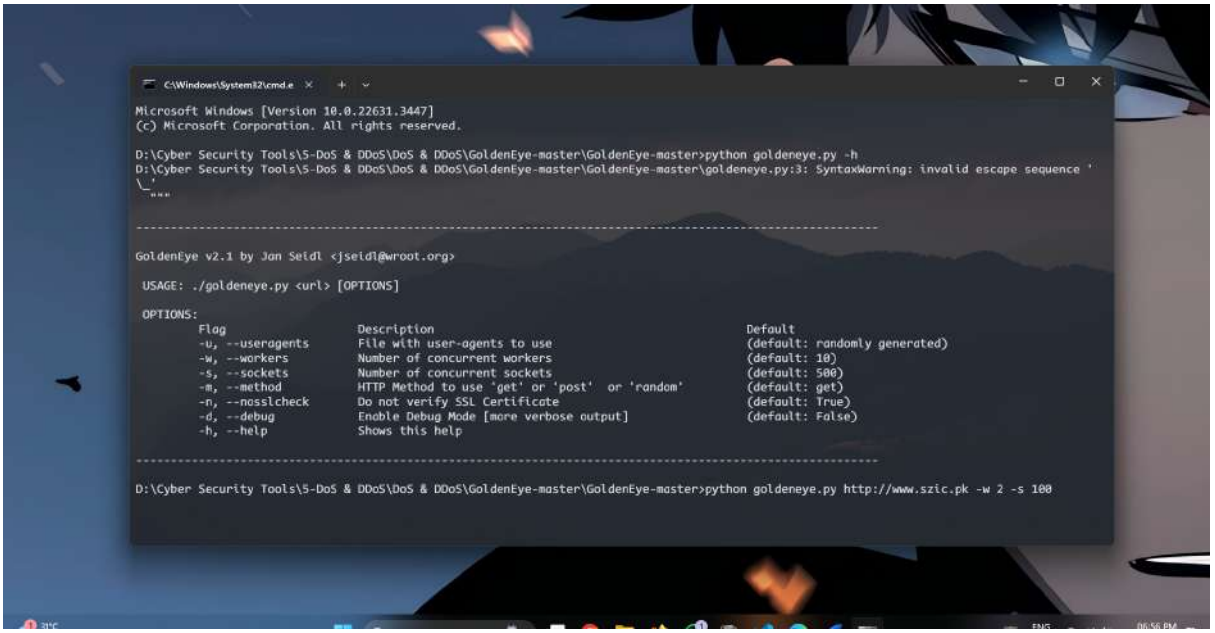
STEP 5 - The cmd will open on the specified path.



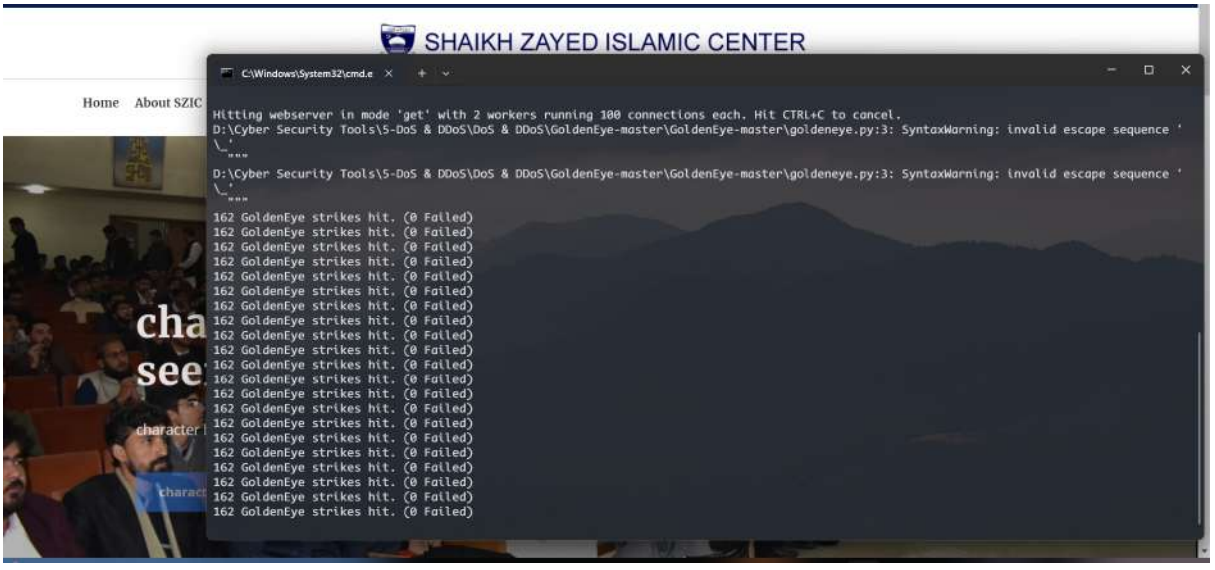
STEP 6 - Run command `python goldeneye.py -h` to see help list.



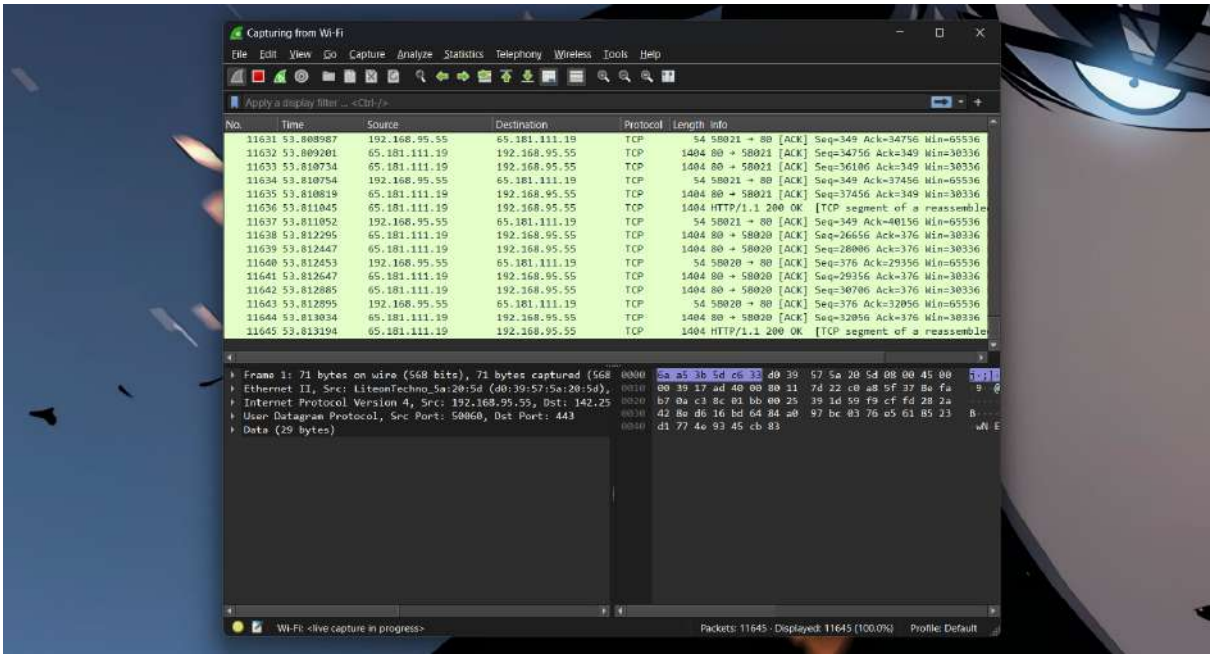
STEP 7 - Enter the command `python goldeneye.py http://www.szic.pk -w 2 -s 100` to attack on the first site.



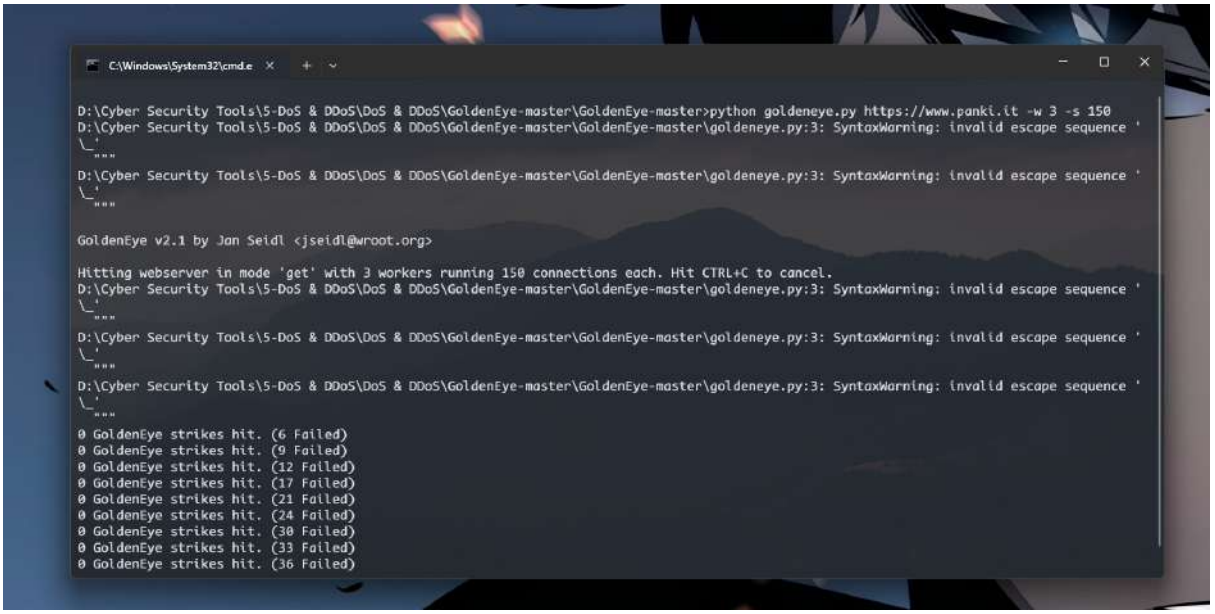
STEP 8 - Hit Enter to see packets hitting.



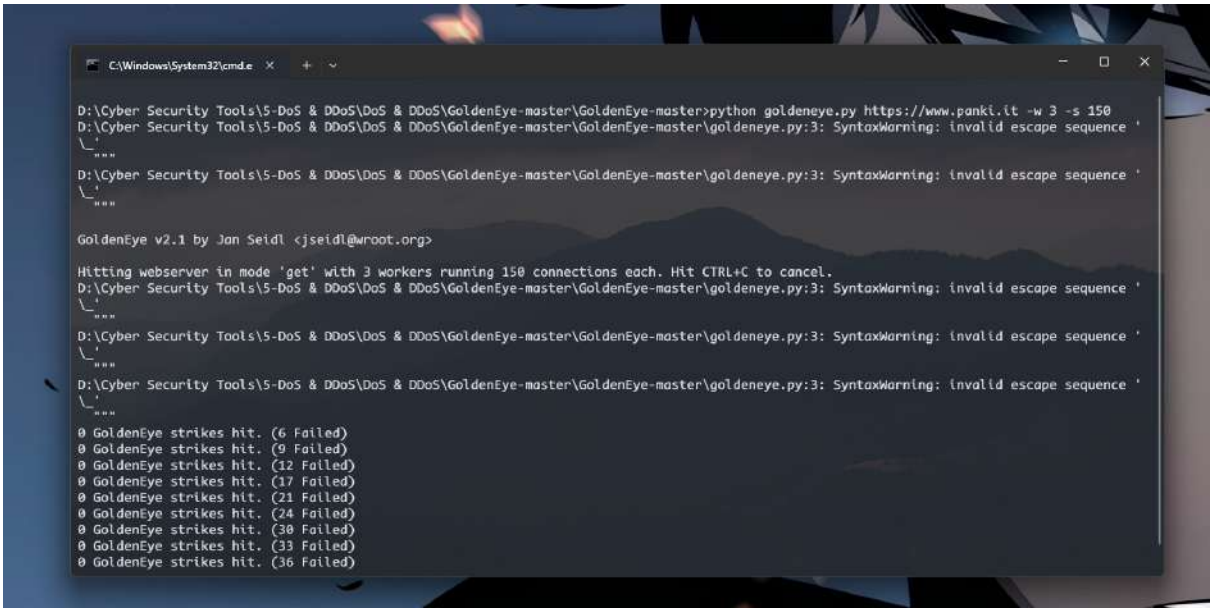
STEP 9 - Open WireShark Tool to observe traffic on the target site from your machine.



STEP 10 - Similarly attacking panki.it also adjusting the parameter values.



STEP 11 - Here is the output shown by WireShark.



Conclusion

Wireshark and GoldenEye are powerful tools used for network analysis and website testing. Wireshark is a network protocol analyzer that allows you to capture and interactively browse the traffic running on a computer network. It provides detailed information about network traffic such as source and destination IP addresses, packet size, and protocols used.

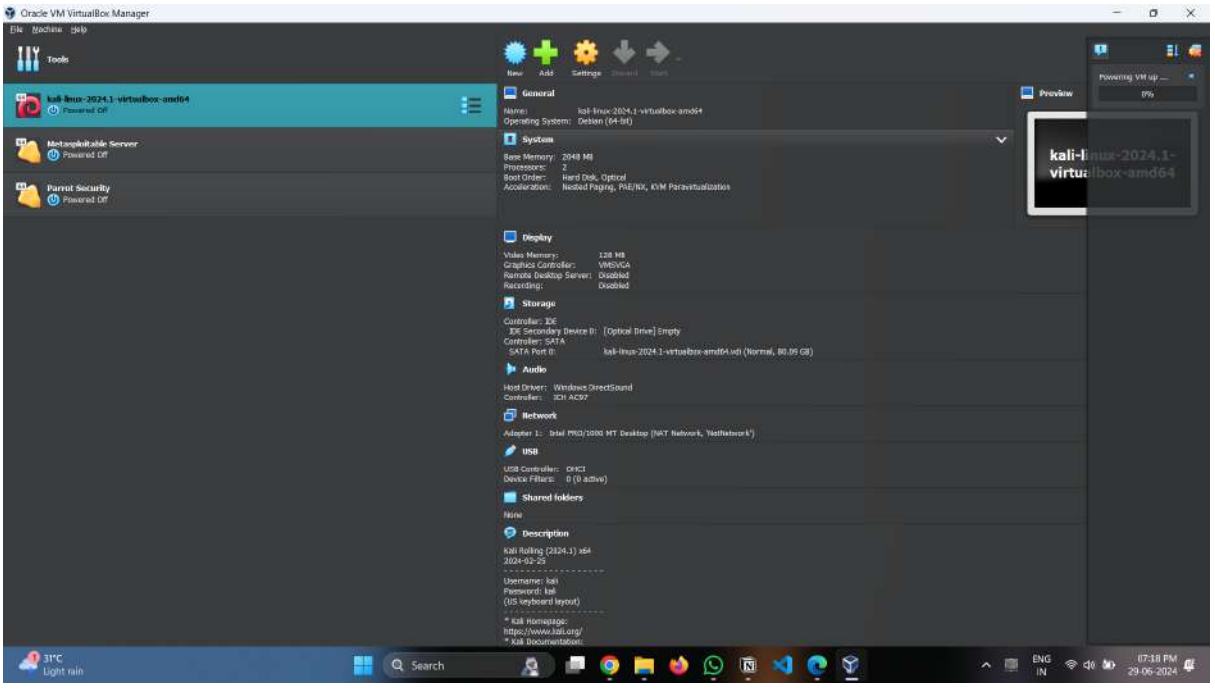
GoldenEye, on the other hand, is a tool used for load testing. It allows you to simulate DoS attacks on a network to test its robustness under high traffic conditions. It creates multiple processes and threads to maximize the stress on the target network or website.

In website testing, these tools can be used together to simulate high traffic conditions with GoldenEye and monitor the effects on the network using Wireshark. This can help identify potential vulnerabilities or performance issues in the website or network infrastructure.

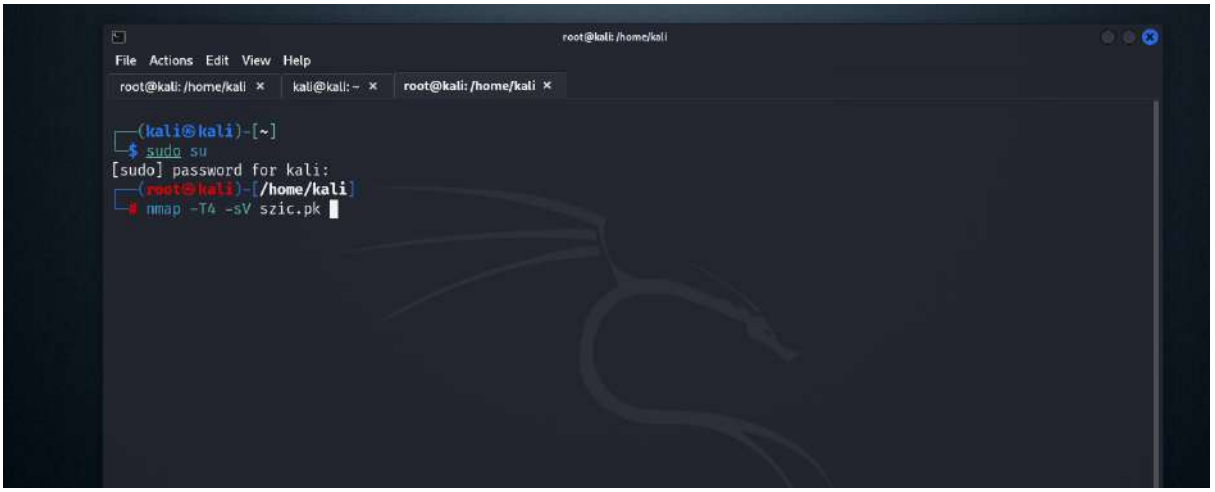
In this example we saw that DoS Attack effects the performance of the Websites.

Task C) Perform a Backdoor on a target website using the Metasploit tool.

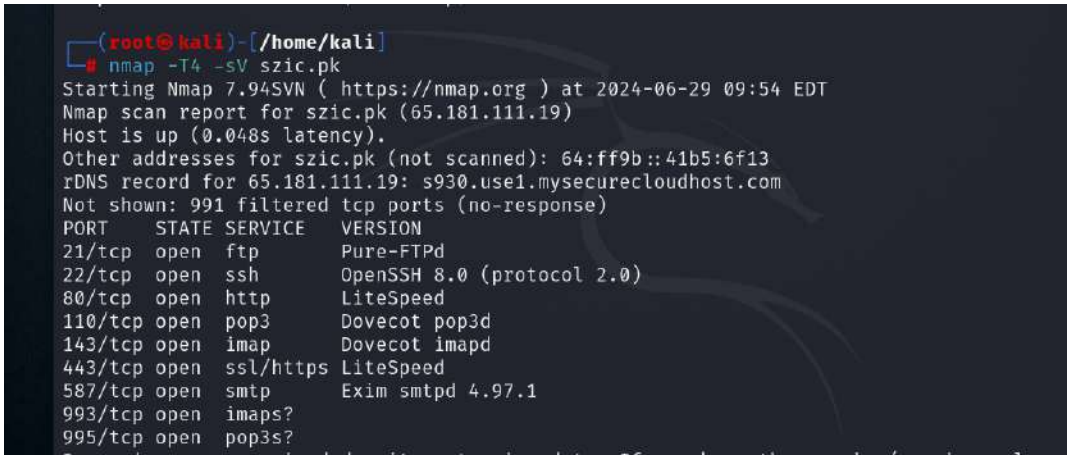
STEP 1 - Open your Kali Linux Terminal.



STEP 2 - Enter the Command `nmap -T4 -sV szic.pk`



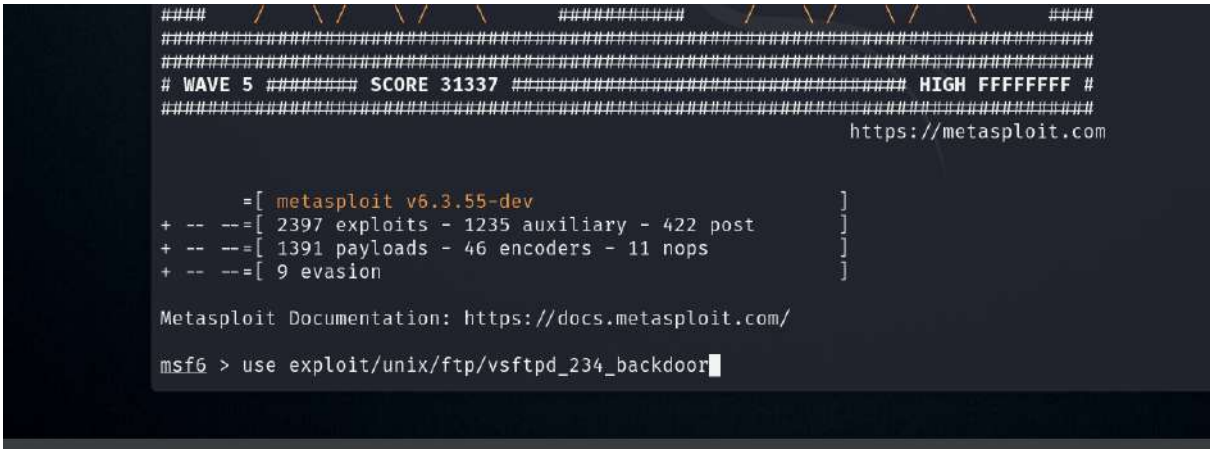
STEP 3 - All the open ports along with their Service Versions will appear on your screen.



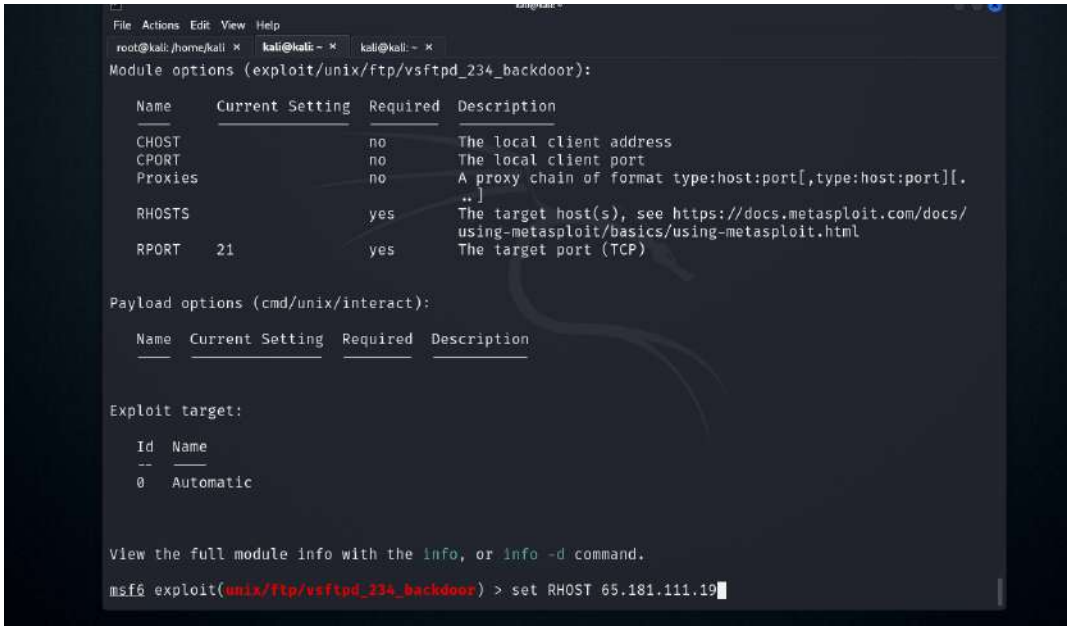
STEP 4 - Since we found the ftp vulnerability, type msfconsole to start the Metasploitable Framework.

STEP 5 - Search for the Service Vulnerability using searchsploit sV

STEP 6 - Use the Exploit `use exploit/unix/ftp/vsftpd_234_backdoor`.

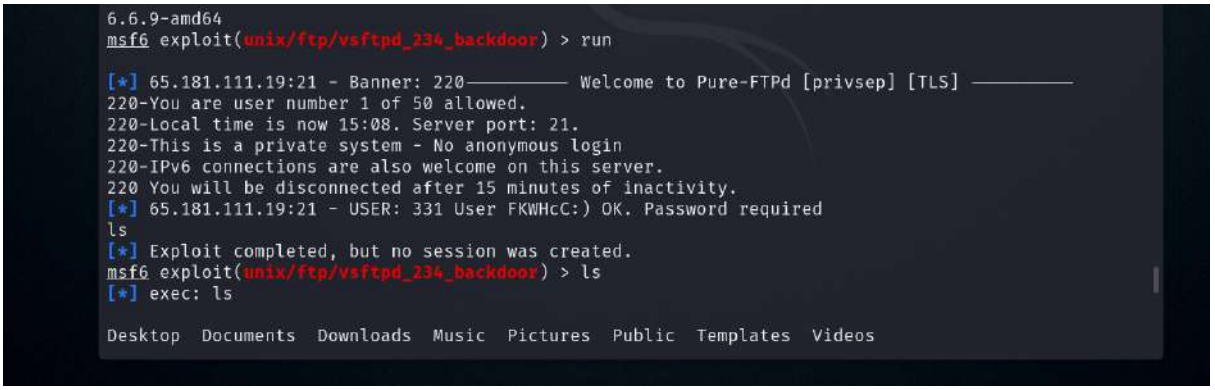


STEP 7 - Set RHOSTS value to the ip address of the target website.



STEP 8 - Finally run the exploit.

STEP 9 - A BackDoor has been successfully Created and we can see the files in the system using ls command.



Summary-

This report provides a detailed walkthrough of three key tasks related to network security and penetration testing.

The first task maps out the process of **blocking web applications** - using Instagram as an example - by altering firewall rules. It provides a step-by-step guide of finding the IP address of the target website and creating a new rule in the Windows firewall to block it.

The second task demonstrates a **Denial of Service (DoS)** attack on two non-Indian websites using **GoldenEye**, a load testing tool used for simulating DoS attacks. The traffic during the attack is monitored using Wireshark, a network protocol analyzer. This task aims to test the robustness of the target websites under high traffic conditions.

The final task involves creating a **backdoor** on a target website using the **Metasploit framework**. This part of the document guides through the process of identifying open ports and service vulnerabilities on the target website, and leveraging them to create a backdoor.

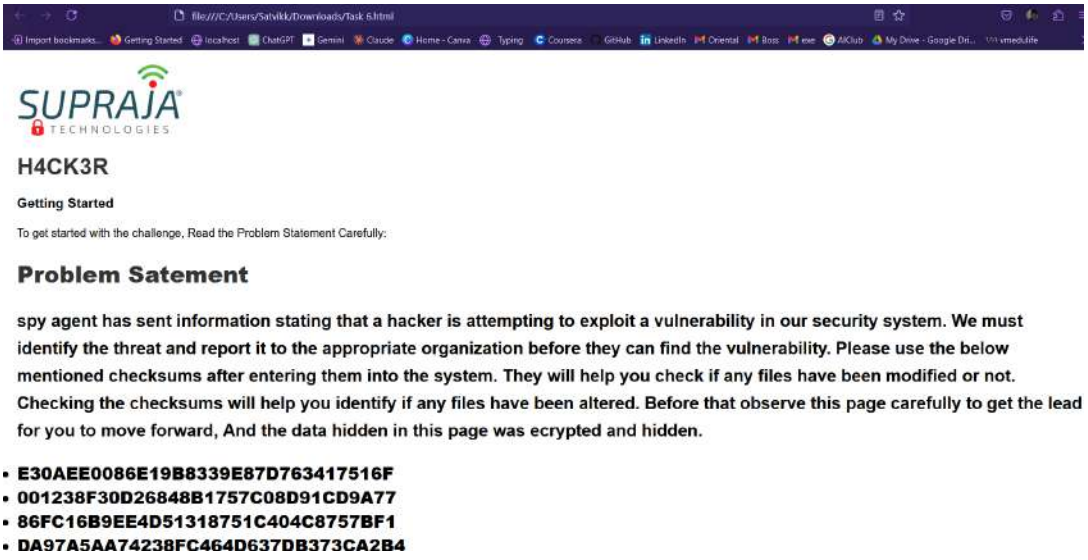
23EO5-ST#IS#6653-Task-6

Objective - Find the Flag {*****} that is in the Vulnerable System

A. Identify the hidden message in the README/HTML file

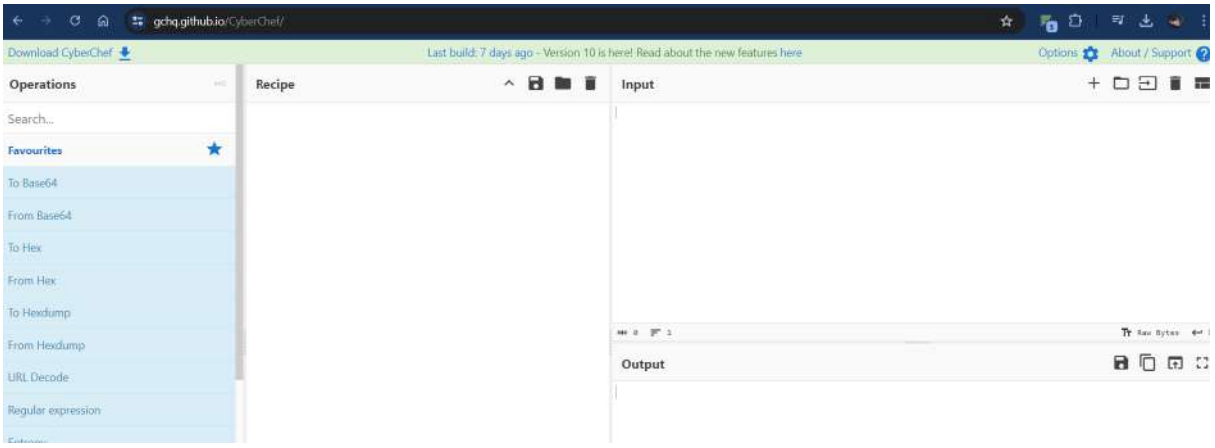
STEP 1. Decrypt the Secret Data to get a link

1.1 Open the html file in any browser.

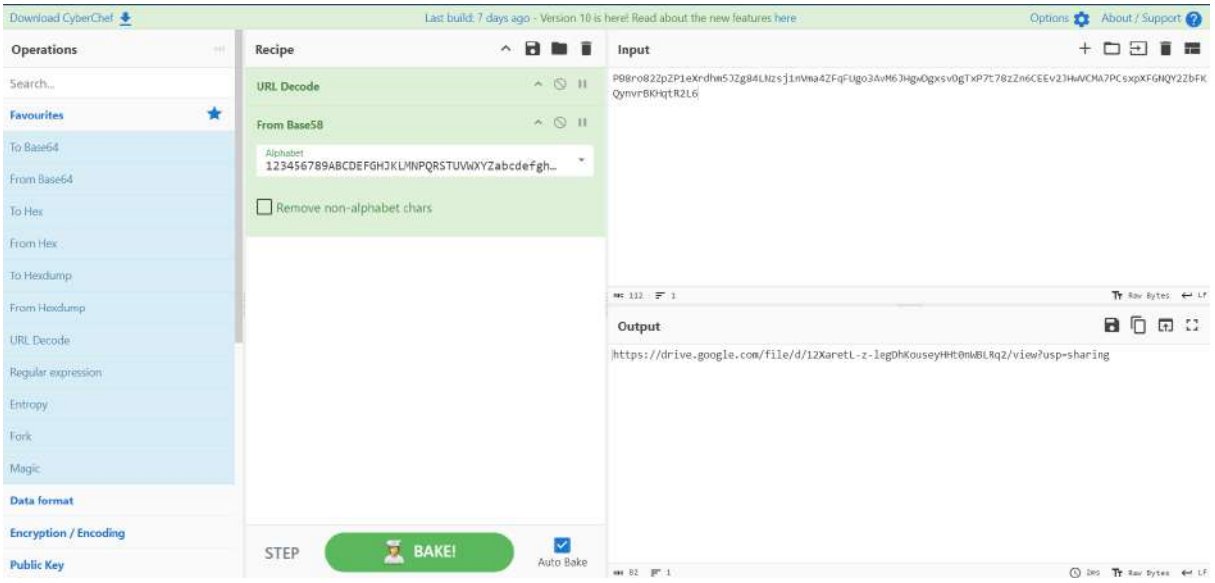


1.2 Look for the hidden message (Drag the element to reveal the Encrypted Text)

1.3 Copy the text and head to CyberChef website - https://gchq.github.io/CyberChef/



1.4 Now Paste the Link in the Input Section and Choose the Operation to URL Decode and From Base 58 then click on Bake! -

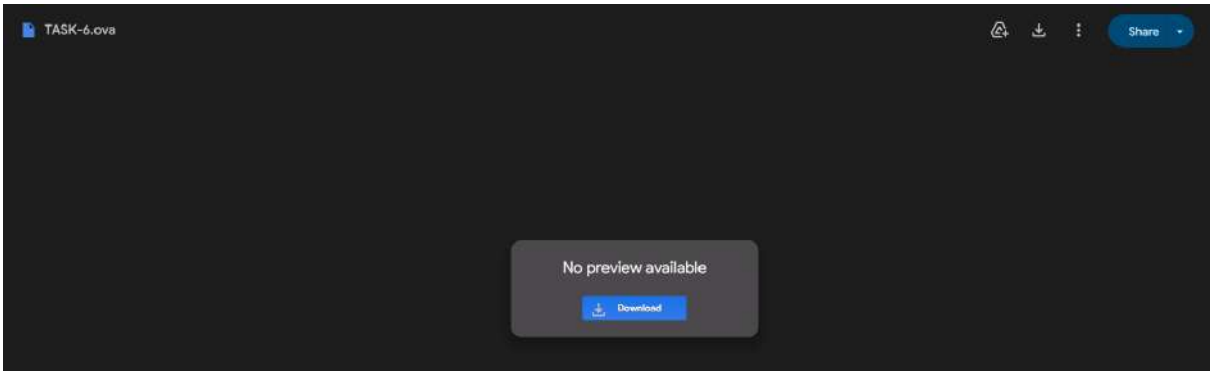


1.5 The Decrypted Text will appear on the Screen!

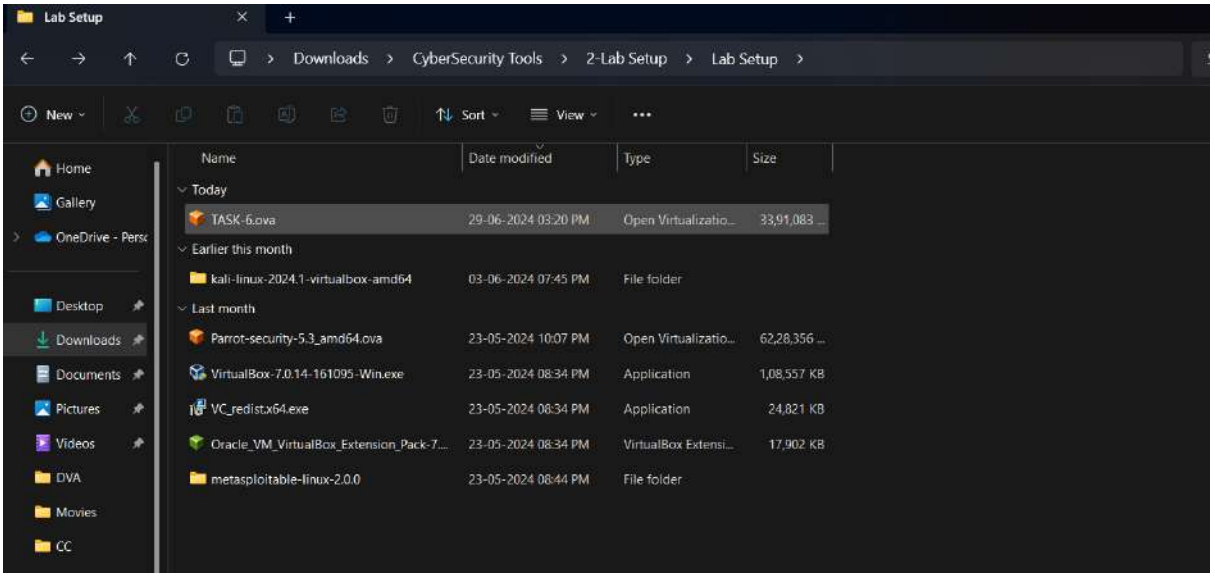
The Link is https://drive.google.com/file/d/12XaretL-z-legDhKouseyHHt0nWBLRq2/view?usp=sharing

STEP 2 Download and Import the OVA file

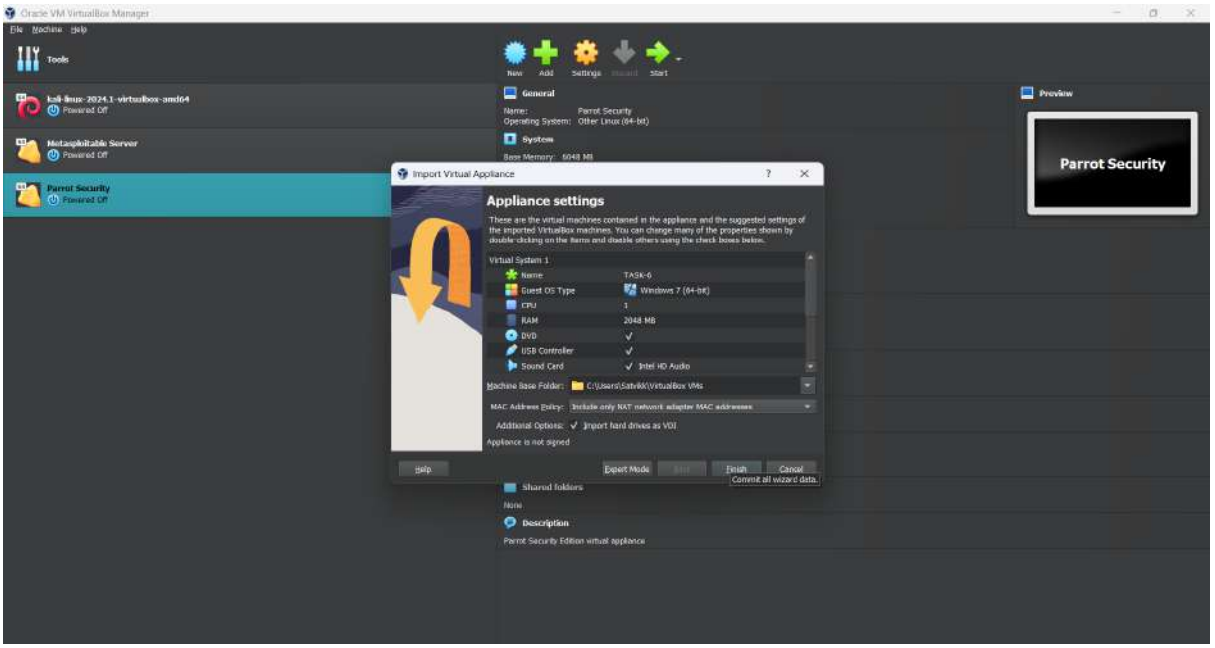
2.1 Get the OVA file Downloaded



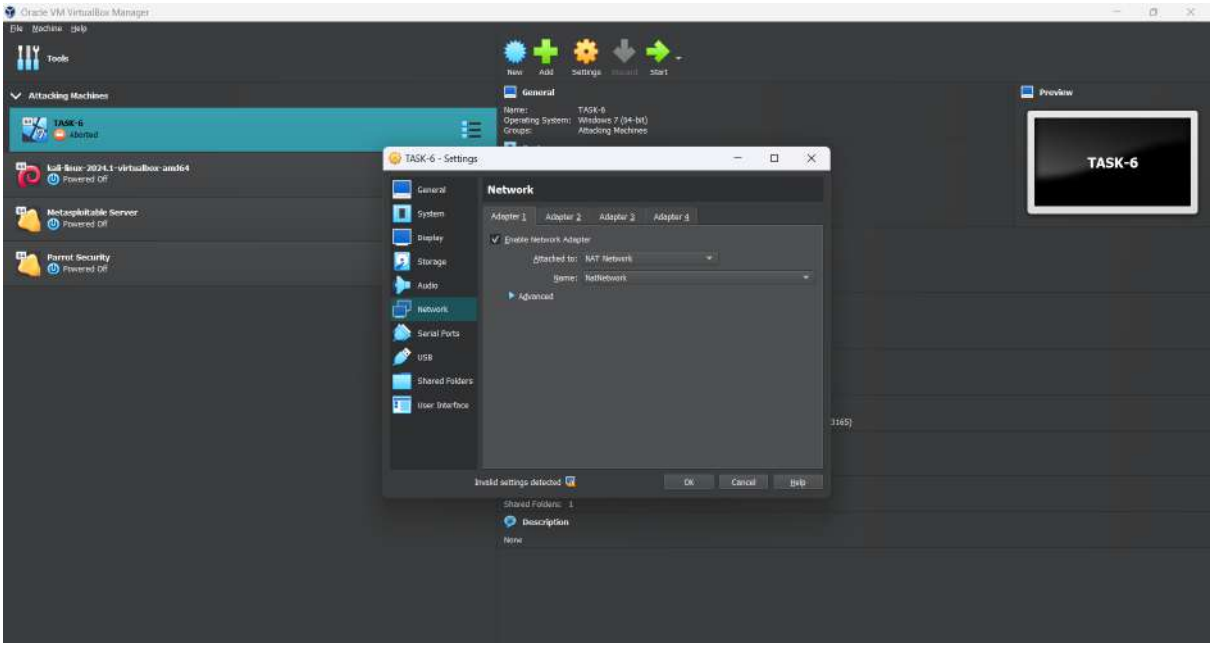
2.2 Open the file in VirtualBox.



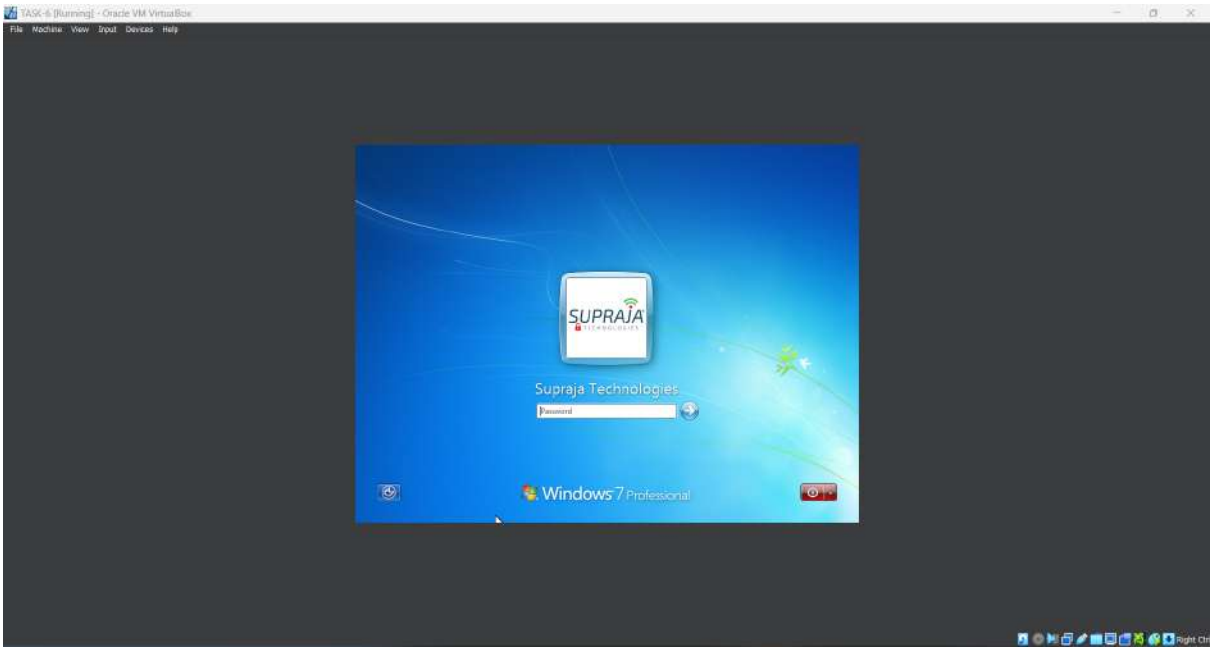
2.3 Proceed the Deployment and wait for the appliance to import completely.



2.4 After importing check if the Network Setting is set to NAT Network



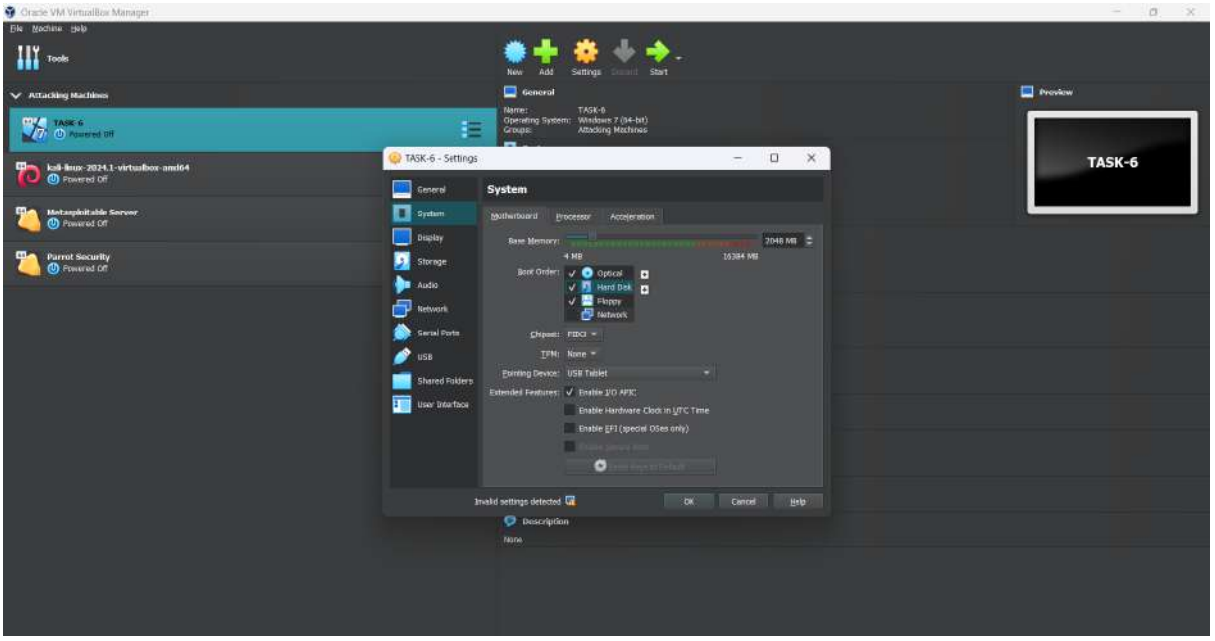
2.5 As you can see upon opening the Windows 7 System it asks for a password, we will use **OPH Crack tool** to crack its password.



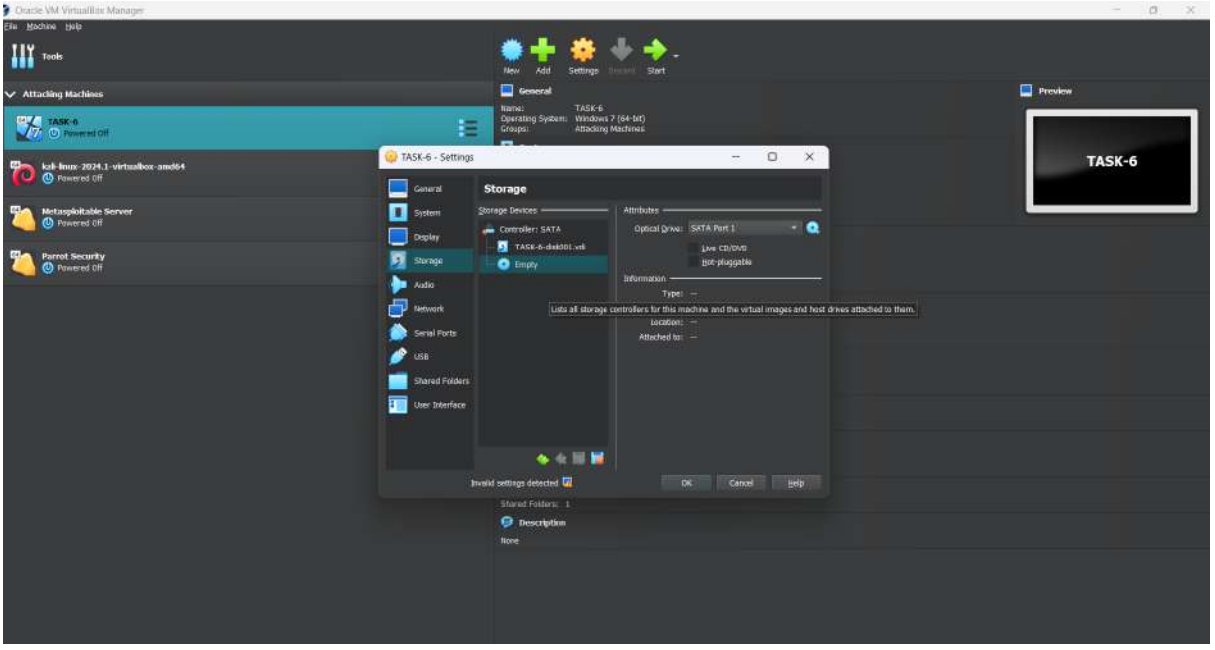
B. Gaining Access

STEP 1 Crack the system password using OPH Crack Tool

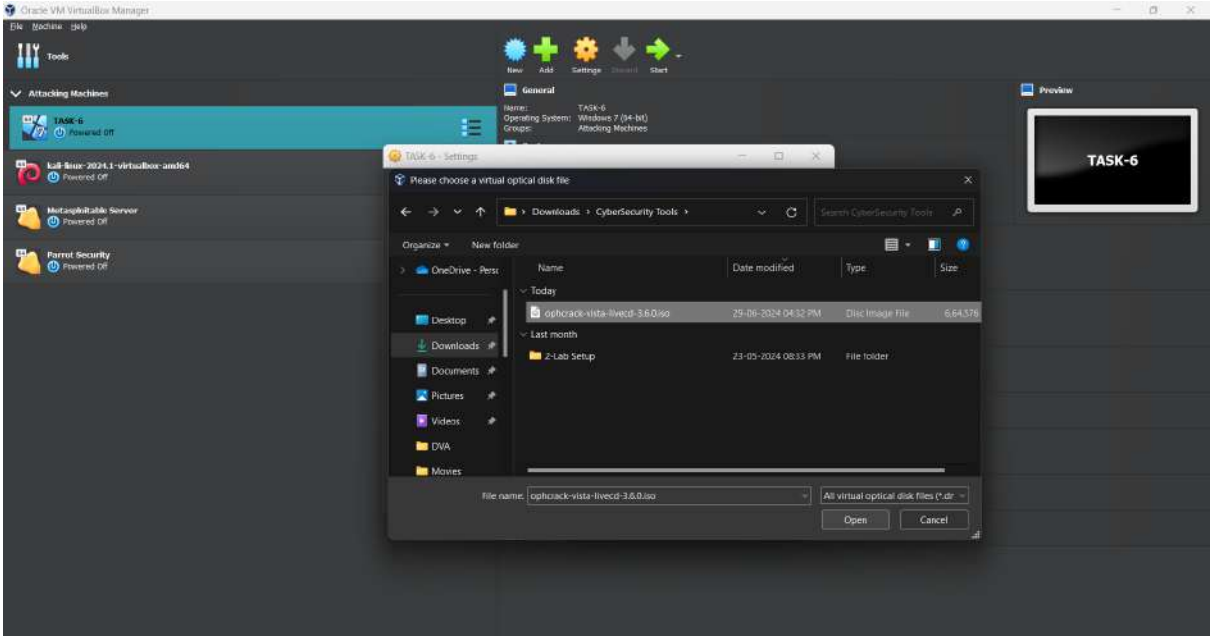
1.1 Change the Boot Order of the Machine and Bring Optical Disk to the top



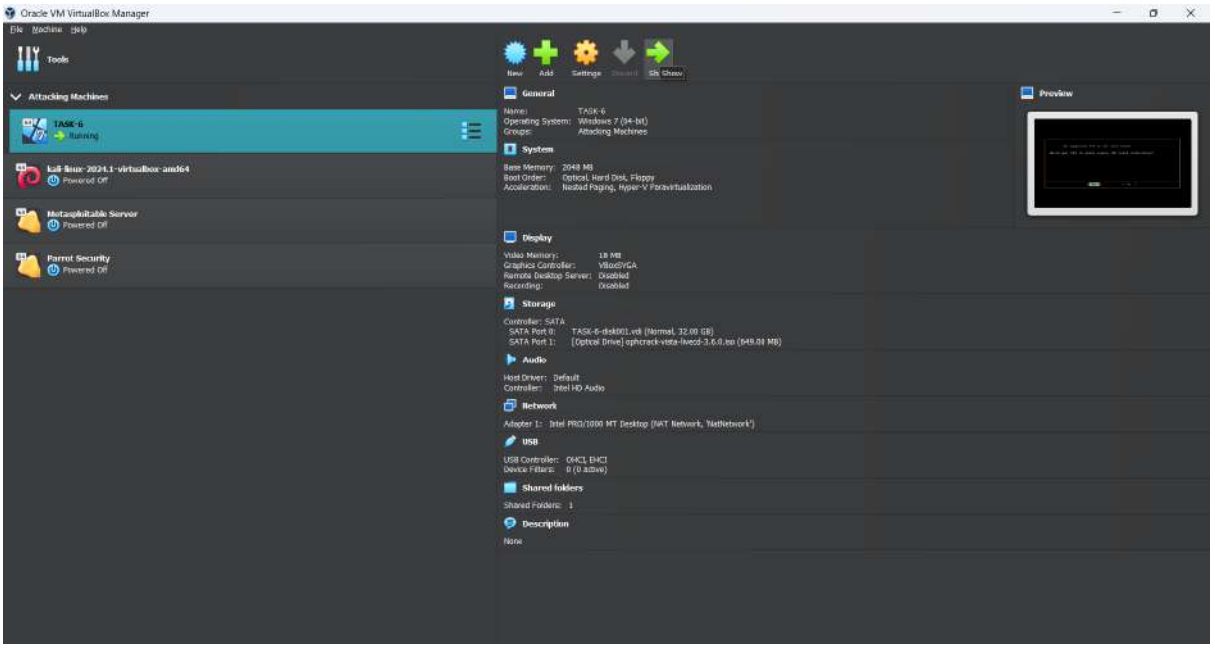
1.2 Now head to the Storage Section, and select - **Choose a disc file...** to modify the iso.



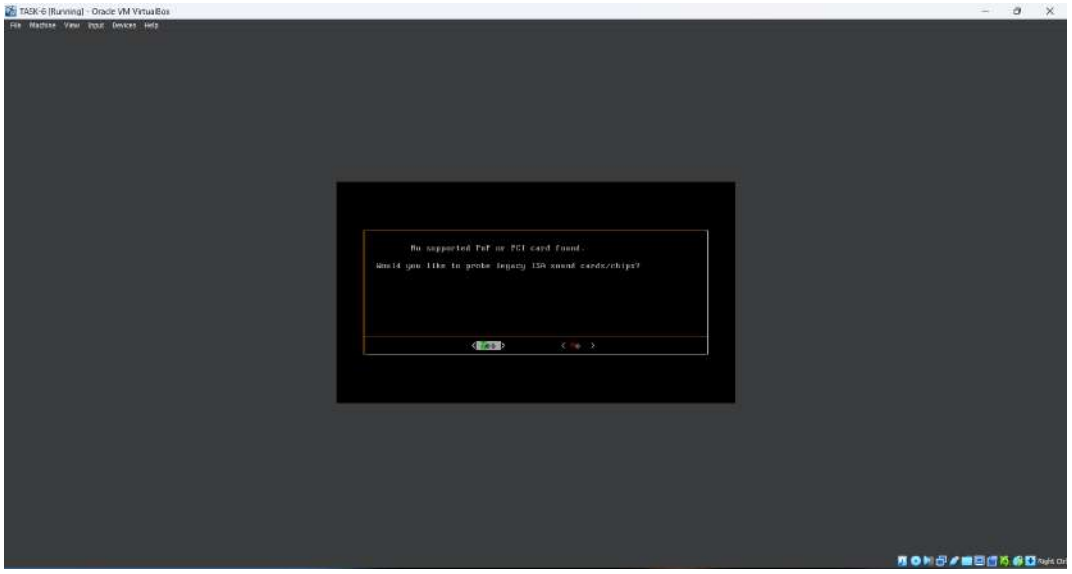
1.3 Select the OPH Crack tool from its directory and Save everything.



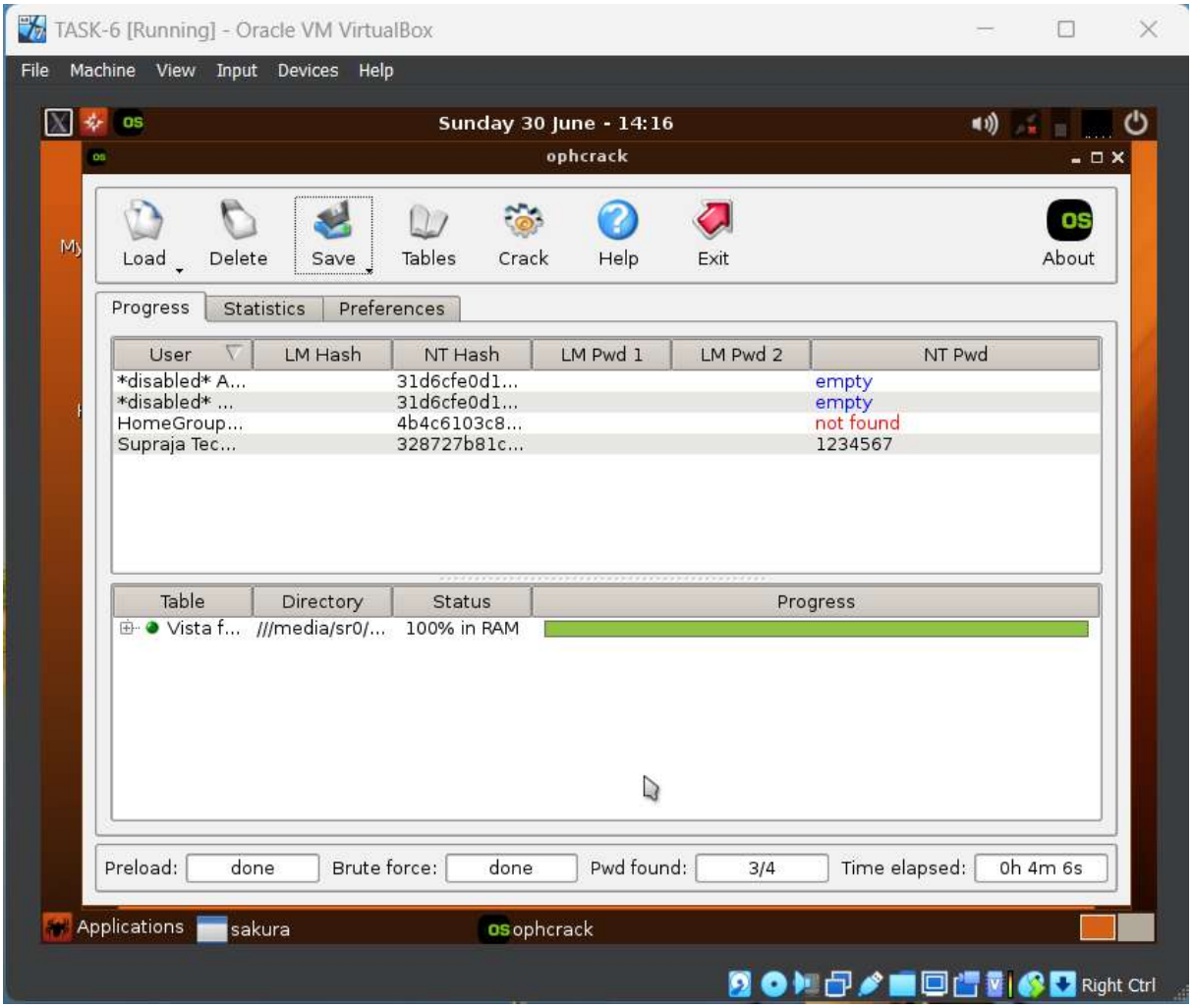
1.4 Start the Windows 7 OS now...



1.5 The OPH Crack Tool is now Booted, just click Enter three times to start cracking the password process.



1.6 Select the user and Click **Crack**

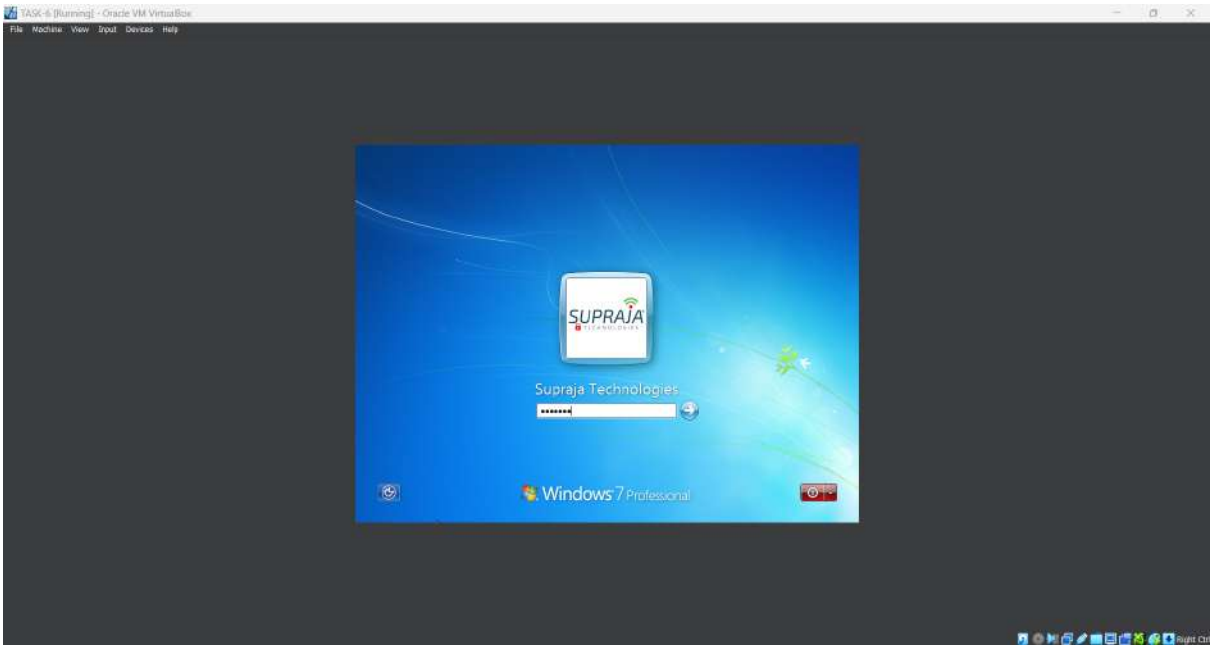


Conclusion-

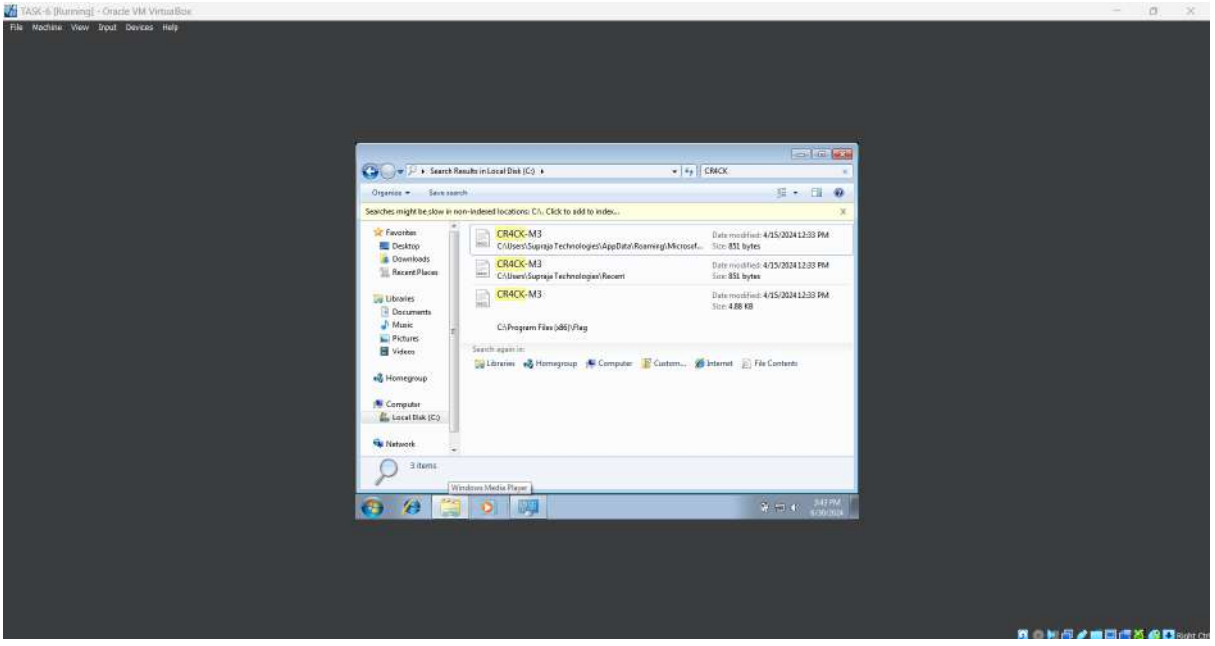
As we can see the password is **1234567**

STEP 2 - Check the machine, if it consists of any files.

2.1 Once the password is cracked, Run the Windows 7 Virtual Machine and try to login with the cracked password i.e 1234567.



2.2 Once Logged In, Check for any files in the C Drive.

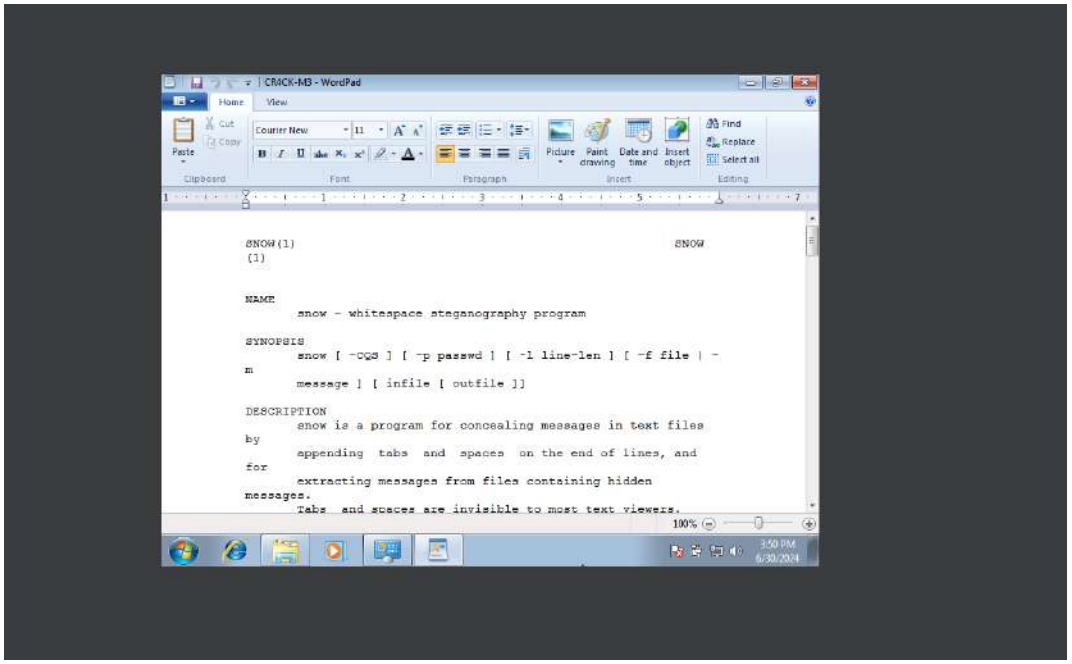


Conclusion-

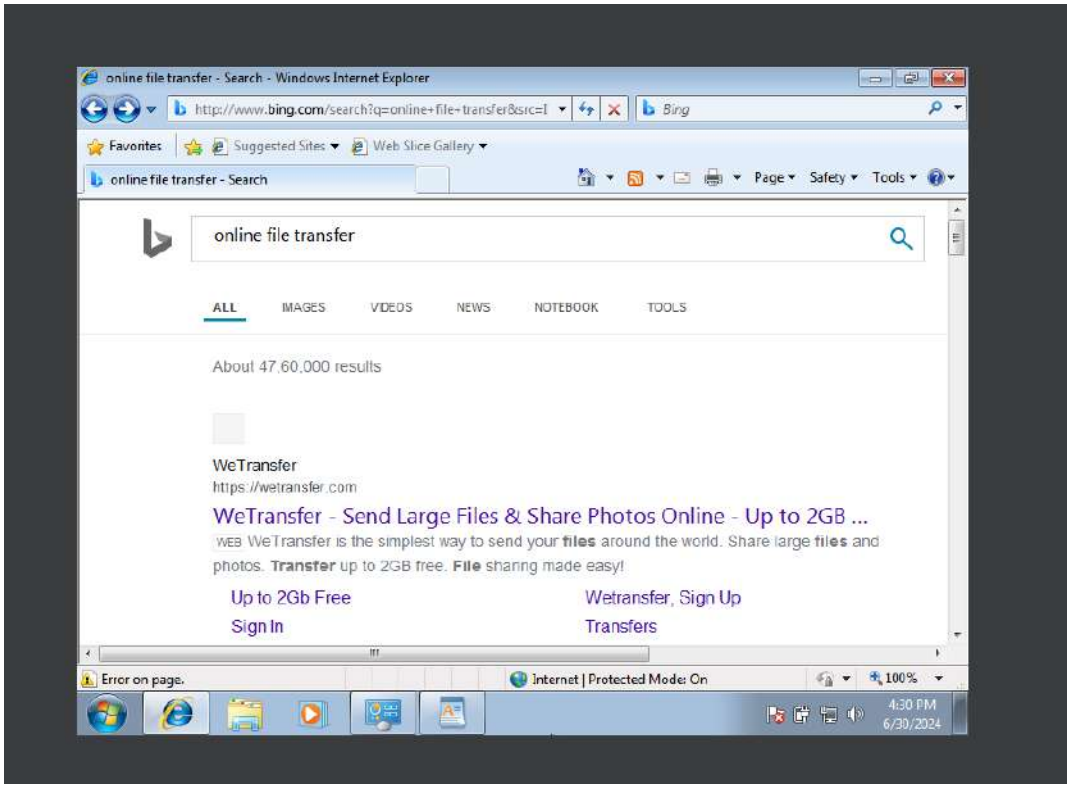
We found a Flag at this Location - **C:\Program Files (86x)\Flag\CR4CK.txt**

C. Analysing the Checksums

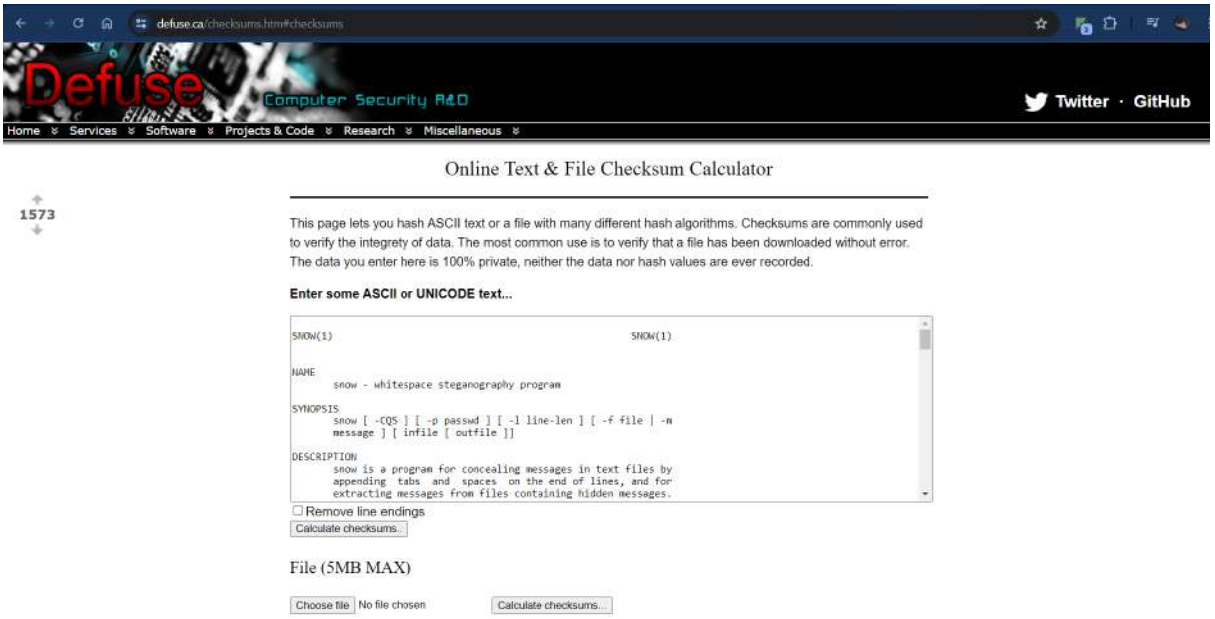
STEP 1 - Check the files in the system



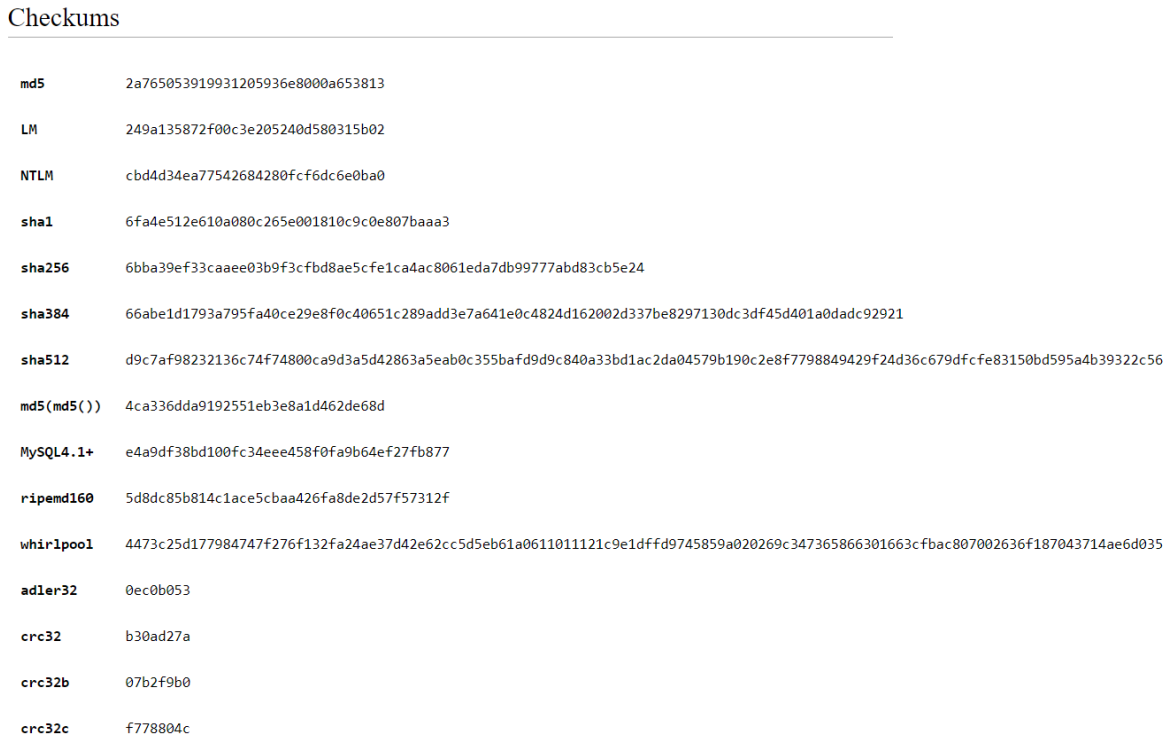
STEP 2 - Transfer the file to your host machine for analysing it.



STEP 3 - Calculate the Checksums for it using Checksum Claculator Tool. Just go to this online tool defuse.ca.



STEP 4 - These are the calculated Checksums



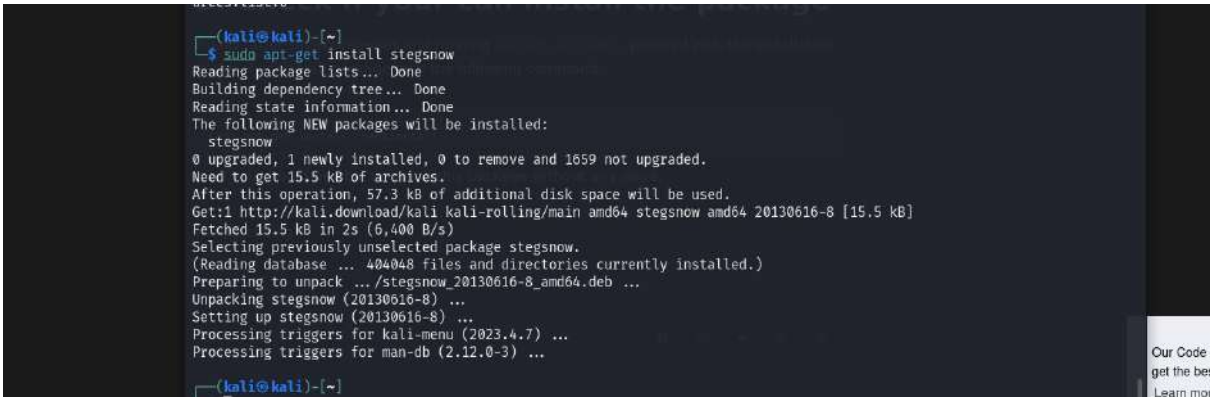
Conclusion-

Since these checksums do not match with those provided in the README file, this file is suspected to be tampered.

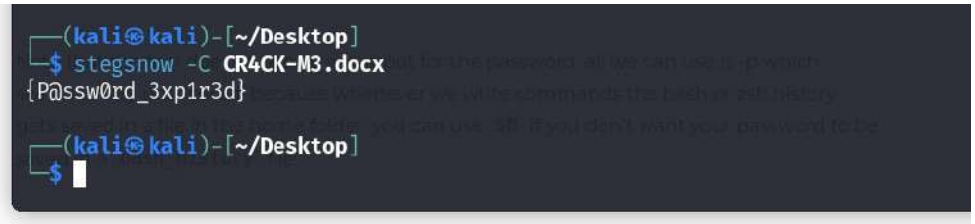
STEP 5 - Try to Identify the hidden data inside the Tampered document.

We will try to use SNOW Stenography Tool to identify the hidden message in the document. For this first open kali linux terminal.

STEP 6 - Now install the stegsnow package with `sudo apt-get install stegsnow`



STEP 7 - Once stegsnow is Installed, go ahead and give this command `stegsnow -C CR4CK-M3.docx` **to find the hidden message.**



Conclusion - Identify the FLAG {***
}**

Finally the hidden message is revealed which is **{P@ssw0rd_3xp1r3d}**

Summary-

This report outlines the process of cracking a password-protected system, specifically a Windows 7 virtual machine, to find a hidden flag. It started with decrypting a secret data link using the **CyberChef** tool. Then, an OVA file was downloaded and imported into a VirtualBox environment. Using the OPH Crack tool, the system password was cracked and used to login to the machine. On exploration, a flag was found in a specific directory.

The **checksums** of the file were then calculated and compared with those in the README file, which revealed the file to be tampered with. Using the **SNOW Stenography Tool** on Kali Linux, a hidden message was uncovered in the tampered document, which revealed the final flag which was **{P@ssw0rd_3xp1r3d}**.

Supraja Technologies

23E05-ST#IS#6653-TASK7

A. Find 2 websites vulnerable to Directory/Path traversal Vulnerability by using different payloads of Local File Inclusion.

Local File Inclusion Vulnerability

- **CVSS Score** - 7.5 (High)
- **Related OWASP Top 10** - A5:2021 - Security Misconfiguration
- **Explanation**: Local File Inclusion (LFI) is a vulnerability that allows an attacker to include files located on the same server as the vulnerable application. This typically occurs when an application uses user-supplied input to include files without proper validation or sanitization. Attackers can exploit this vulnerability to access sensitive files, execute malicious scripts, or potentially achieve remote code execution under certain conditions.
- **Impact**:

1. Information disclosure: Unauthorized access to sensitive files (e.g., configuration files, system files).
2. Code execution: Potential execution of malicious code if the server is misconfigured.
3. Application compromise: Access to application source code or configuration details.
4. Privilege escalation: Possible escalation of privileges if sensitive information is obtained.

- **Recommendations**:

1. Input validation: Implement strict input validation for all user-supplied data used in file inclusion.
2. Whitelist approach: Use a whitelist of allowed files or directories that can be included.
3. Avoid using user input: If possible, avoid using user-supplied input for file inclusion altogether.
4. Use mapping: Implement a mapping system where filenames are associated with predetermined paths.
5. Disable directory traversal: Ensure that directory traversal sequences (../) are properly neutralized.

- **References**:

1. **OWASP: Testing for Local File Inclusion:** https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/07-Input_Validation_Testing/11.1-Testing_for_Local_File_Inclusion

- **Procedure:**
    - **Target Website -** Confiture de bali

- **Payload -** ../etc/passwd

There are 5 payloads of Local File Inclusion.

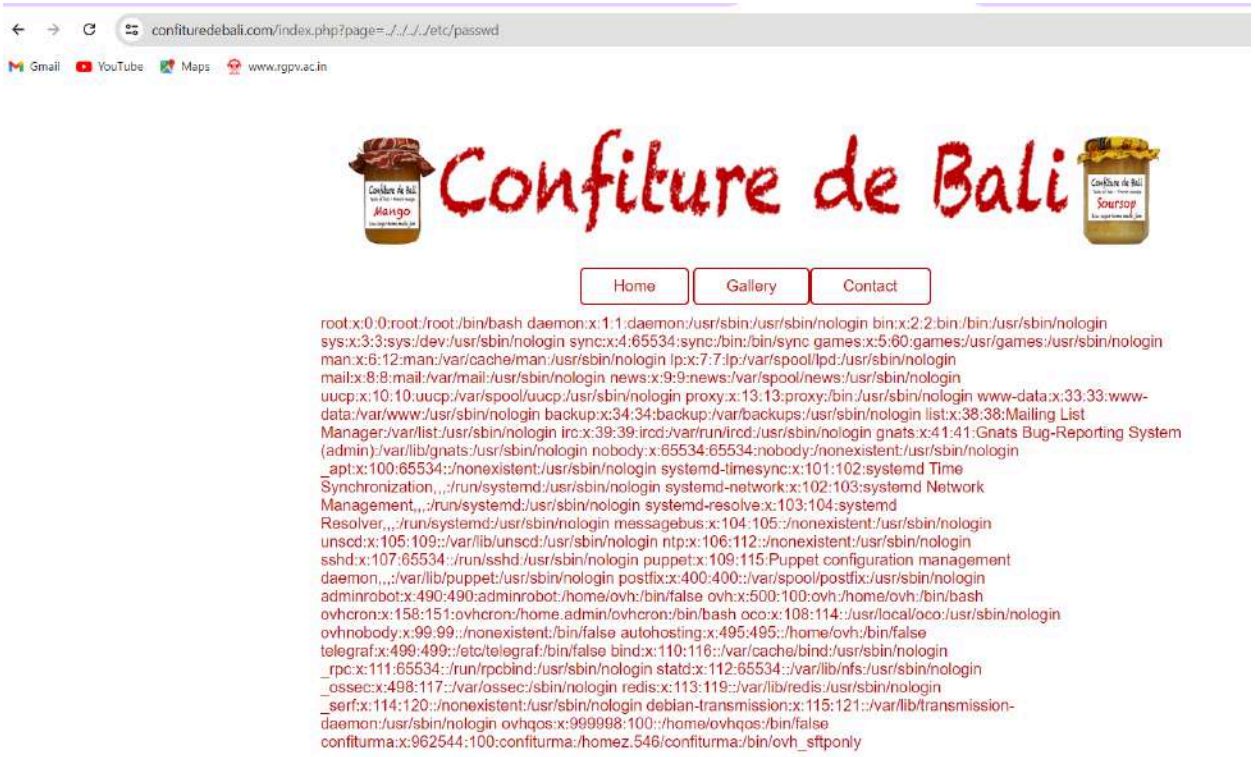
a. LFI using simple case - ../etc/passwd
b. LFI using Null Byte - %00
c. LFI using single URL encoder
d. LFI using double URL encoder
e. LFI using start path

STEP -1 Open browser and search for website confiture de bali.

STEP -2 Go for gallery url and do path traversal using different payloads of Local File Inclusion .

1.(a) LFI using simple case - ../etc/passwd

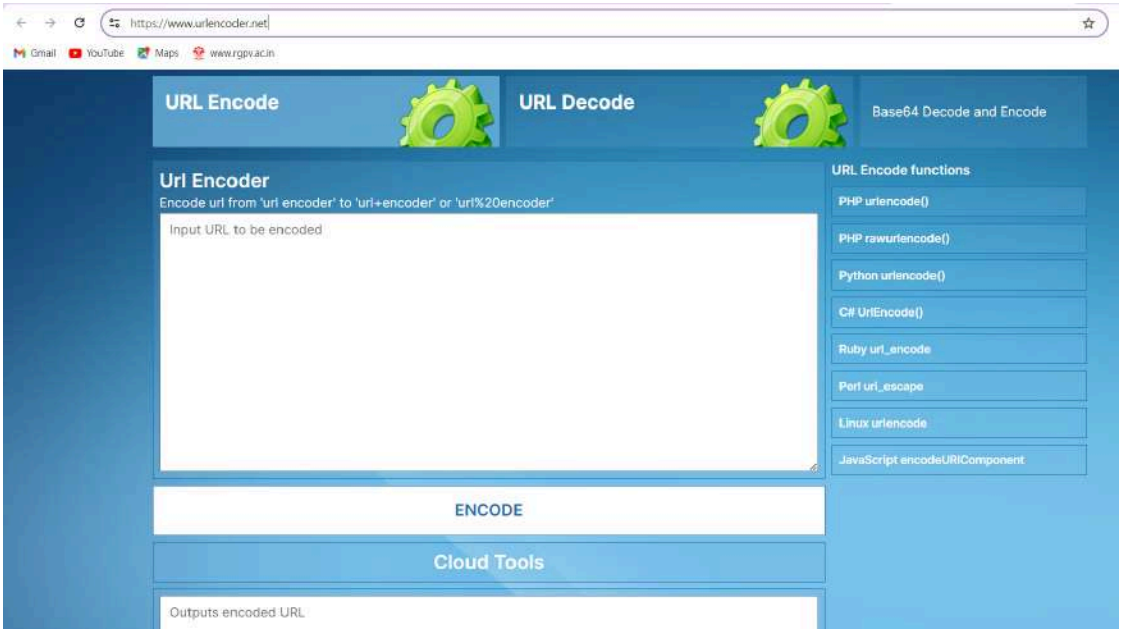
Result



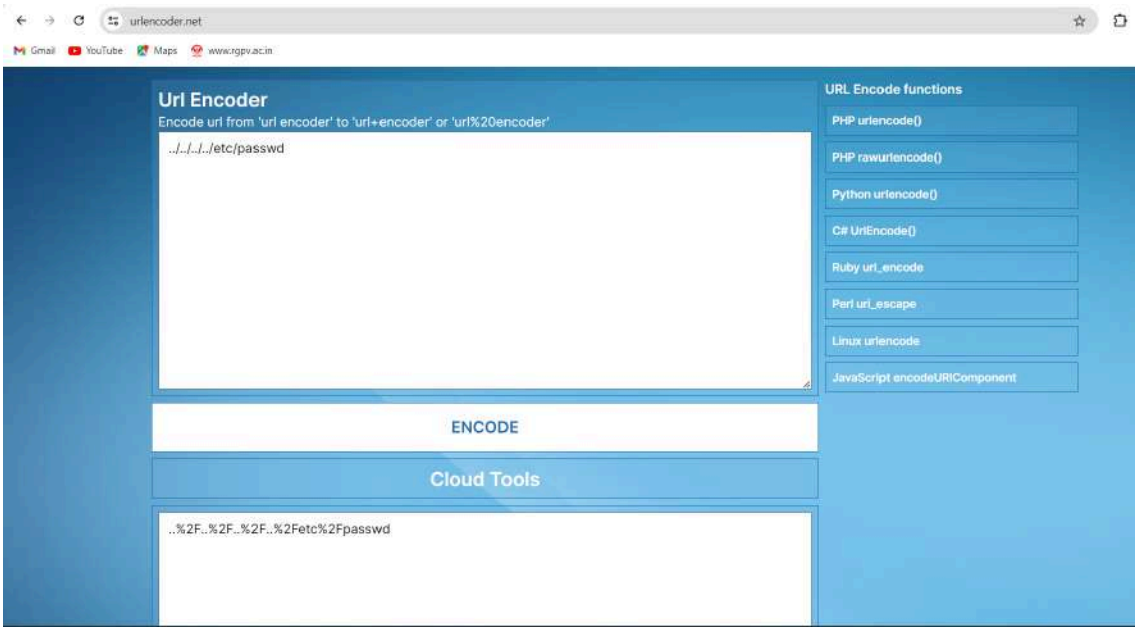
1.(b) LFI using Null Byte - ../../../etc/passwd %00

1.(c) LFI using a single URL encoder .

STEP-1.1 Open browser and search for url encoding.



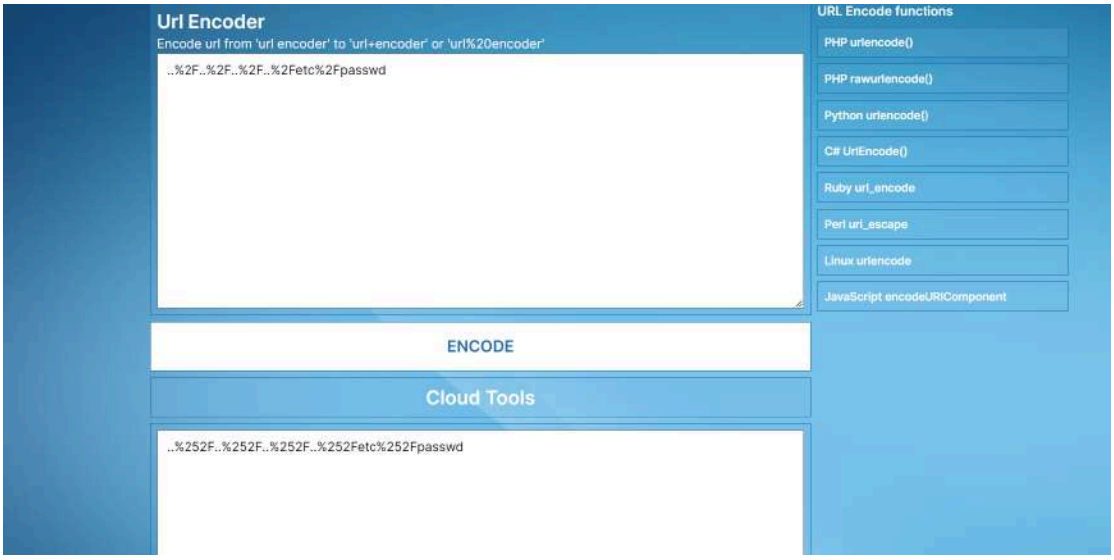
STEP-1.2 Encode URL and copy encoded output.



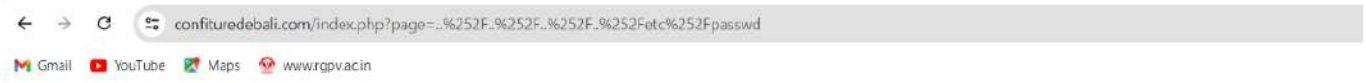
STEP-1.3 Paste encoded output at the URL Section.

1.(d). LFI using a double URL encoder.

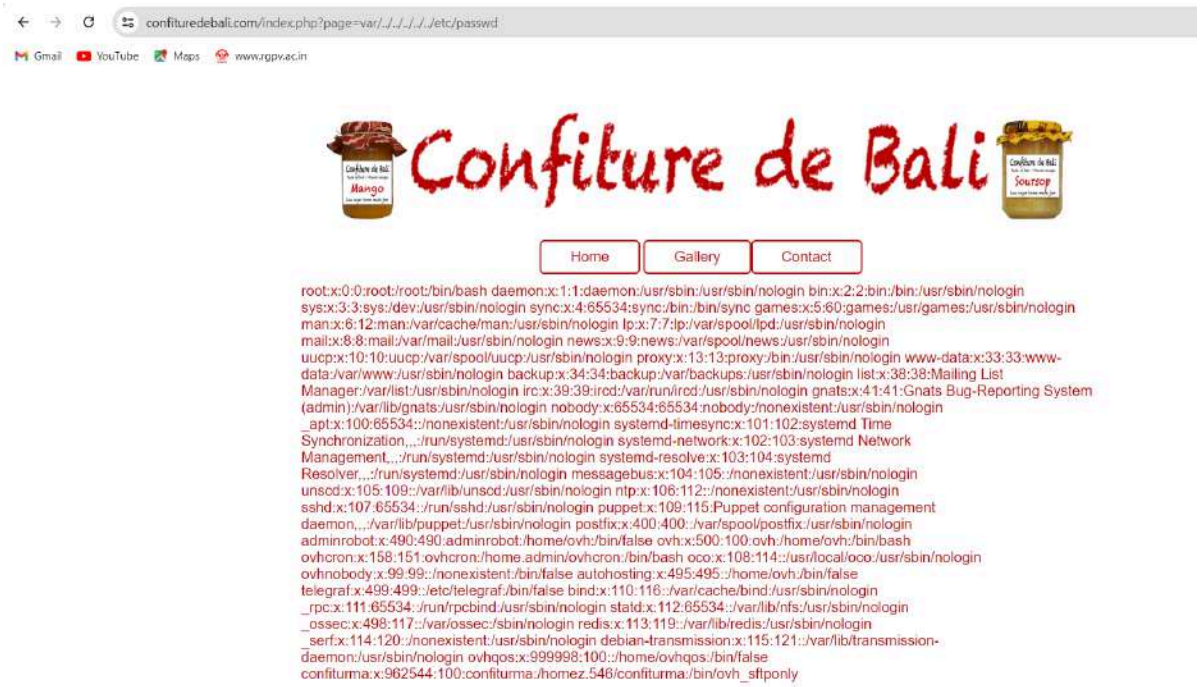
STEP-1.1 Encode the Single Url encoded output and copy double encoded output.



STEP-1.2 Paste the double encoded URL output.



1.(e). LFI using the start path.



B. Find 2 websites vulnerable to HTML Injection Vulnerability.

HTML Injection Vulnerability

- **CVSS Score** - 6.1 (Medium)
- **Related OWASP Top 10** - A03:2021 - Injection
- **Explanation**: HTML Injection is a type of injection vulnerability where an attacker can insert arbitrary HTML code into a vulnerable web page. This occurs when user input is not properly sanitized or encoded before being rendered in the browser. While less severe than some other injection types, HTML injection can still lead to various security issues, including laying the groundwork for more serious attacks like Cross-Site Scripting (XSS).
- **Impact**:

1. Input validation: Implement strict input validation for all user-supplied data.
2. Output encoding: Always encode user-supplied data before rendering it in HTML context.
3. Content Security Policy (CSP): Implement a strong Content Security Policy to mitigate the impact of successful injections.

4. Use security-focused libraries: Utilize well-maintained, security-focused libraries for handling user input and HTML rendering.
5. Sanitization: Use proper HTML sanitization techniques to remove potentially dangerous HTML elements and attributes.

❖ **Recommendations**:

1. Validate and sanitize the Host header
2. Use allowlists
3. Avoid using user-supplied Host headers
4. Implement proper URL parsing
5. Use HTTPS
6. Implement additional security headers like X-Frame-Options, Content-Security-Policy, and Strict-Transport-Security.

❖ **References**:

1. *OWASP XSS Prevention Cheat Sheet:* https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html,

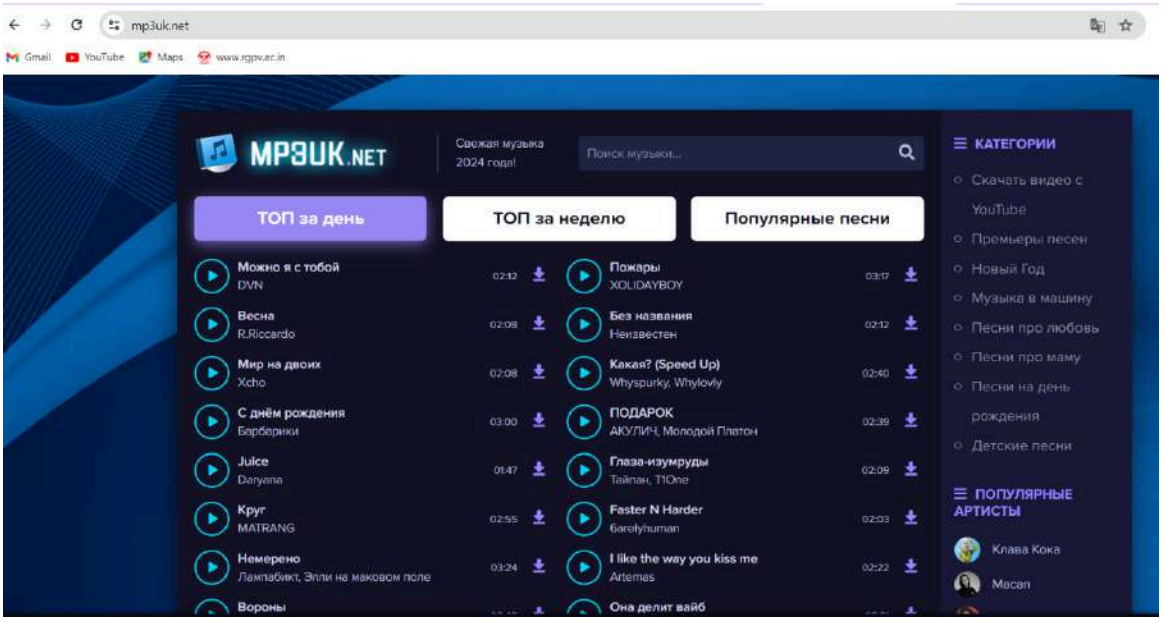
❖ **Procedure:**

➢ **Target Website -** https://mp3uk.net/ and https://boxberry.ru/

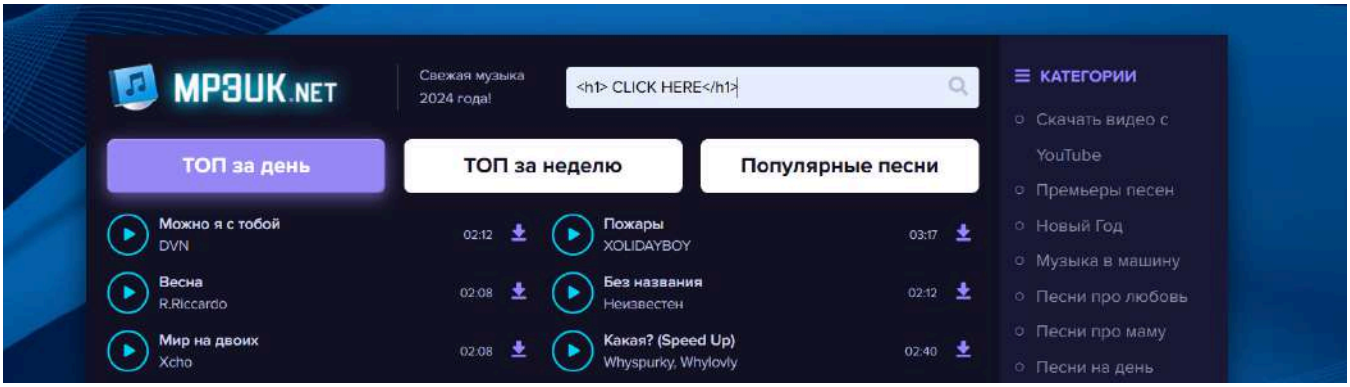
➢ **Payload -** <h1> CLICK HERE</h1>

➢ **Steps -**

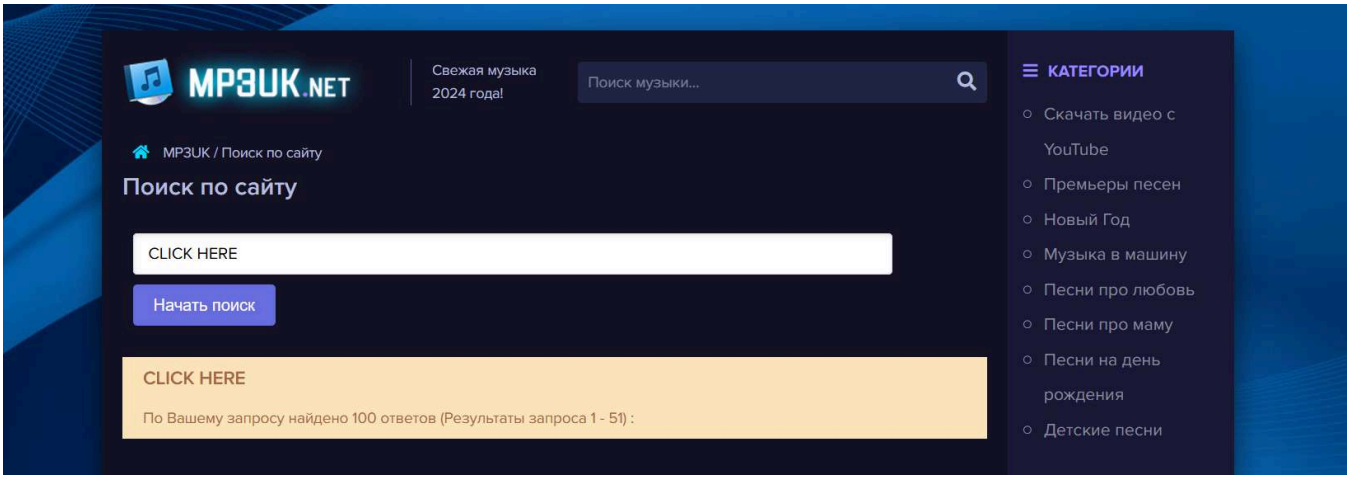
STEP-1: Open your browser and go to https://mp3uk.net/



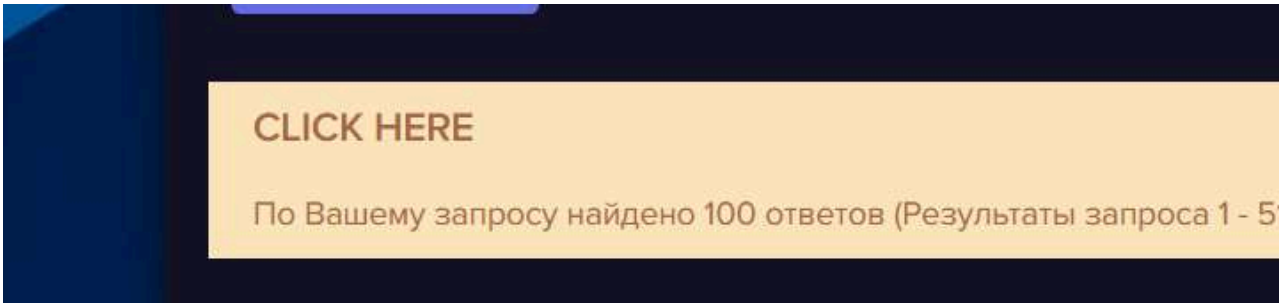
STEP 2 - Now check for HTML injection by entering the HTML payload in the search field.



STEP 3 - Click Enter and Check for any Heading in the Results.



STEP 4 - As we can see the heading CLICK HERE is rendered on the website, this confirms that the site has HTML Injection Vulnerability.



For 2nd website:

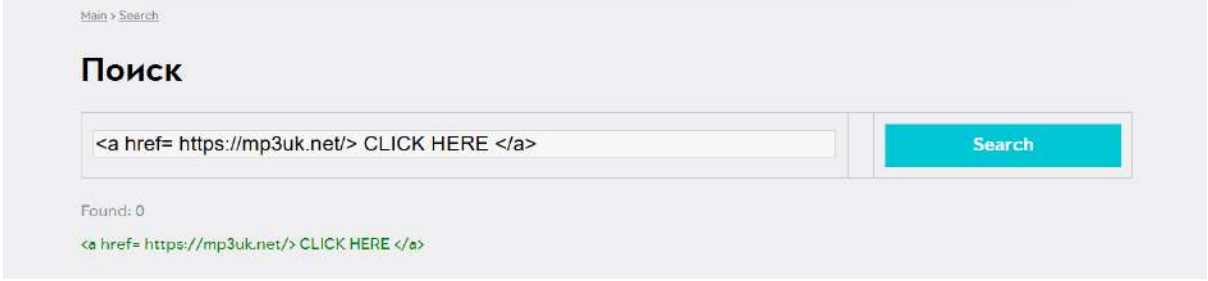
.

a. **Payload -** CLICK HERE
b. **Target - https://old.jewish-museum.ru/**

STEP 1 - Go to the Target Website.

STEP 2 - Inject the payload in the search field and click enter.

STEP 3 - Check the Results.

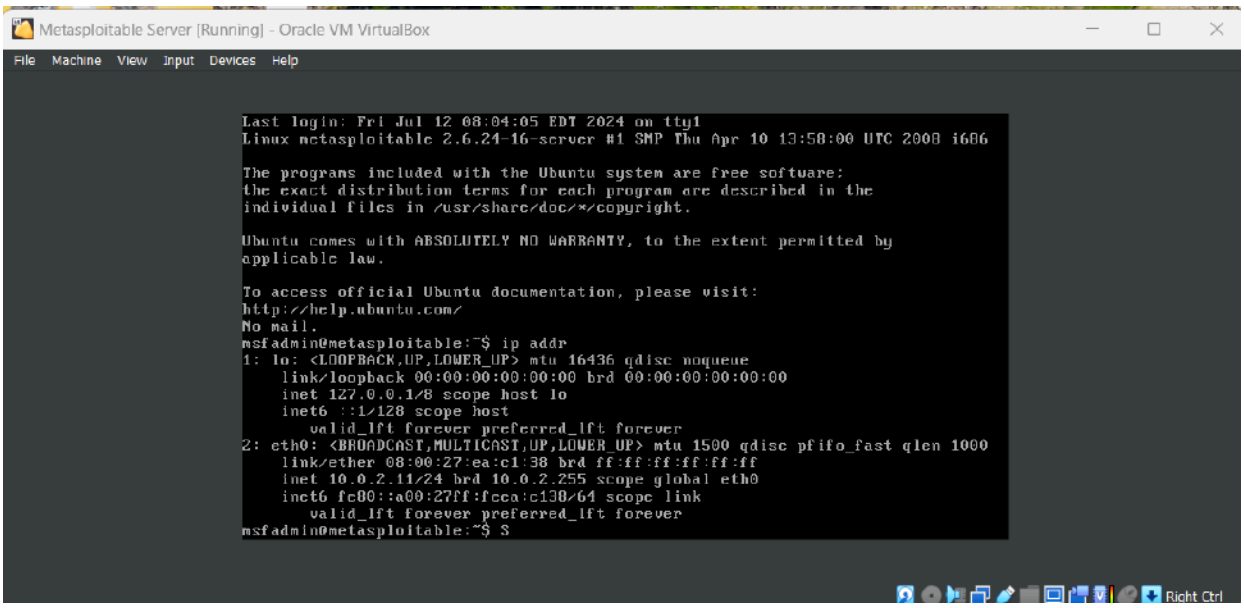


C. Find 2 websites vulnerable to File Upload Vulnerability on each test case below.

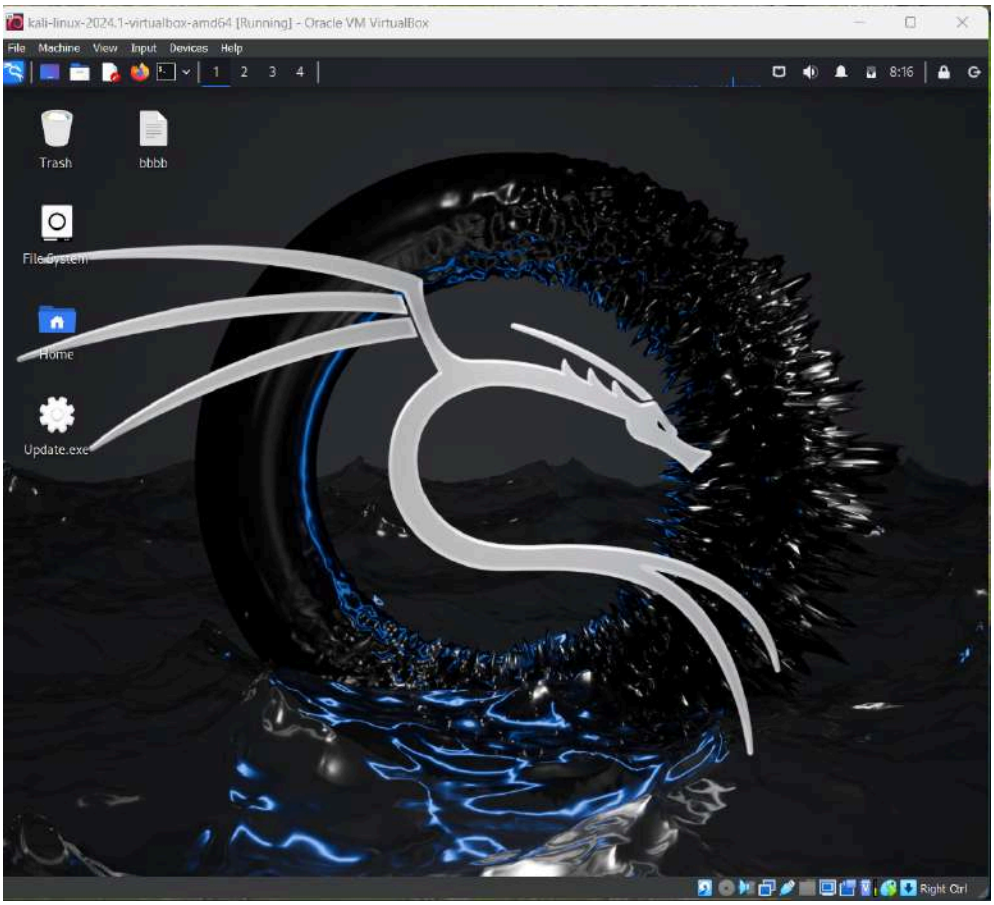
a. Uploading larger PDF files than the specified size.
b. Uploading images in the place of pdf.
c. Uploading malicious PHP code in the place of pdf.

STEP-1.1 Open Virtual box .

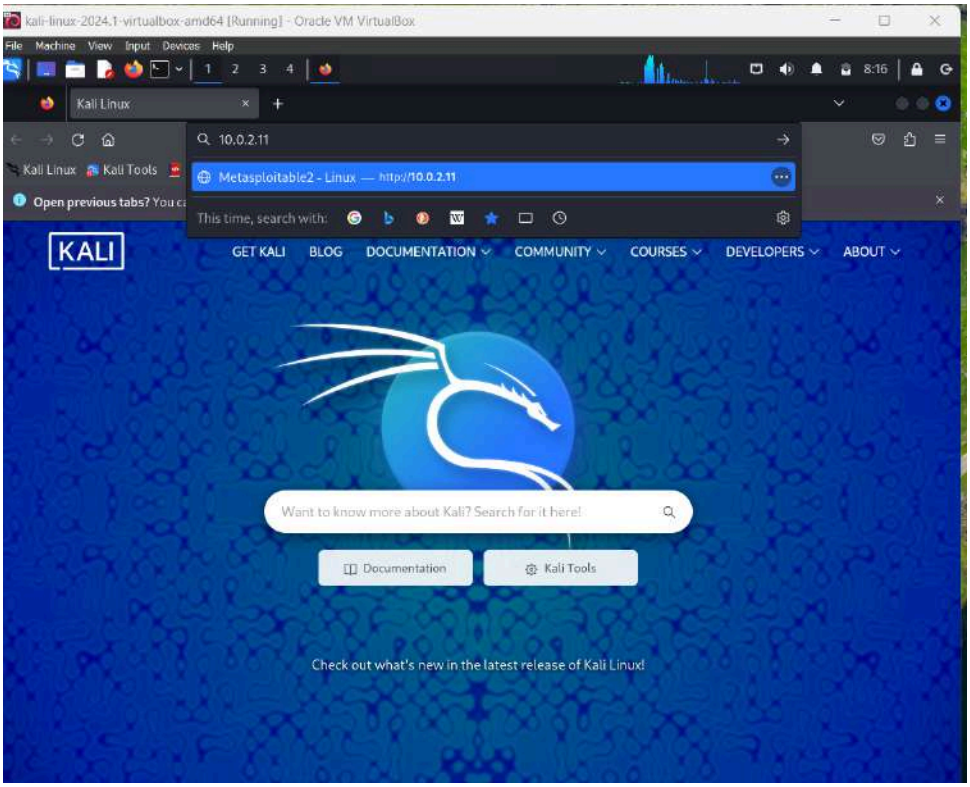
STEP- 1.2 Start metasploitable server and check the ip address of metasploitable server.



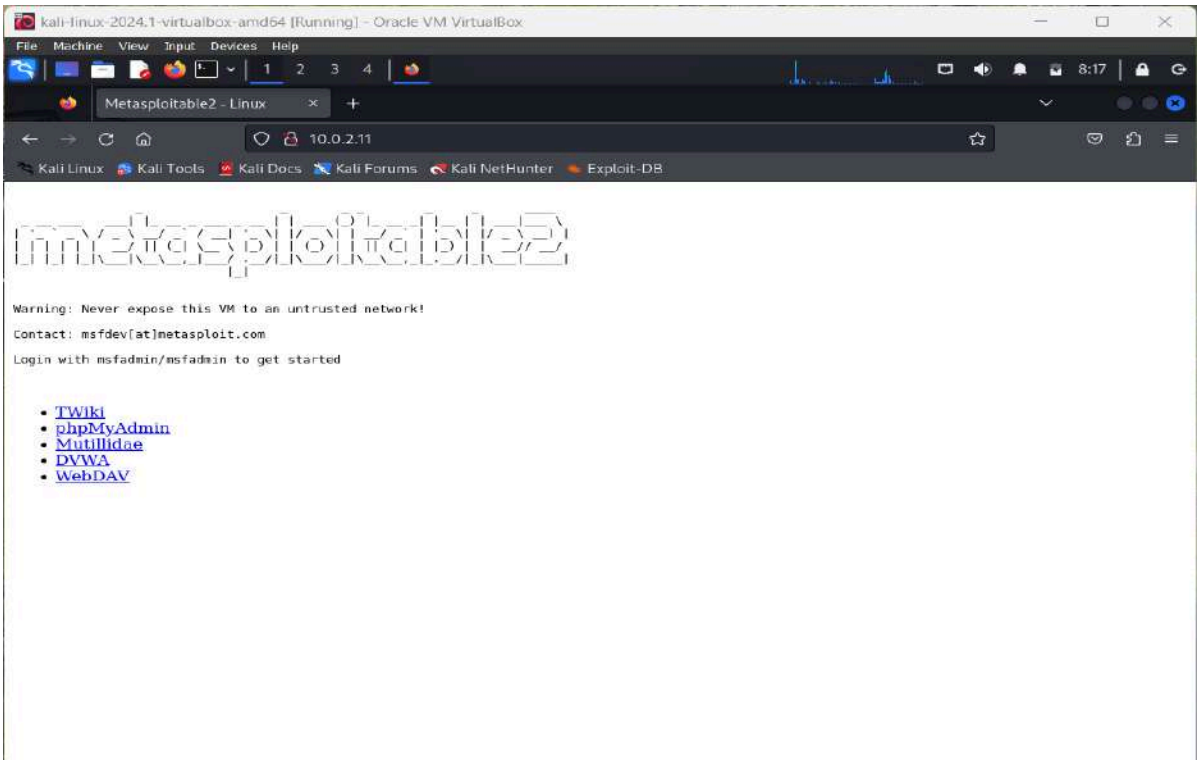
STEP-1.3 start kali linux with the same network as metasploitable server.



STEP-1.4 Open firefox in kali linux and search for metasploitable ip address.



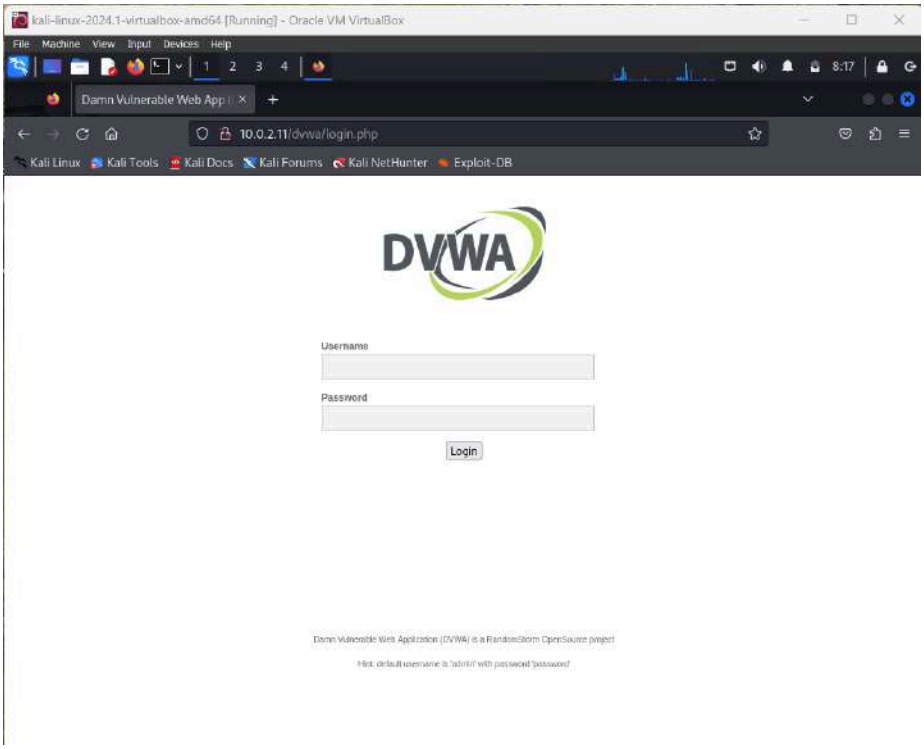
STEP-1.5 Now open DVWA.



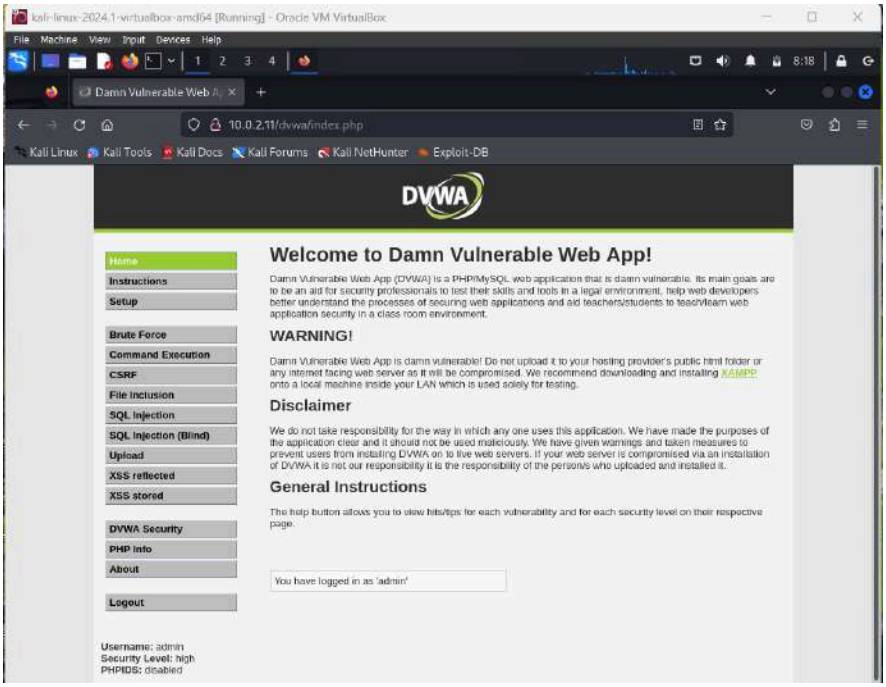
STEP-1.6 Login in DVWA.

With username as admin.

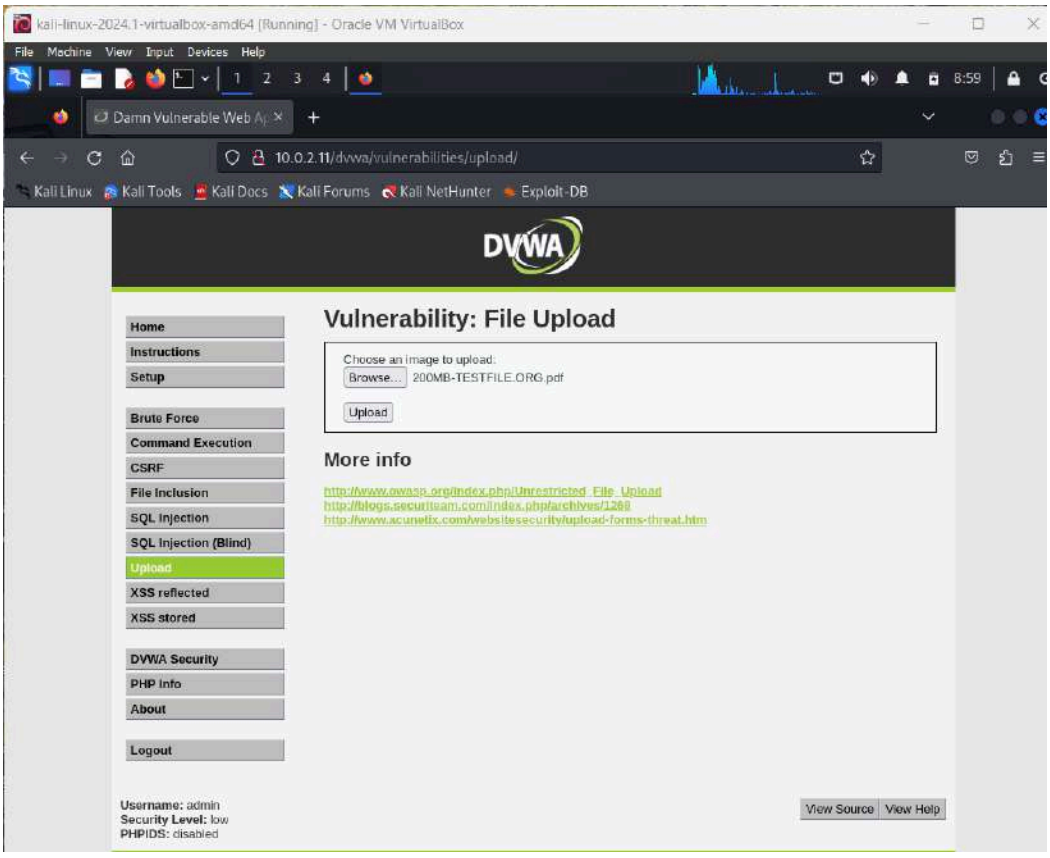
And password as password.



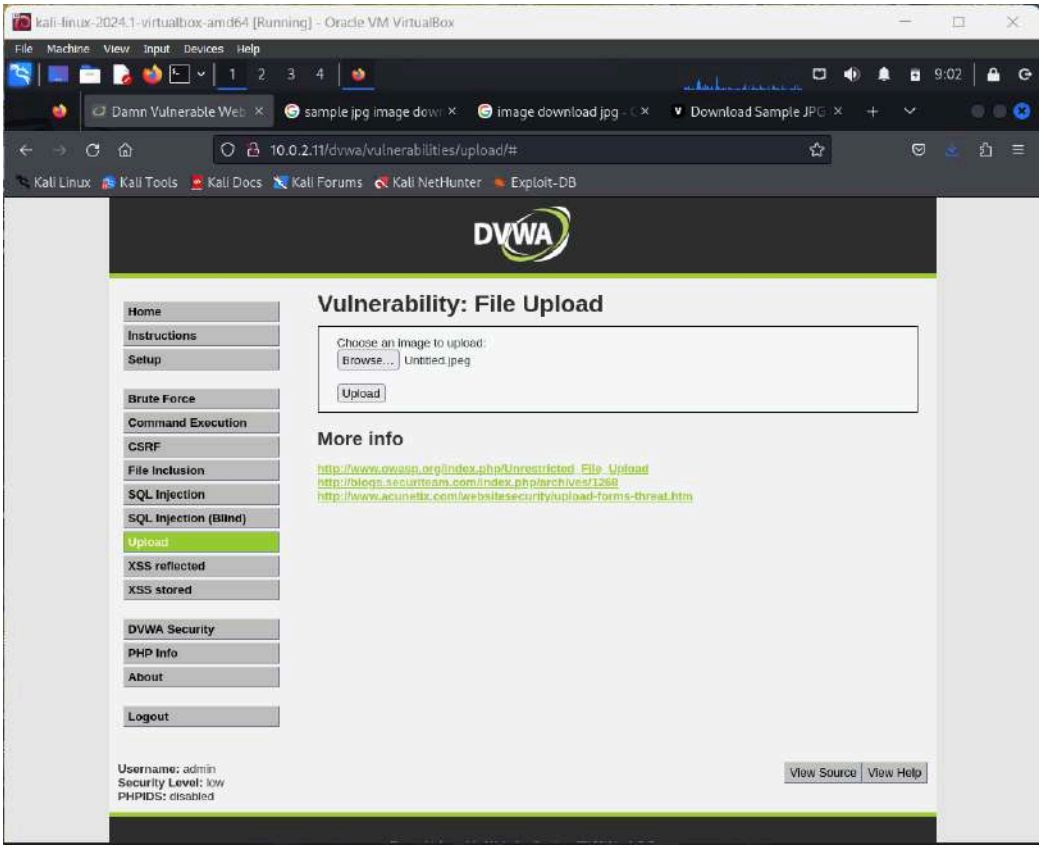
STEP-1.7 Now click on upload section and also change the DVWA security high to low.

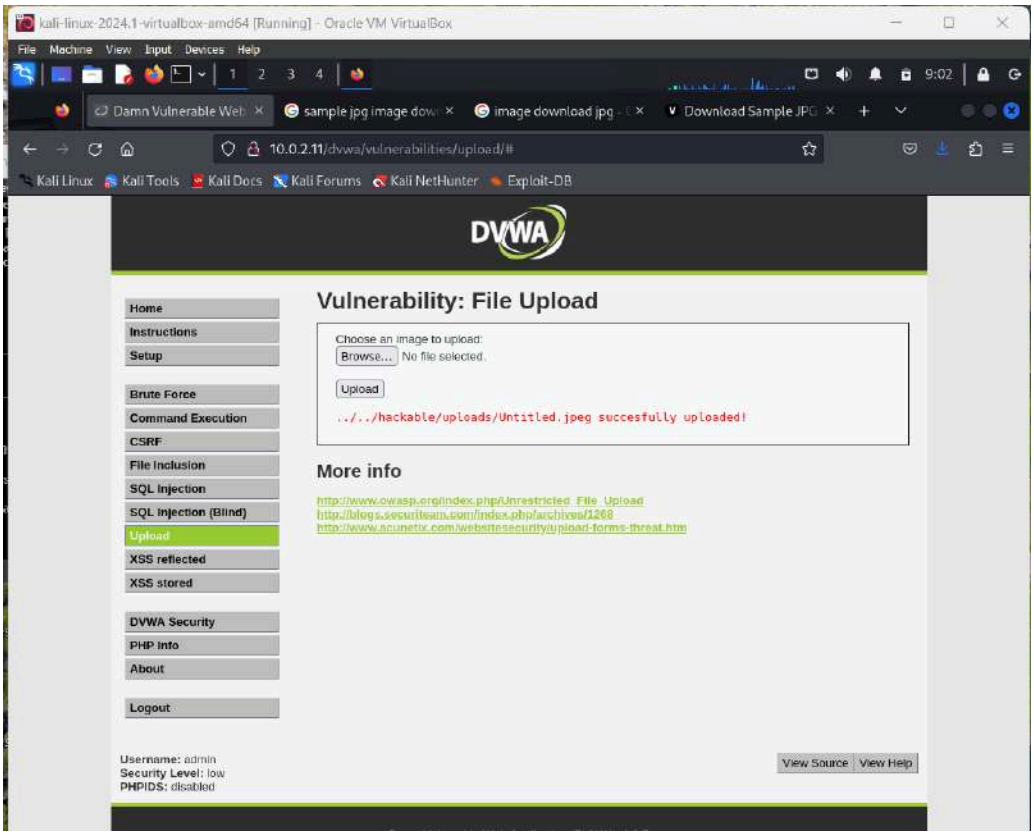


3.(a) upload larger pdf file in place of smaller.

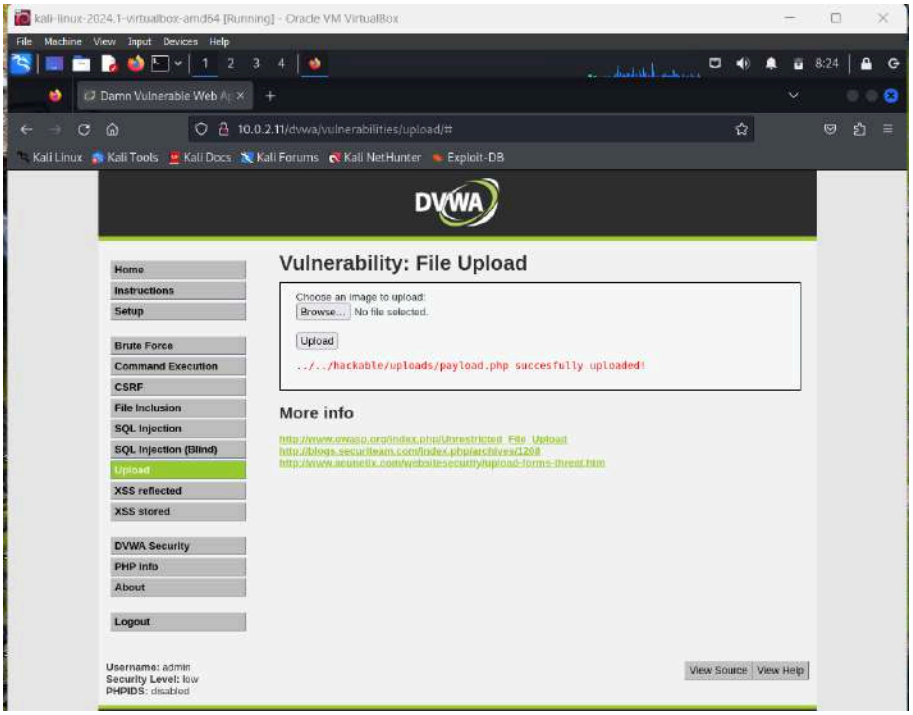


3.(b) upload jpeg and jpeg file uploaded successfully that means file is vulnerable.





3.(c) Upload malicious php code in place of pdf and the file is uploaded successfully that means particular file is vulnerable.



Supraja Technologies

23E05-ST#IS#6653-TASK8

A. Find websites vulnerable to Insecure Design Flaws on each test case mentioned below.

a. No Password Policy
b. Password reset link is not getting expired
c. Automatic email confirmation bug
d. Password reset link sent with http
e. Exposure of private information (privacy violation)
f. Old session doesn't expire

Insecure Design Flaws Vulnerability

- **CVSS Score** - 7.5 (High)
- **Related OWASP Top 10** - A04:2021 - Insecure Design
- **Explanation**: Insecure Design Flaws are vulnerabilities that stem from poor architectural decisions and design choices made during the software development lifecycle. These flaws are often rooted in the lack of threat modelling, secure design patterns, and security-first thinking.
- **Impact**:

1. Cache Poisoning: Injecting malicious content into the cache, affecting multiple users.
2. Password Reset Poisoning: Manipulating password reset links to direct users to attacker-controlled sites.
3. SSRF: Server side request forgery.
4. Phishing

- **Recommendations**:

1. Validate and sanitize the Host header
2. Use allowlists
3. Avoid using user-supplied Host headers
4. Implement proper URL parsing
5. Use HTTPS
6. Implement additional security headers like X-Frame-Options, Content-Security-Policy, and Strict-Transport-Security.

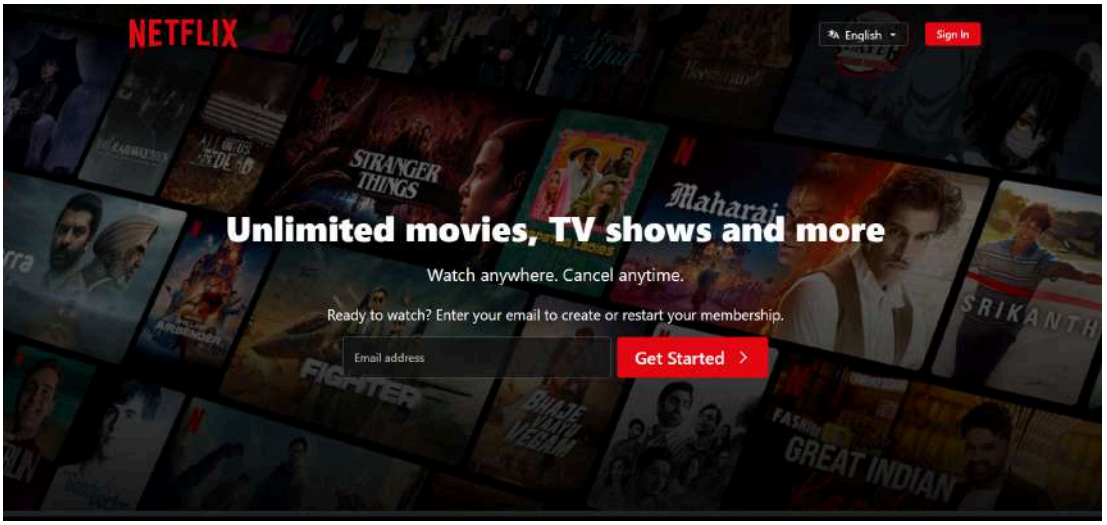
- **References**:

1. OWASP Host Header Attack
2. PortSwigger Research on HTTP Host header attacks

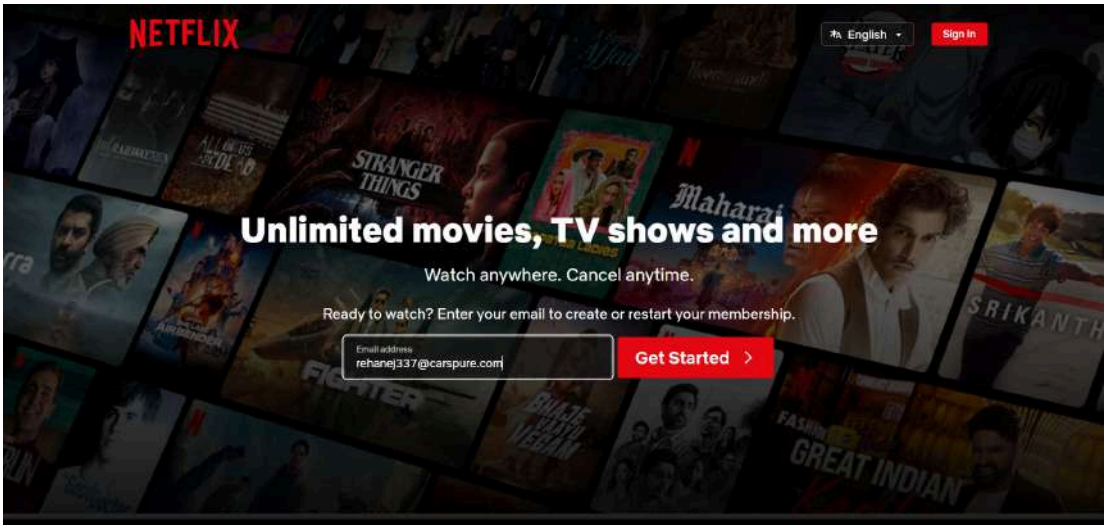
- **Procedure:**
  - **Target Website -** Netflix.com and shopify.com
  - **Steps -**

a. No password policy

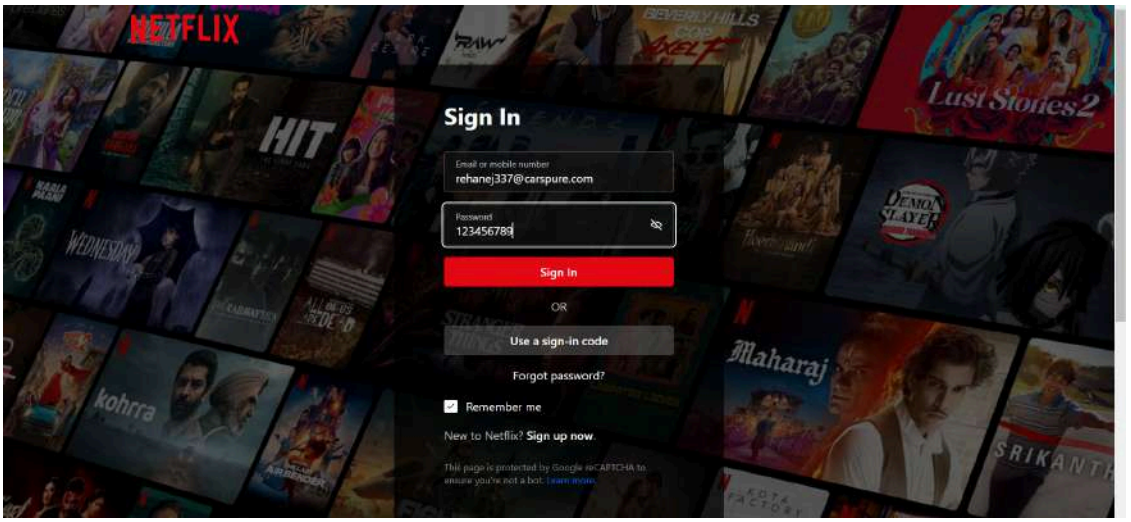
- Select a target website



- 2.Create an account



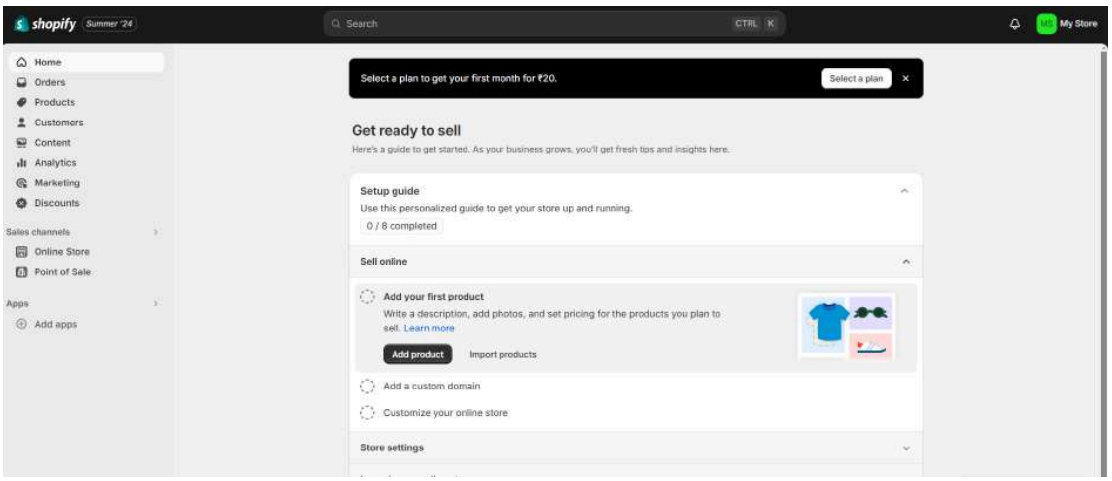
- Set password as anything



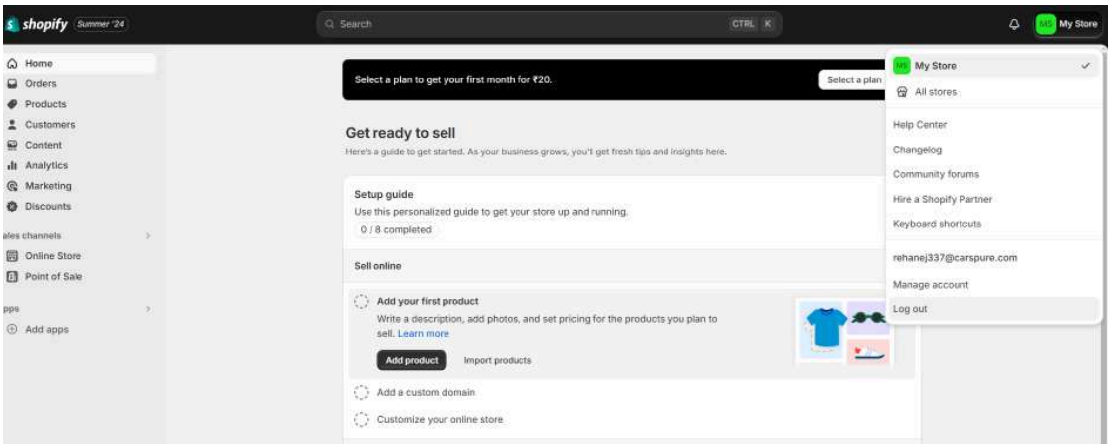
Here we can see that we can set passwords as anything and there is no password policy.

b. Password reset link is not getting expired

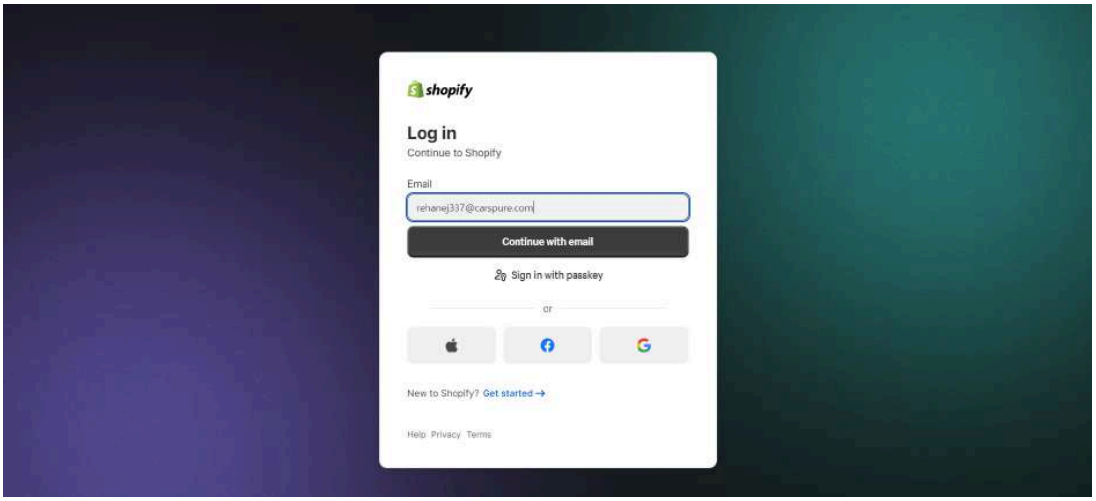
- Create account on target (shopify)



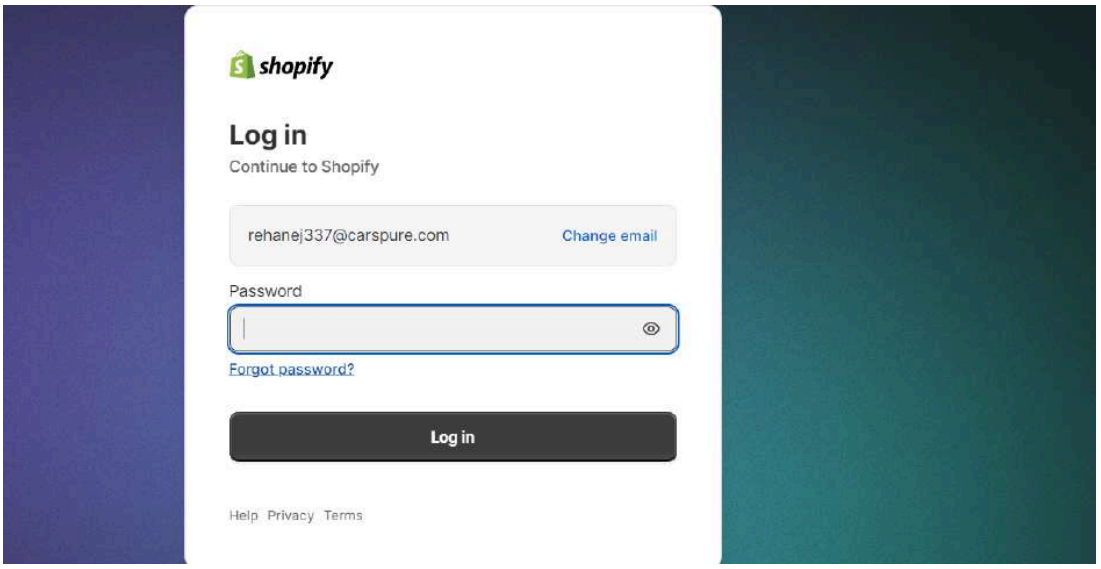
- Log out



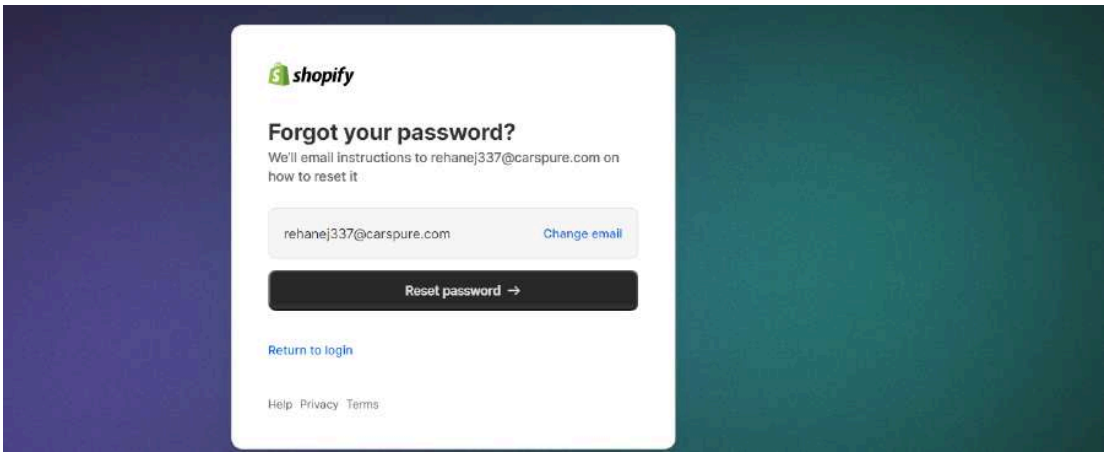
- Log in again



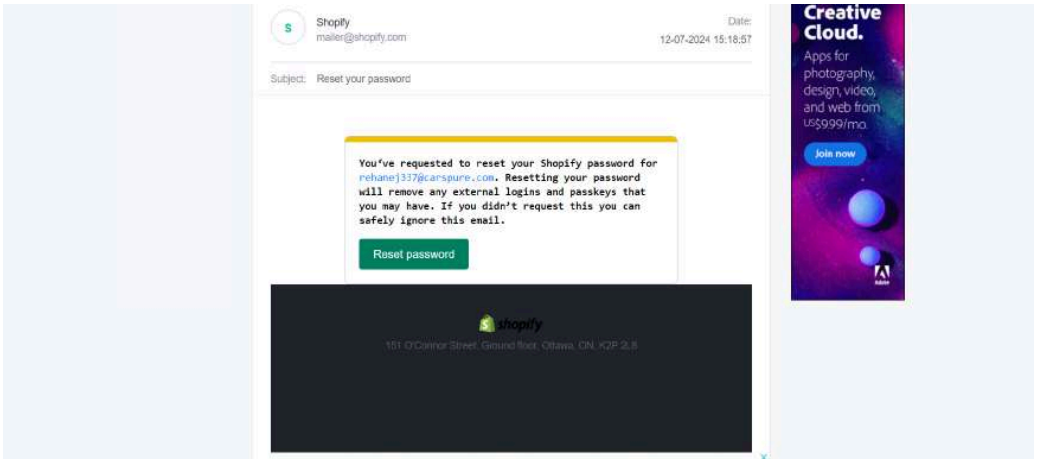
❖ Click on forgot password



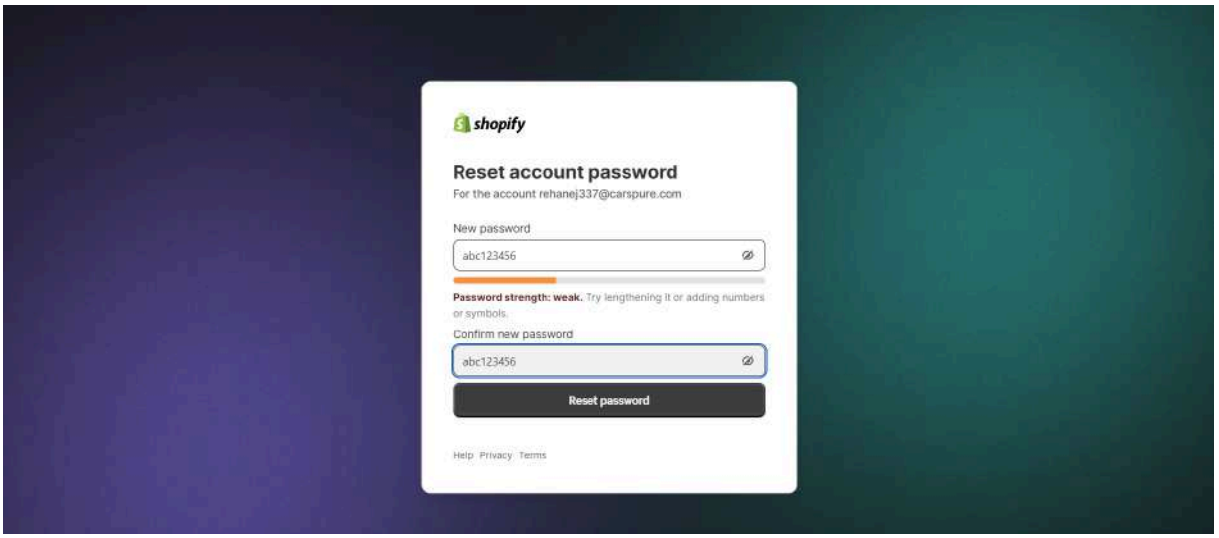
❖ Click on reset password



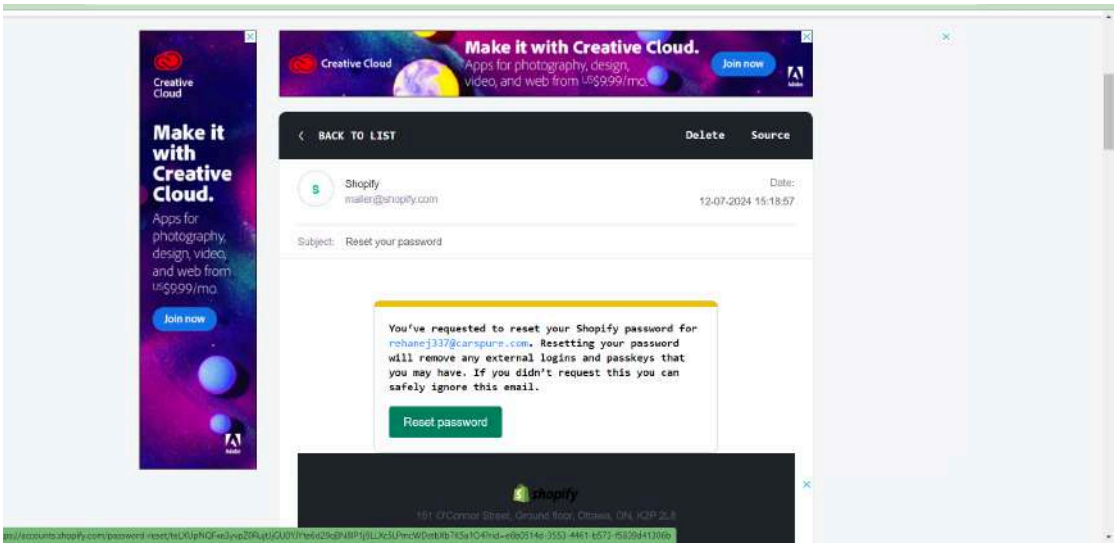
❖ Check for reset link in temp mail

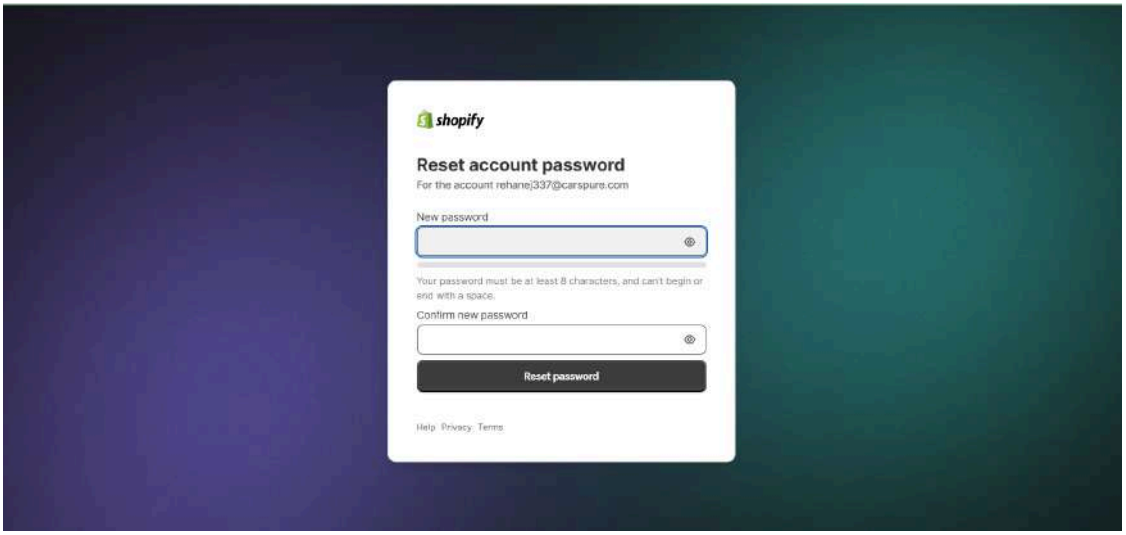


- Reset password



- Now go back to temp mail and click on the reset link again

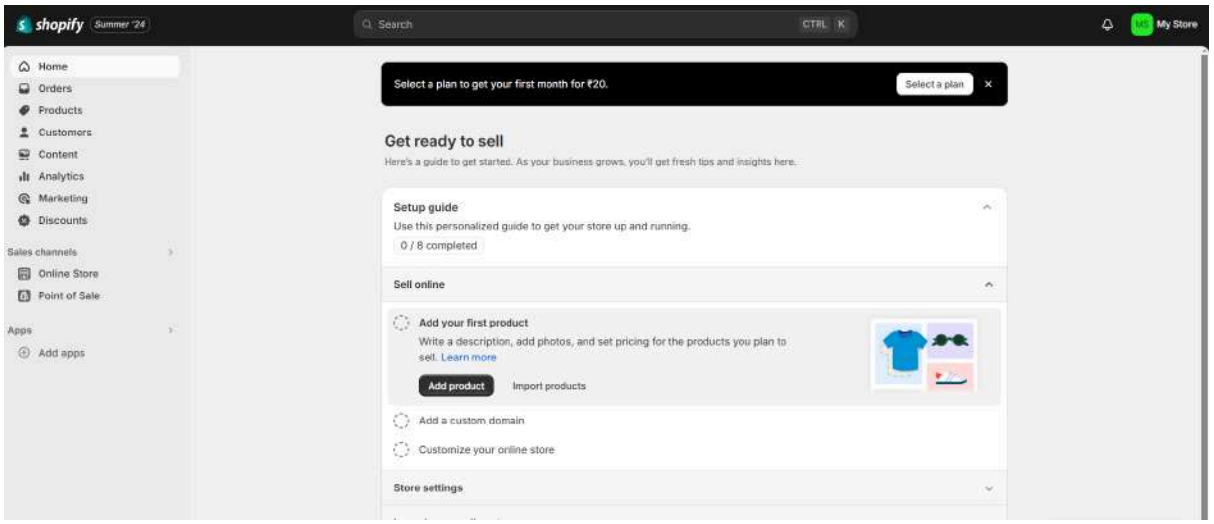




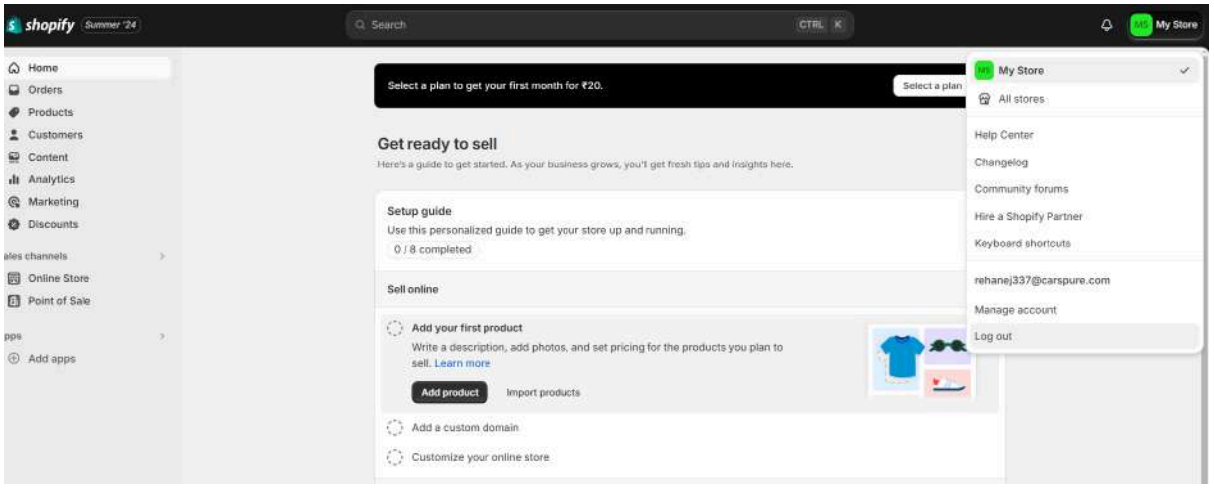
The login page reappears , hence the link does not get expired and hence it is vulnerable

c. Password reset link sent with http

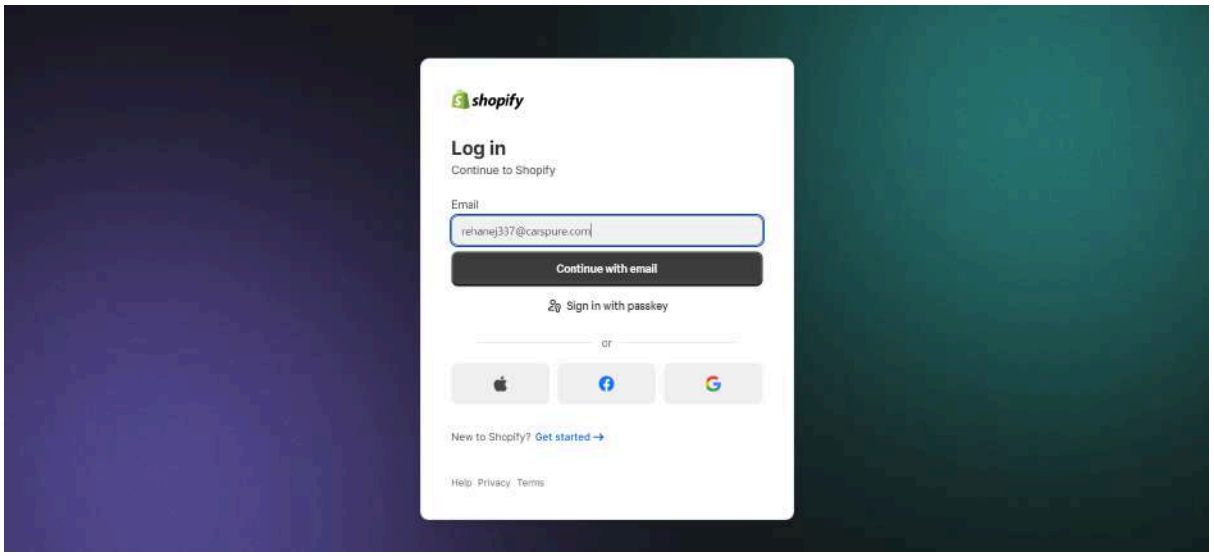
1.Create account on target (shopify)



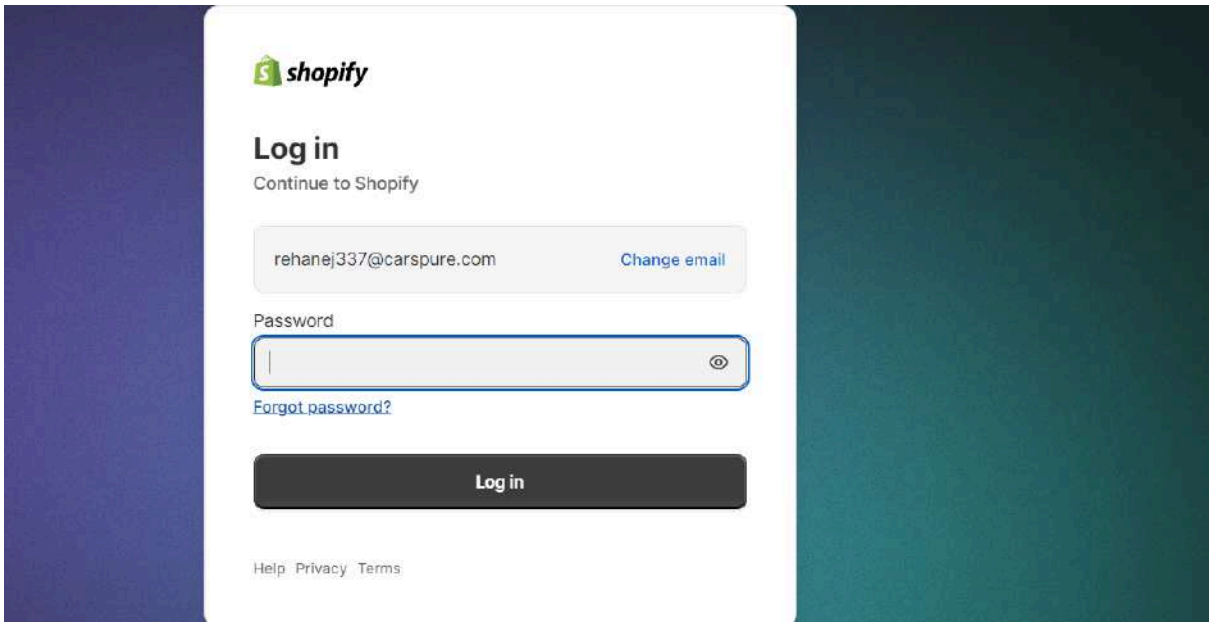
2.Log out



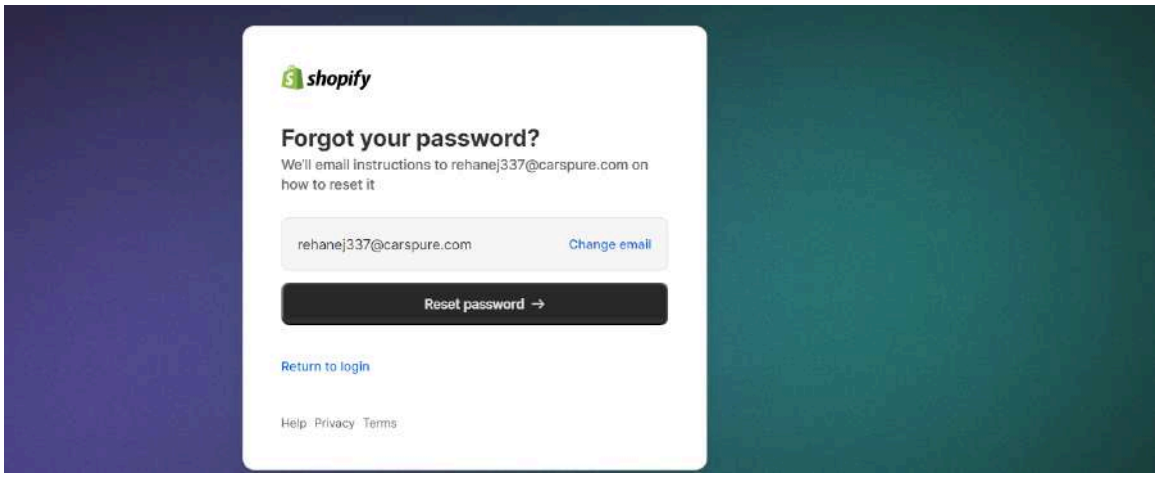
3.Log in again



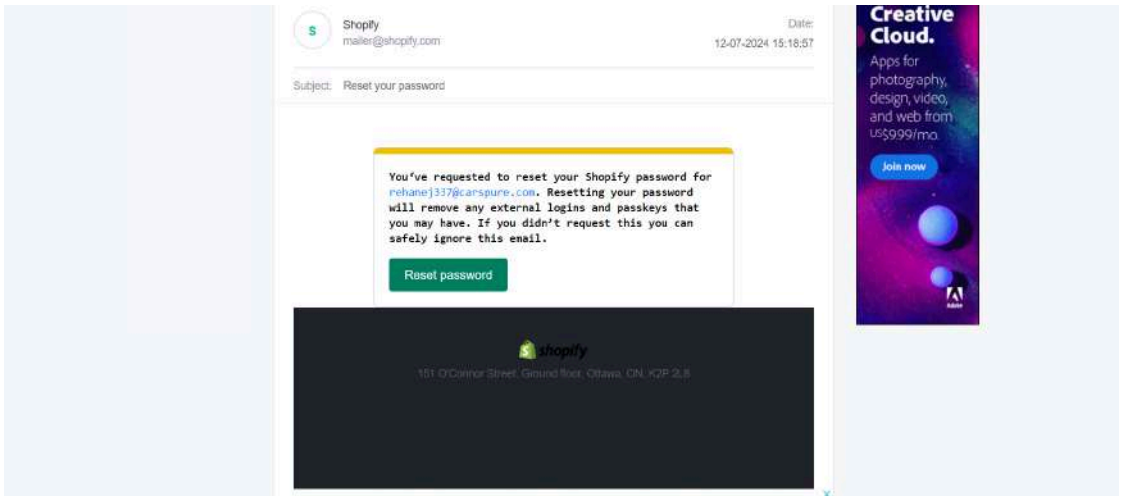
4.Click on forgot password



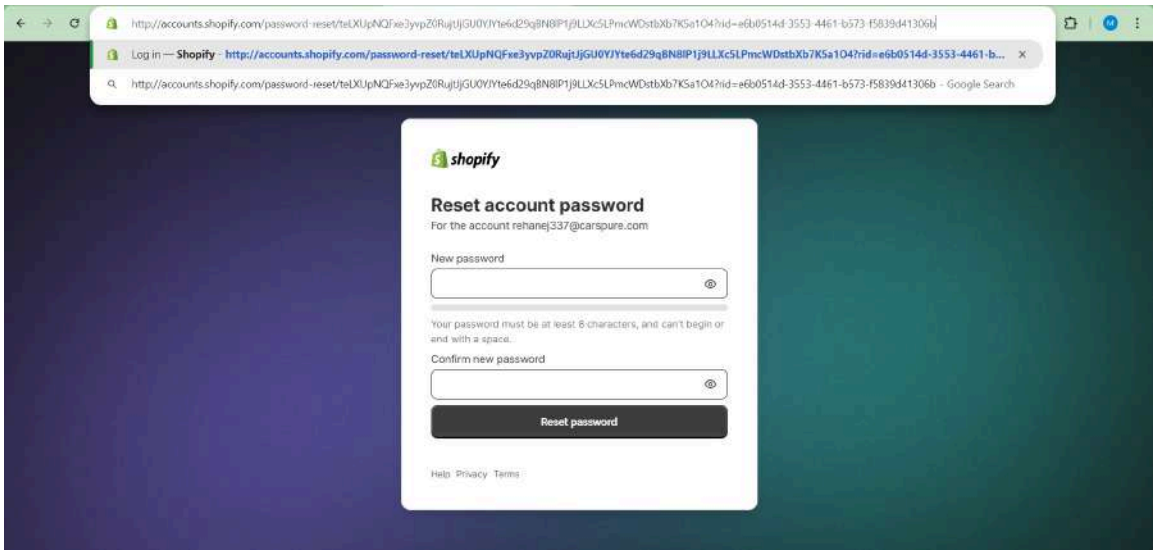
5.Click on reset password



6. Check for reset link in temp mail



7. Open link in new tab



8.Examine link --
http://accounts.shopify.com/password-reset/teLXUpNQFxe3yvpZ0RujtJjGU0YJYte6d29qBN8lP1j9LLXc5LPmcWDstbXb7K5a1O4?rid=e6b0514d-3553-4461-b573-f5839d41306b

---

B. Perform SQL Injection on given targets and dump the data from databases.

- https://www.lagnakaro.com/
- https://comand.edu.pk/

SQL Injection Vulnerability

- ❖ CVSS Score - 8.5 (High)
- ❖ Related OWASP Top 10 - A03:2021 - Injection
- ❖ Explanation: Explanation: SQL Injection is a code injection technique where malicious SQL statements are inserted into application queries to manipulate the database. This vulnerability occurs when user-supplied data is not properly validated, filtered, or sanitised before being included in SQL queries. Attackers can exploit this to bypass authentication, retrieve sensitive data, modify database contents, or even execute administrative operations on the database.
- ❖ Impact:

1. Data breach: Unauthorised access to sensitive information stored in the database.

2. Data manipulation: Attackers can add, modify, or delete data in the database.
3. Authentication bypass: Potential to log in as any user, including administrators.
4. Data loss: Possibility of deleting entire tables or databases.
5. Remote code execution: In some cases, SQL injection can lead to executing commands on the host operating system.

❖ Recommendations:

1. Use parameterized queries: Implement prepared statements with parameterized queries.
2. Input validation: Validate and sanitise all user inputs before using them in SQL queries.
3. Least privilege principle: Use database accounts with minimal necessary privileges for the application.
4. Stored procedures: Use safely parameterized stored procedures for database access.
5. Escaping: If dynamic SQL is necessary, use proper escaping techniques for special characters.
6. ORM frameworks: Utilise Object-Relational Mapping (ORM) frameworks that handle SQL securely.

❖ References:

1. OWASP SQL Injection Prevention

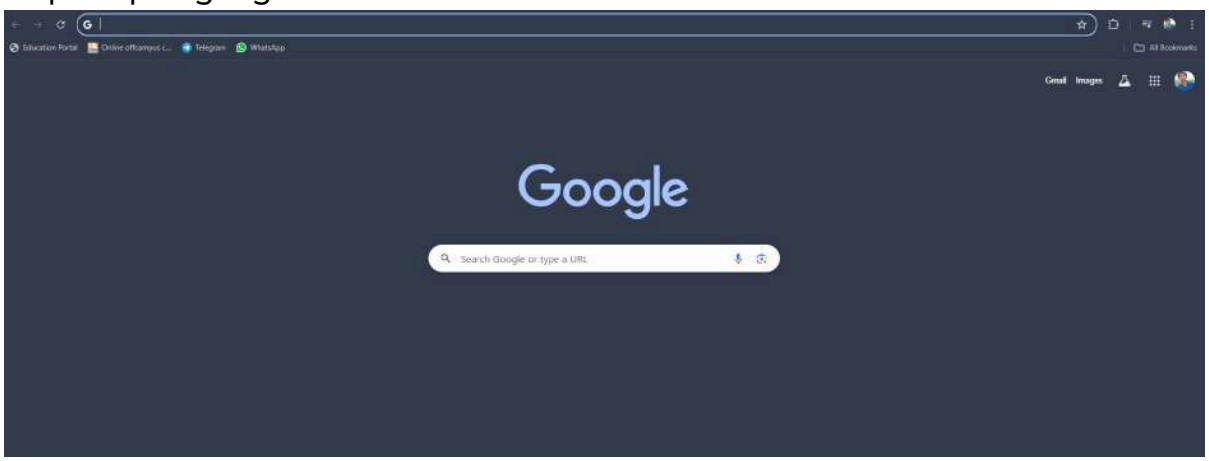
❖ Procedure:

➢ Target Website - https://www.lagnakaro.com/ and https://comand.edu.pk/,

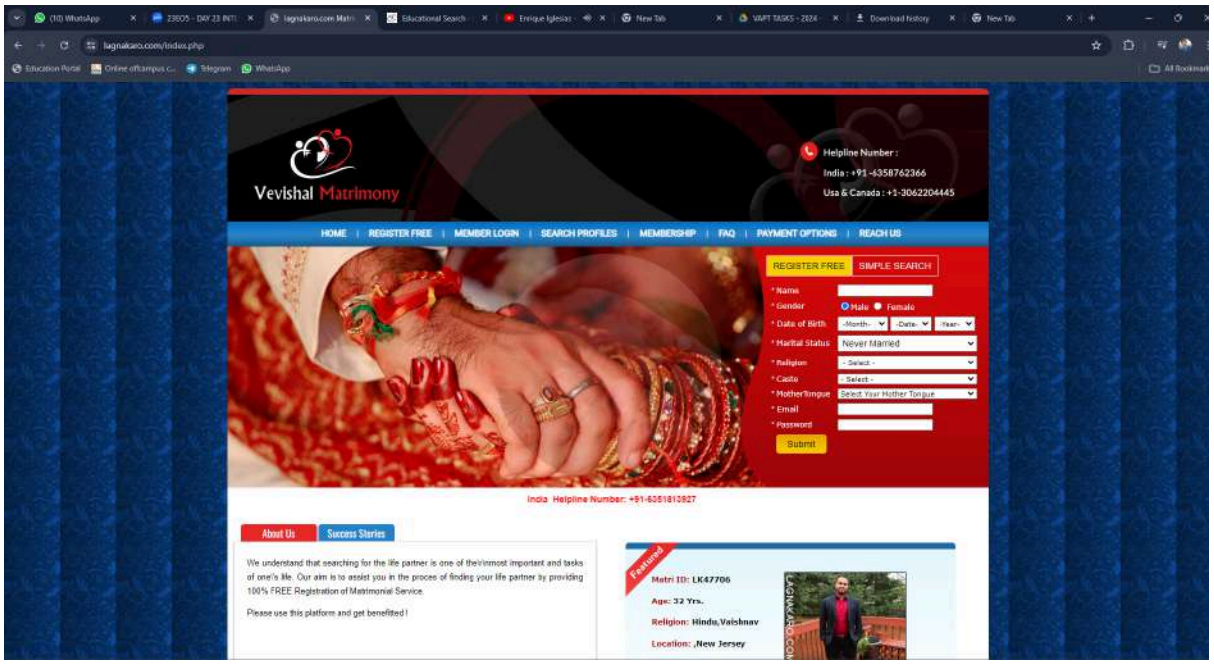
➢ Payload - Host: www.facebook.com

➢ Steps -

Step-1: Open google chrome.



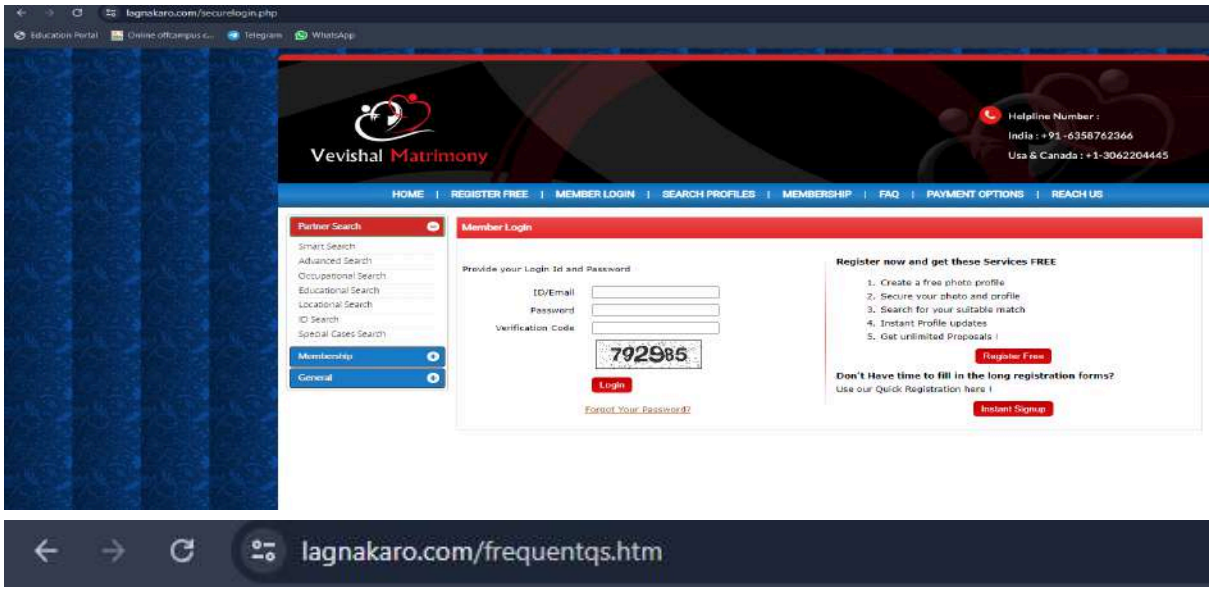
Step-2: Go to the target website.(https://www.lagnakaro.com/)

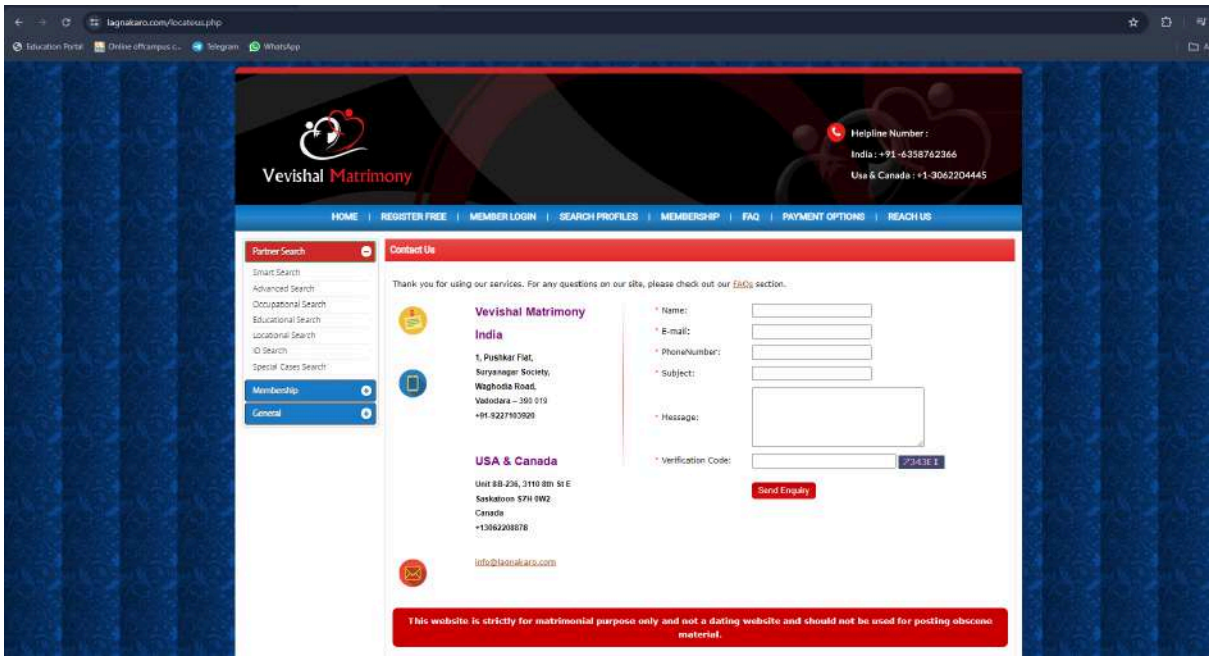


Step-3: Find a entry point in the website
→An entry point in a website can be determined by exploring the different sections and options in the website and finding a url that contains a number at the end of it.
→ example:

→The above url contains ‘50’ at the end which indicates an entry point.

→ For the target website(https://www.lagnakaro.com/):

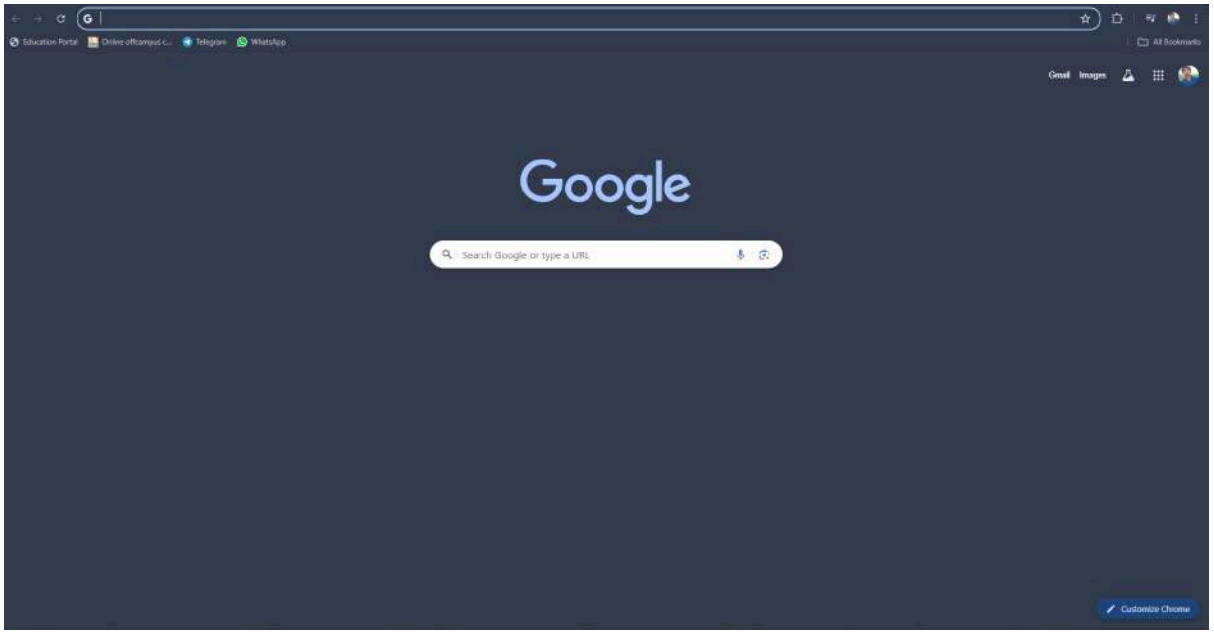




In the target website there was no entry point present as the website was not prone to the vulnerability. Sql injection cannot be performed on it.

TARGET-2(https://comand.edu.pk/)

Step-1: Open google chrome



Step-2: Go to the target website(https://comand.edu.pk/)

Result: Same as the previous website there are no entry points in the website hence sql injection cannot be performed on it.

C. Find a website vulnerable to Business Logic Errors on each test case below.
   a. Currency Arbitrage
   b. Delivery Charges Abuse

Business Logic Errors Vulnerability

- CVSS Score - 6.5 (Medium)
- Related OWASP Top 10 - A04:2021 - Insecure Design
- Explanation: Business Logic Errors are flaws in the design and implementation of an application that allow attackers to manipulate legitimate functionality in unintended ways. These vulnerabilities occur when the application's logic fails to properly enforce the intended business rules, allowing users to perform actions that should be restricted or to bypass critical steps in a process. Unlike technical vulnerabilities, business logic errors often require a deep understanding of the application's intended functionality and can be challenging to detect through automated scanning.
- Impact:

1. Financial loss: Exploitation of pricing errors, discount manipulations, or transaction flaws.
2. Data integrity issues: Unauthorized data modifications that bypass intended workflows.
3. Privilege escalation: Gaining access to functionality or data reserved for higher-privileged users.
4. Regulatory non-compliance: Violation of required business processes or data handling procedures.

- Recommendations:

1. Thorough design review: Conduct comprehensive reviews of business logic during the design phase.
2. Implement proper access controls: Ensure that users can only access functions appropriate to their role.
3. Input validation: Validate all inputs, including those that affect business logic decisions.
4. Server-side validation: Implement all critical checks and validations on the server-side.
5. Secure workflow design: Ensure that multi-step processes cannot be bypassed or performed out of order.

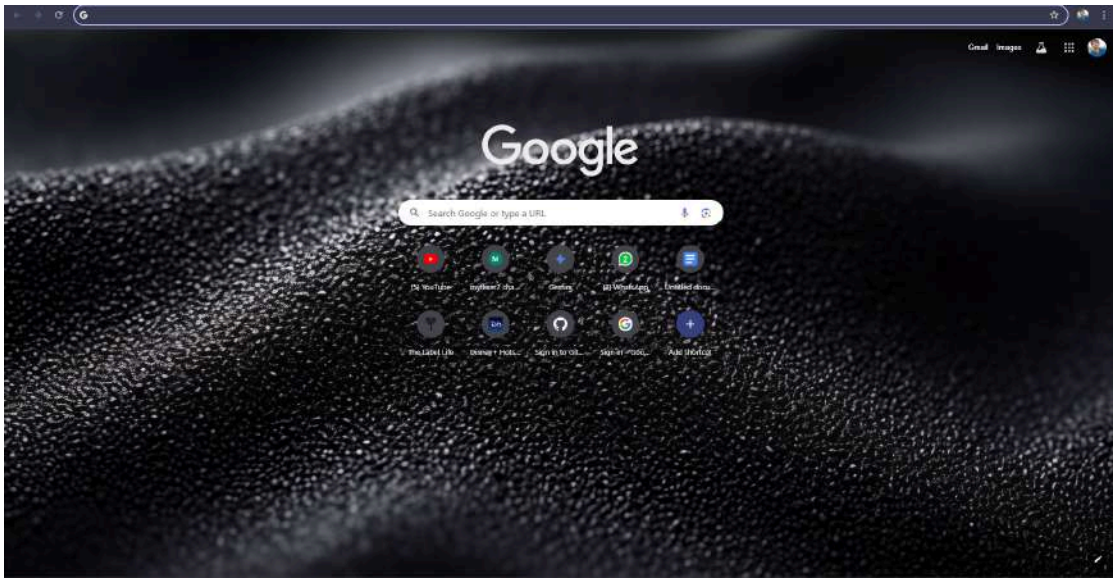
- References:

1. Business Logic Security by OWASP

- Procedure:
    - Target Website - https://store.thelabellife.com/
    - Steps -

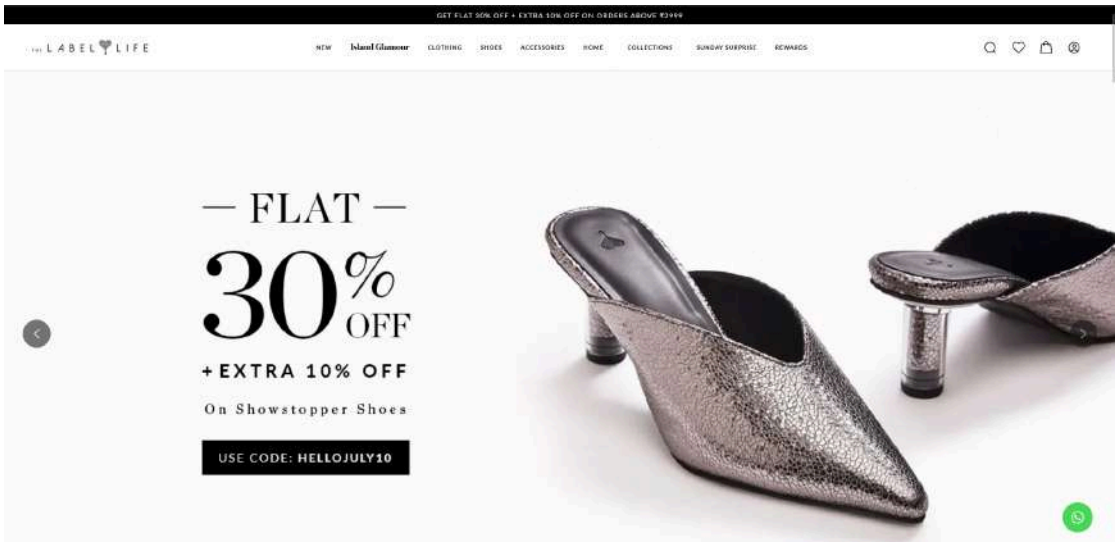
- PROCEDURE

Step-1: Open google chrome

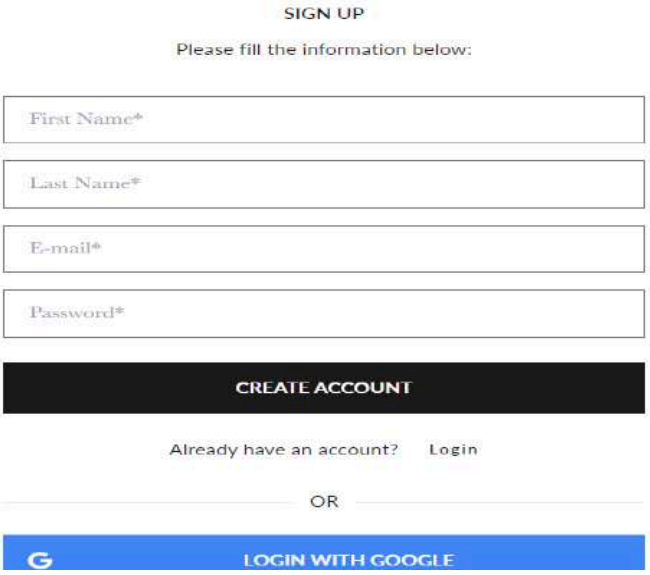


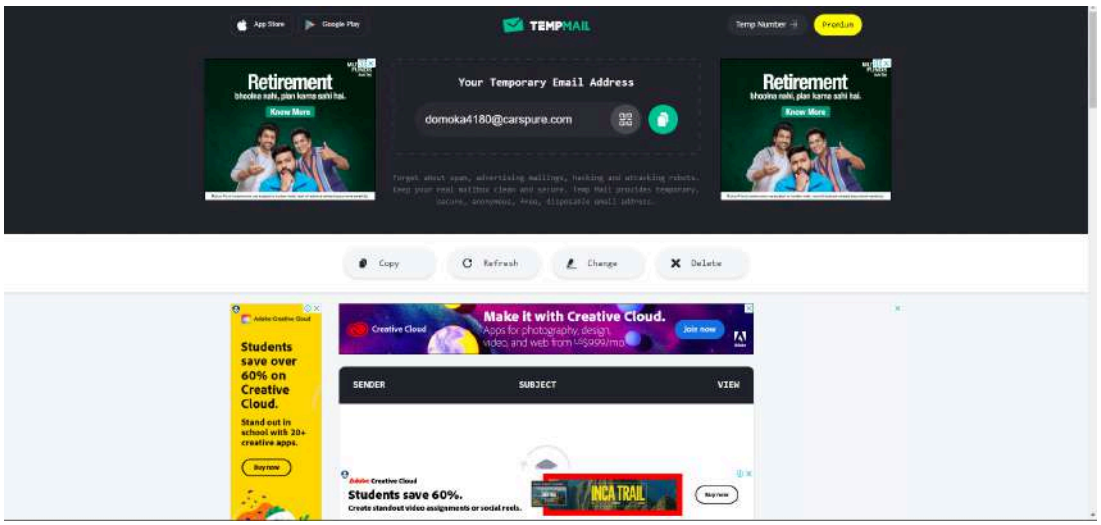
Step-2: Use the google dork→"inurl: Responsible disclosure"
to get a vulnerable website.

Step-3:This practical is performed on the website https://store.thelabellife.com/. Open firefox browser and go to target website.

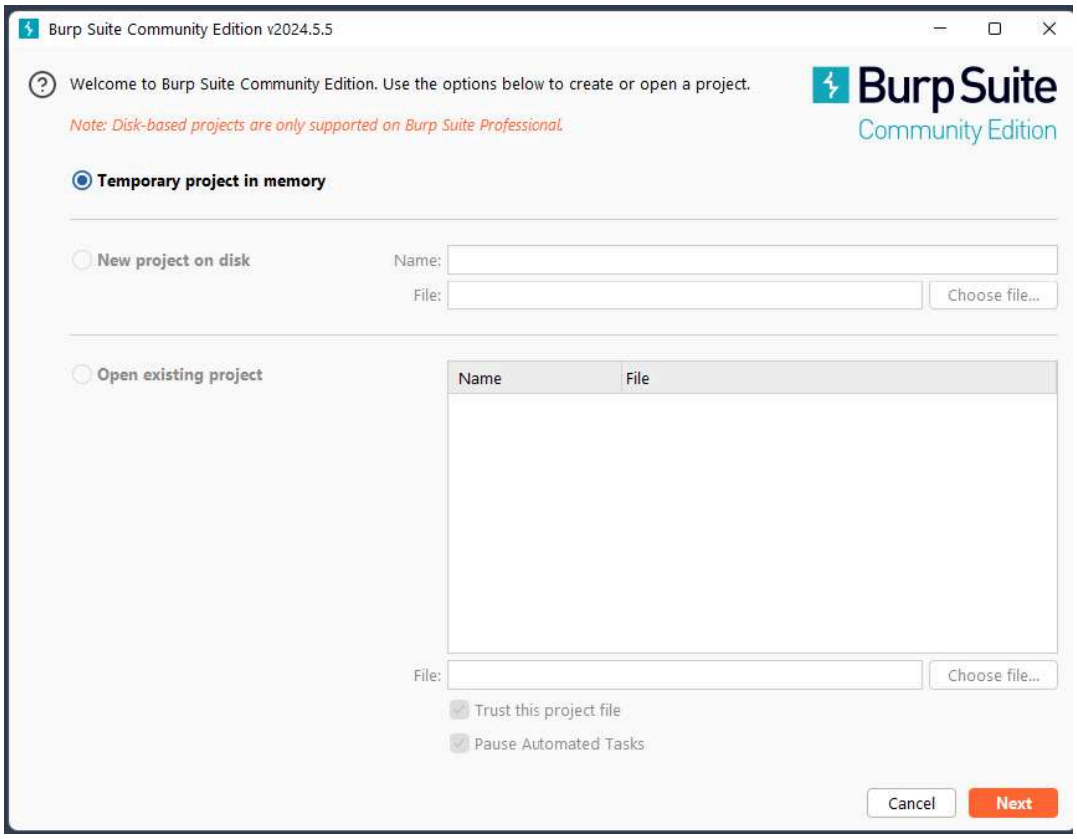


Step-4: Sign in on the website using fake credentials. For obtaining a fake mail id https://temp-mail.org/en/ can be used.

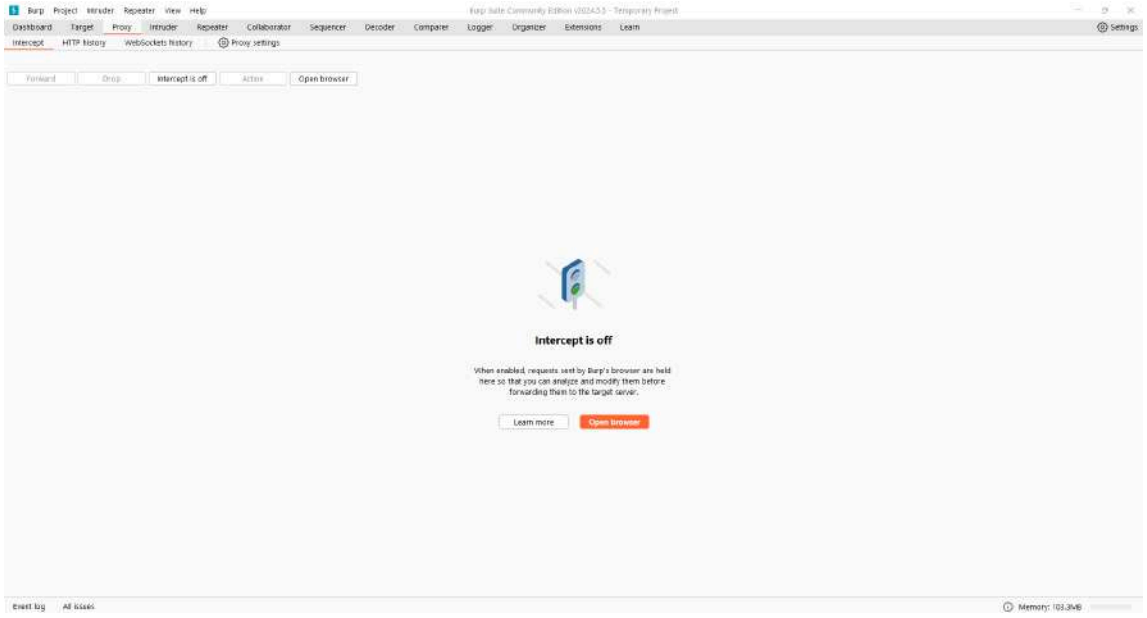




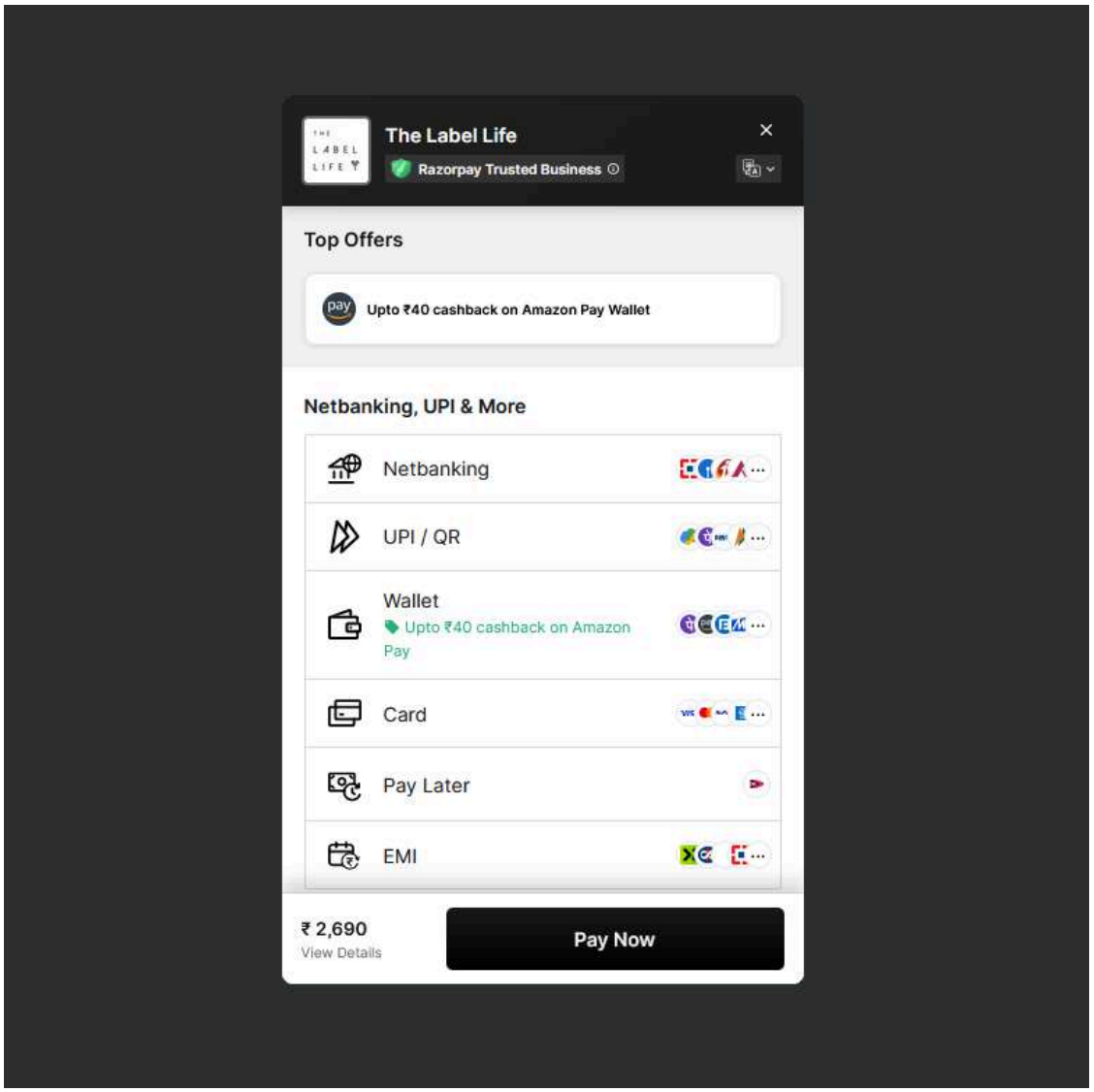
Step-5: Now launch burp suite.



Step-6:Go to the proxy section of the burp suite.

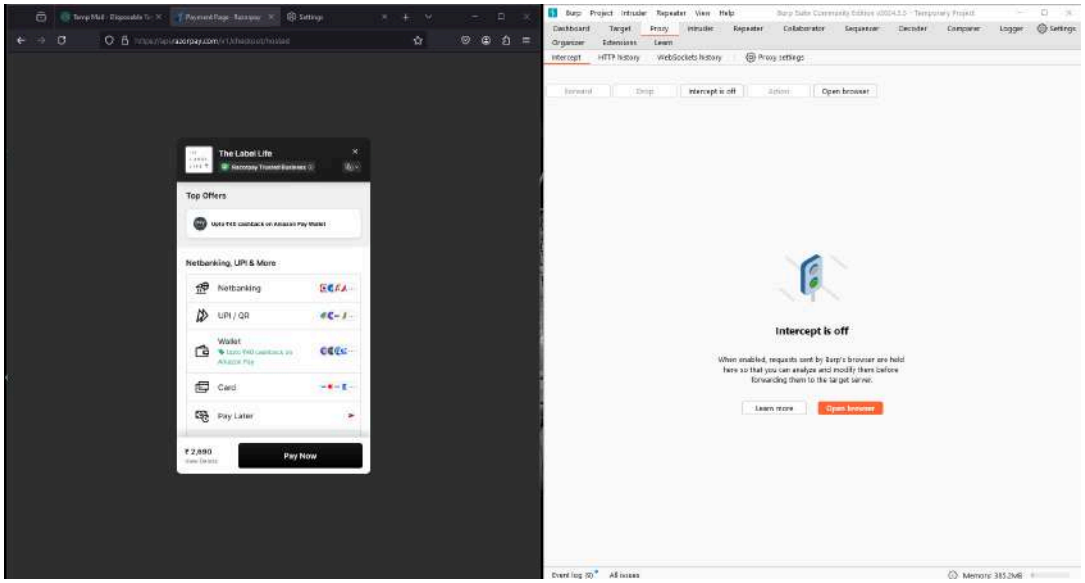


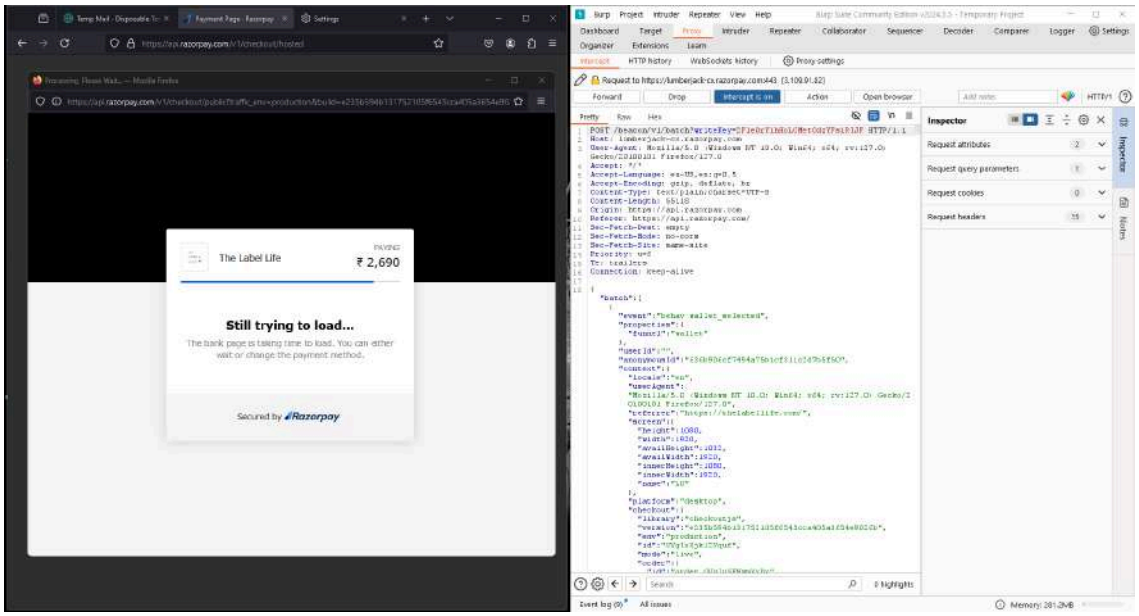
Step-7: Now, in the target add any product in the cart and proceed to the payment page.



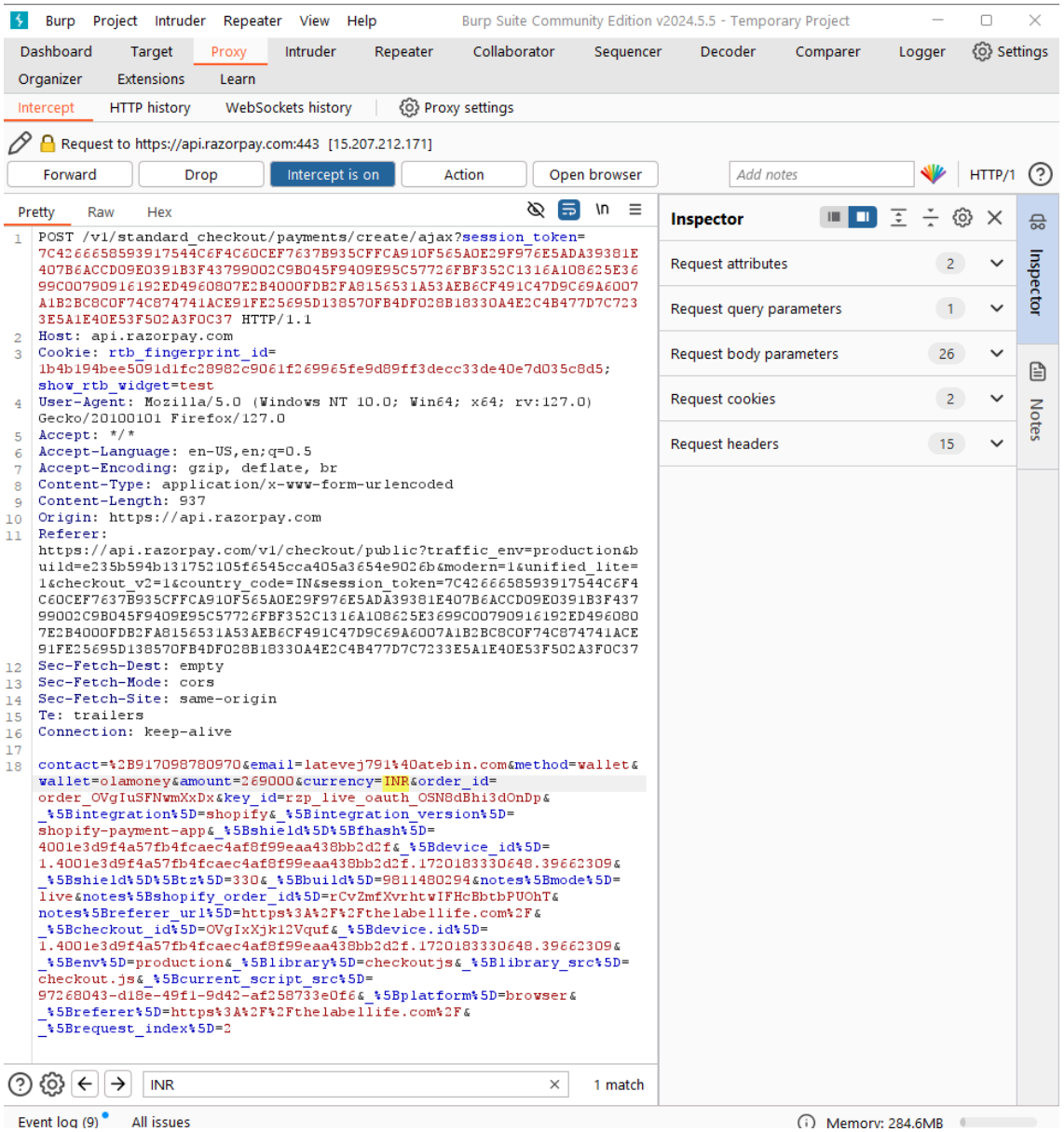
Here the cart value is Rs. 2690.

Step-8: Before proceeding to the payment gateway turn on the intercept in burp suite then go to payment gateway.





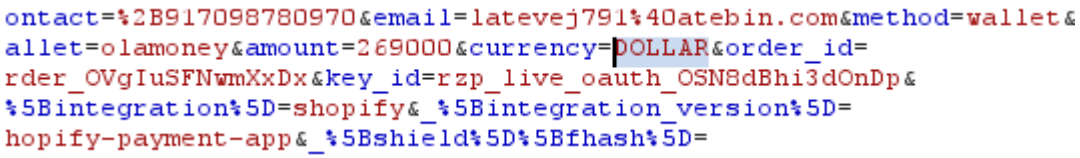
Step-9: Burp suite will start showing entries.



Step-10: For currency arbitrage find the payment currency in the data shown in burp suite.

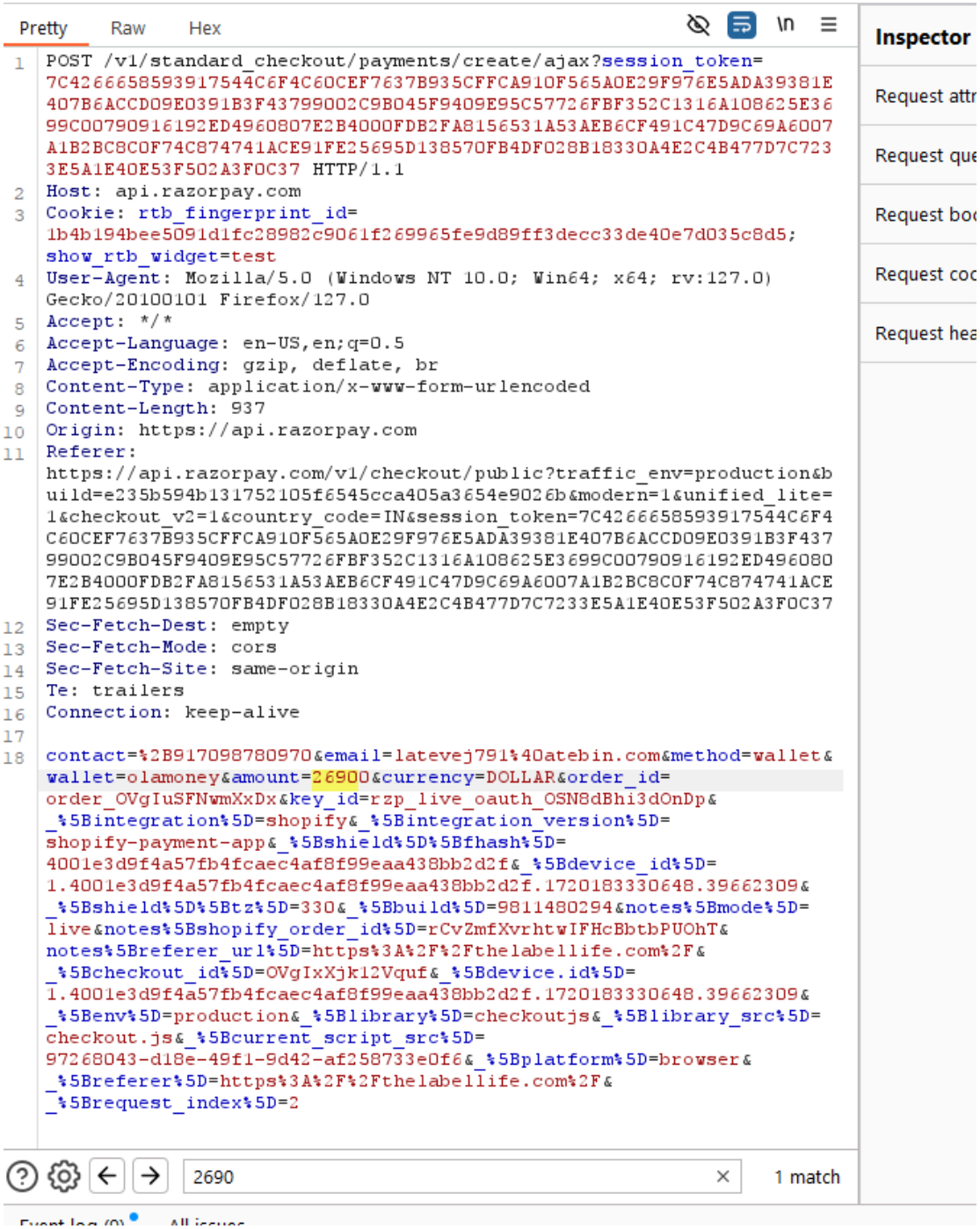
Here the selected part is the currency which is in Indian Rupees (INR).

Step-11: Alter the currency to any other suitable currency.



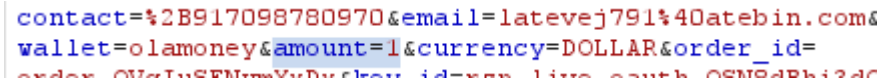
Here, it is changed to Dollars.

Step-12: For delivery charges abuse look for the amount that had to be paid originally.



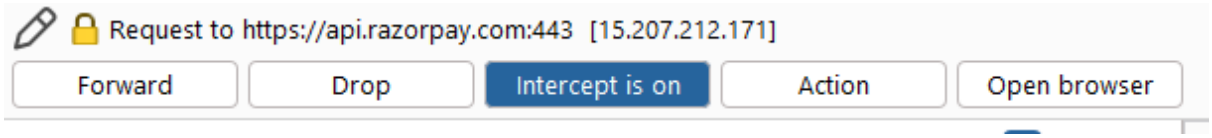
Here it is 2690 as highlighted.

Step-13: Altered it to any amount suitable.



Here it is changed to 1.

Step-14: After the arbitrage and charge abuse is done click on “intercept is on” to turn off the intercept.



CONCLUSION

→ Burp suite is a strong tool to perform currency arbitrage and charge abuse. In fact, it can be used to perform otp bypassing.

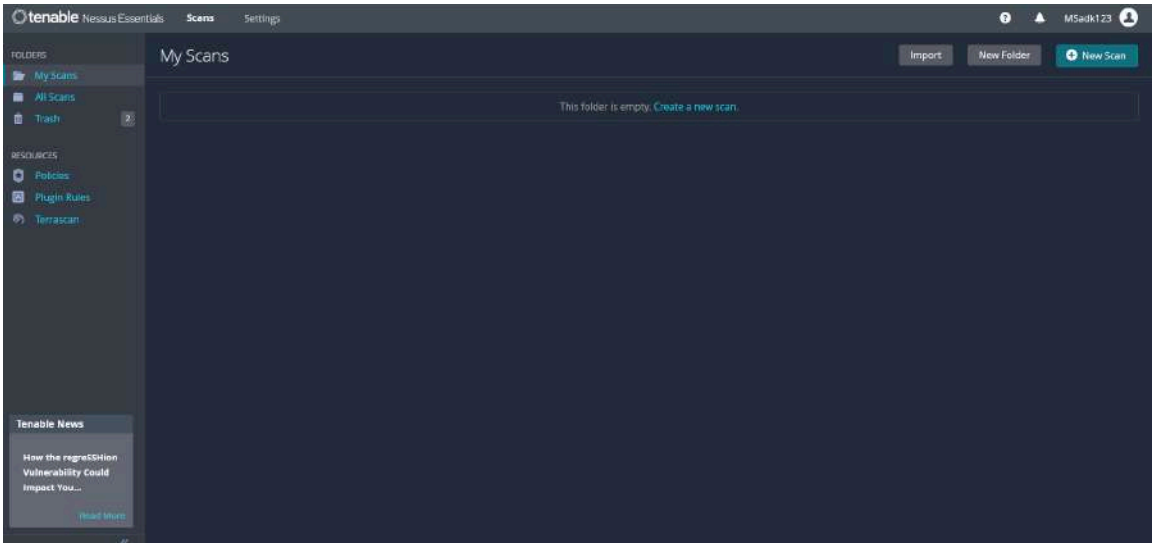
Supraja Technologies

23E05-ST#IS#6653-TASK9

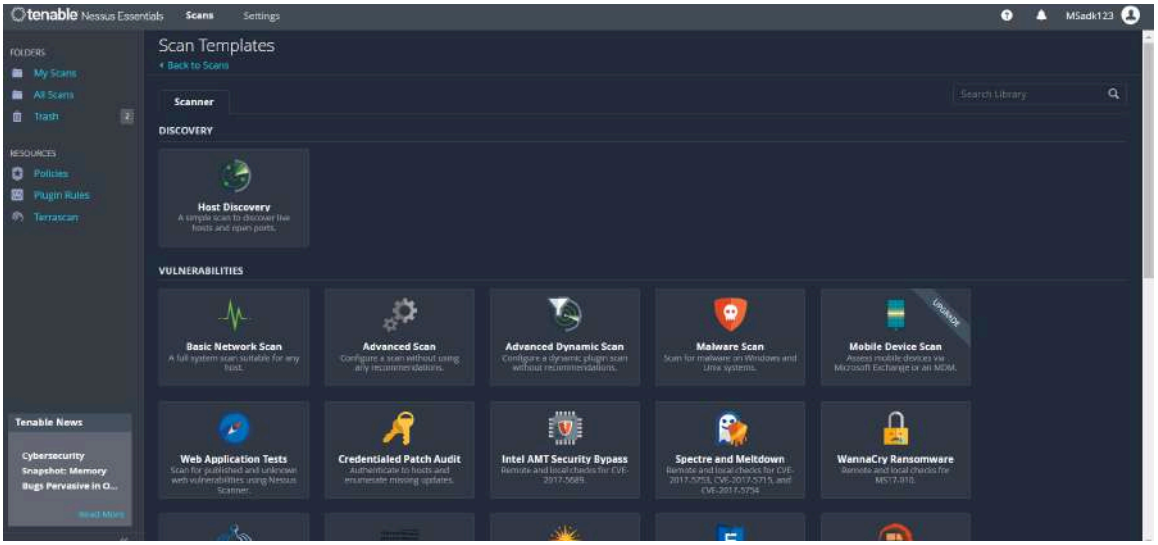
A. Perform different scans on your network using the Nessus tool and generate a report.

a) Host Discovery Scan

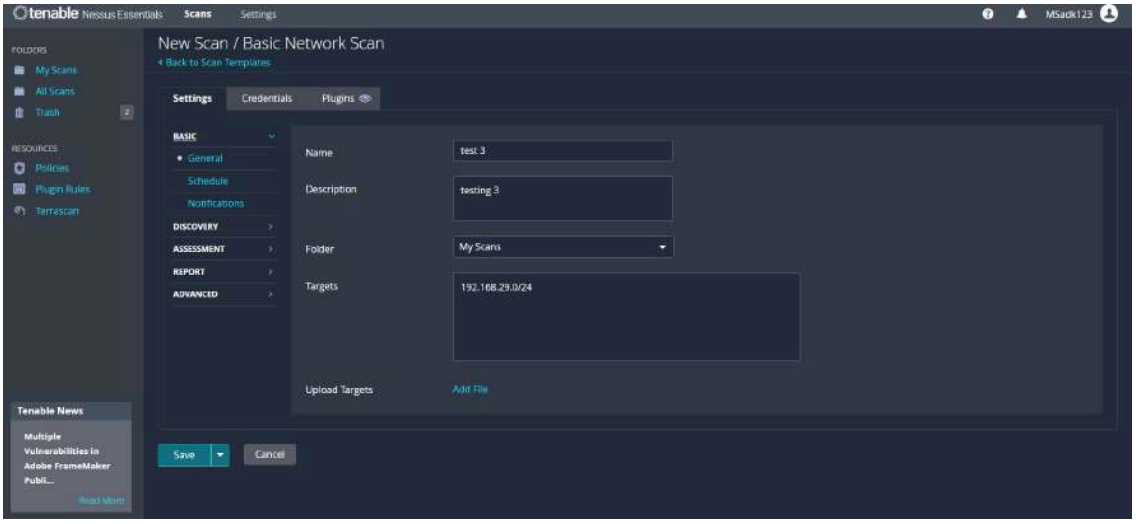
1. Open Nessus on your Machine



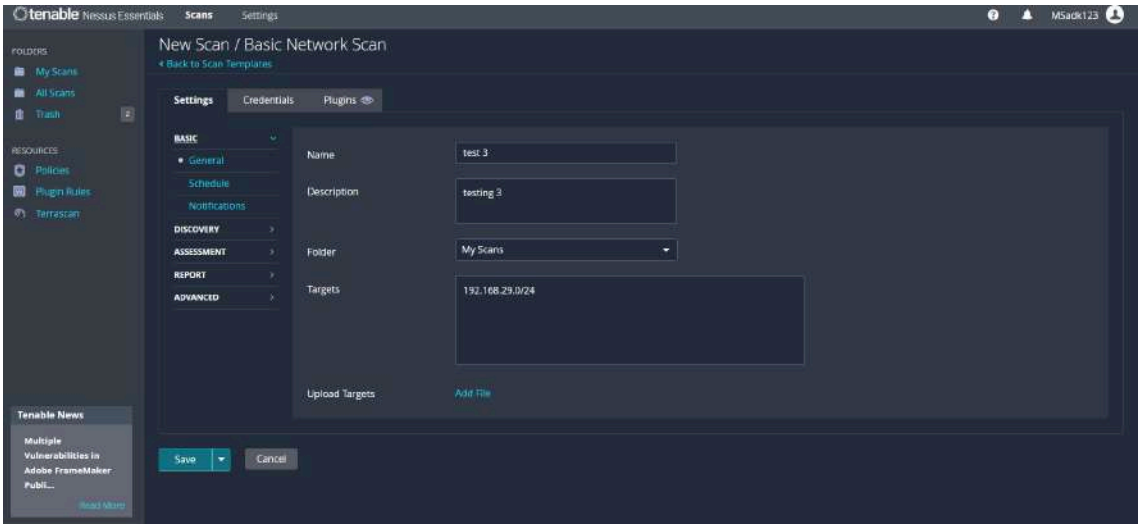
2. Click on new scan



3. Click on host discovery and enter details

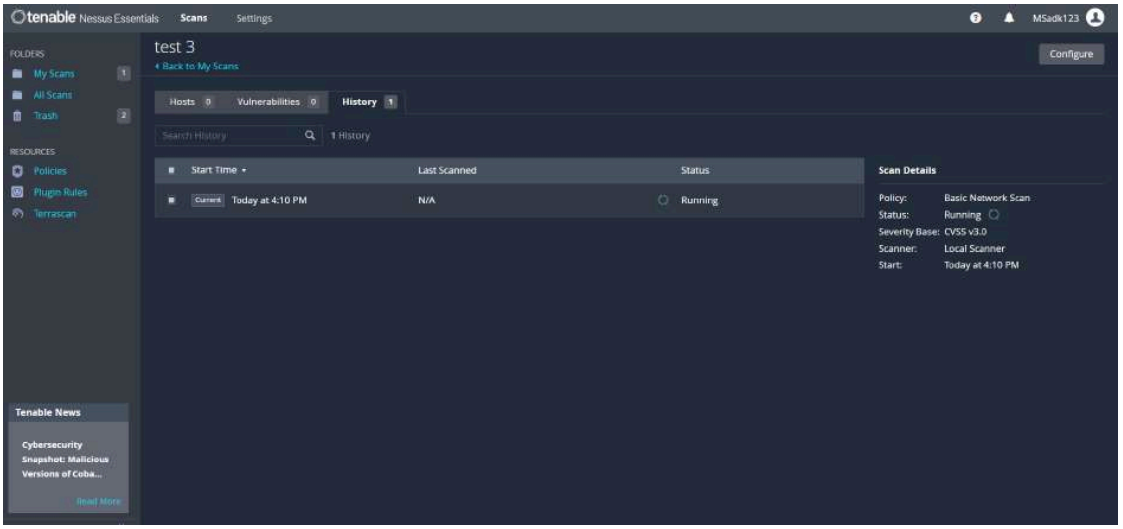


4. Click on save

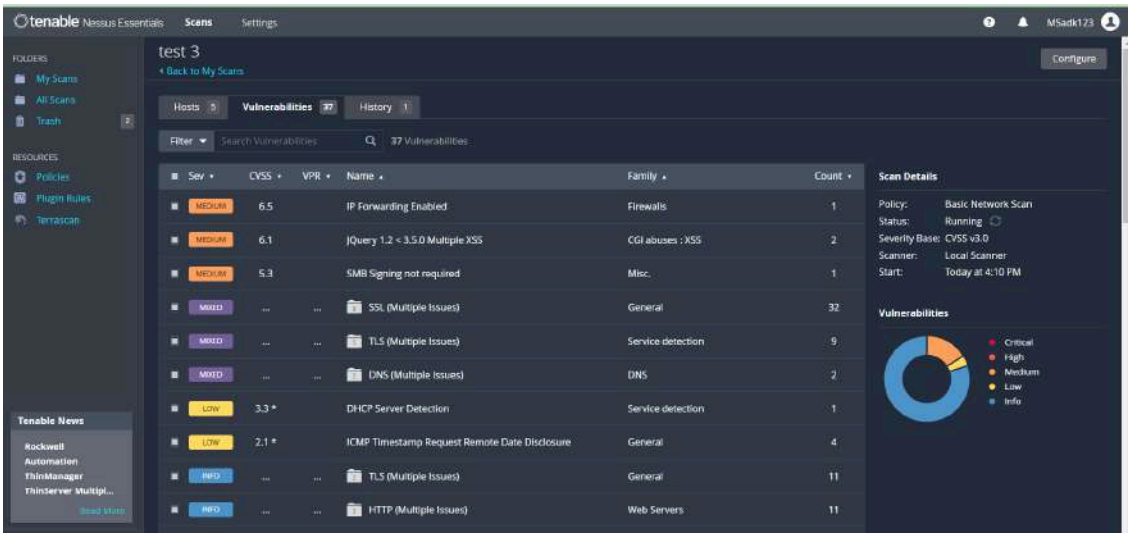


5. Click on launch

6. Scan started

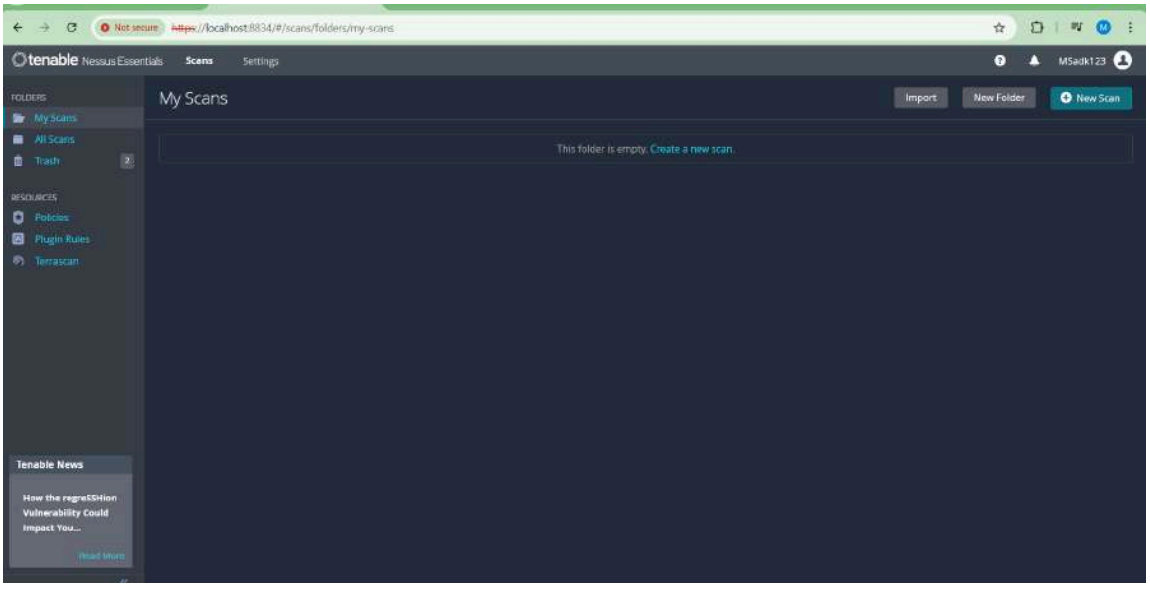


7. Scan completed. Here is the result

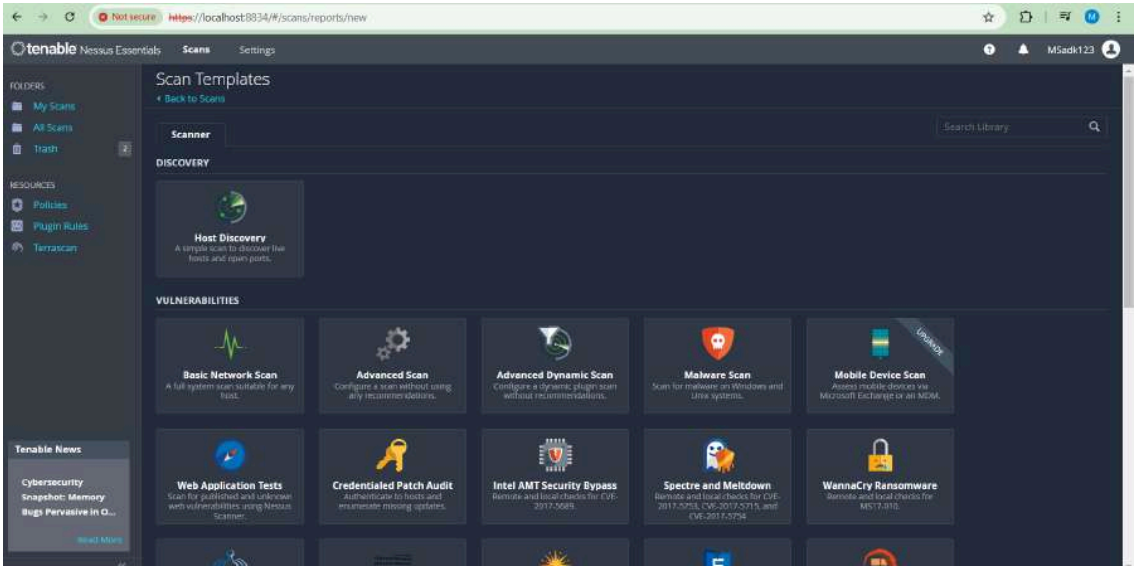


b) Basic Network Scan

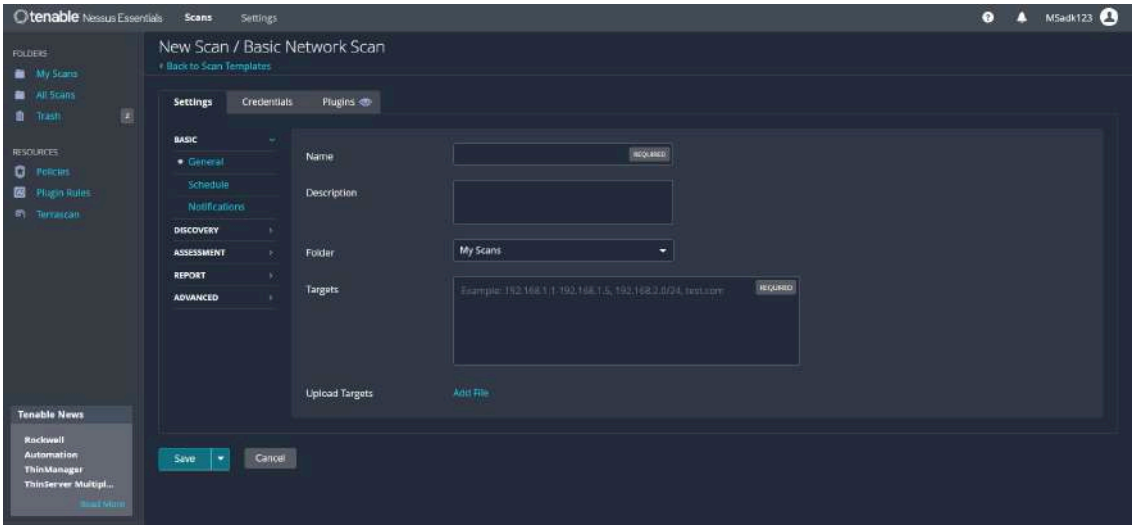
1. Open nessus



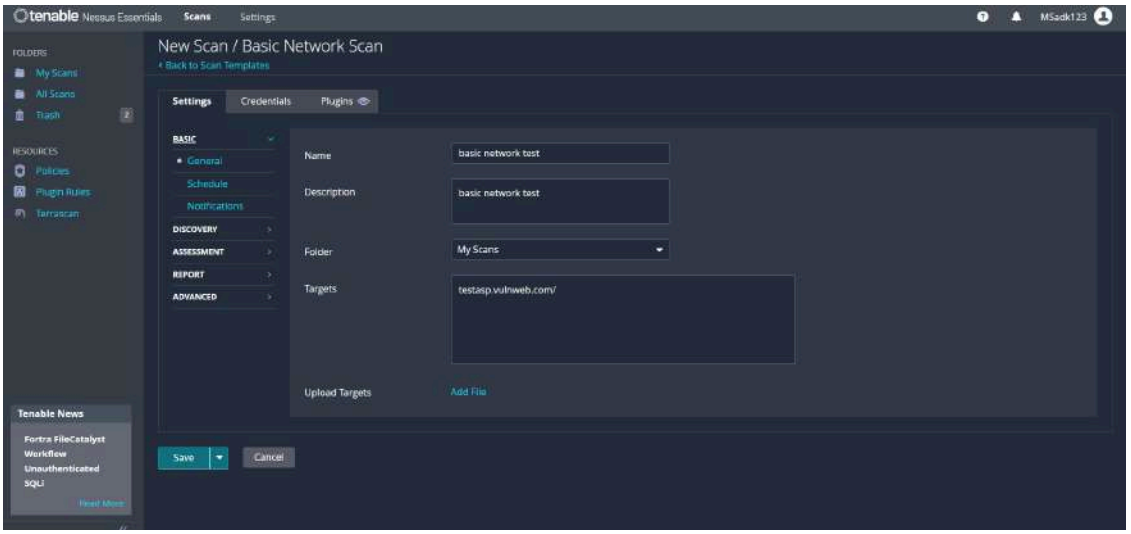
2. Click on new scan



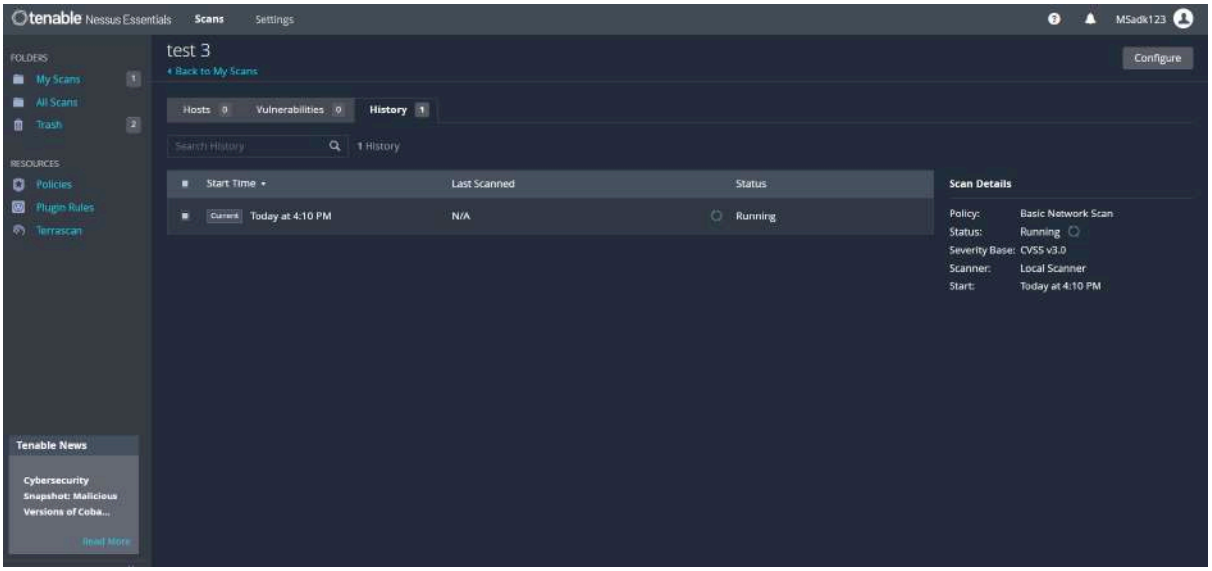
3. Click on Basic network scan



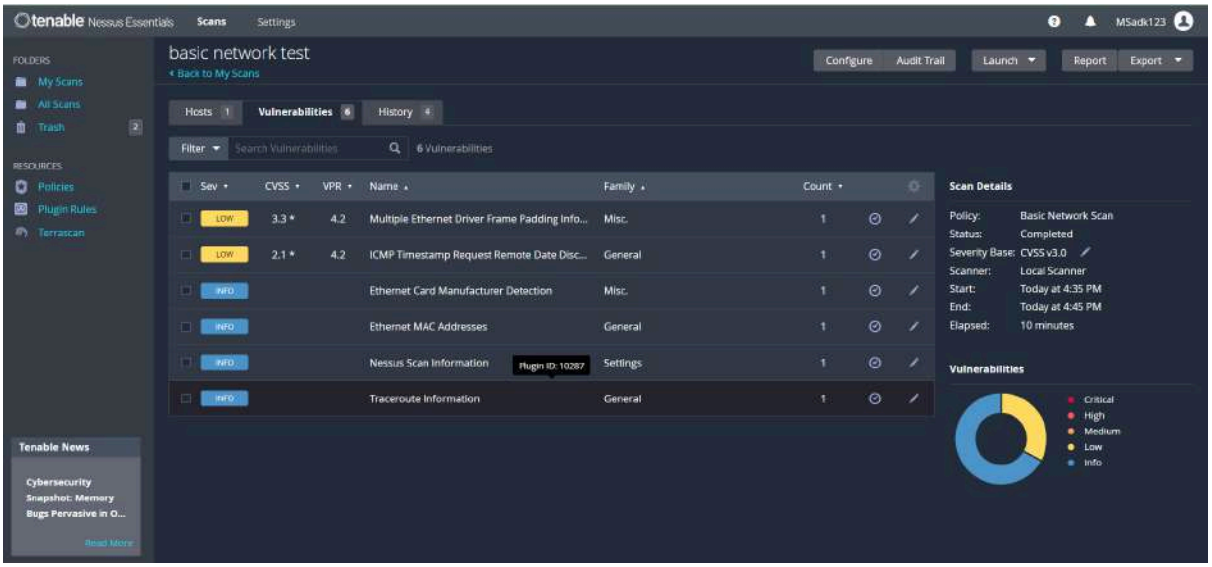
4. Enter target details



5. Click on save and launch , scan started



6. Scan finished, here is the result



<u>CONCLUSION:</u>

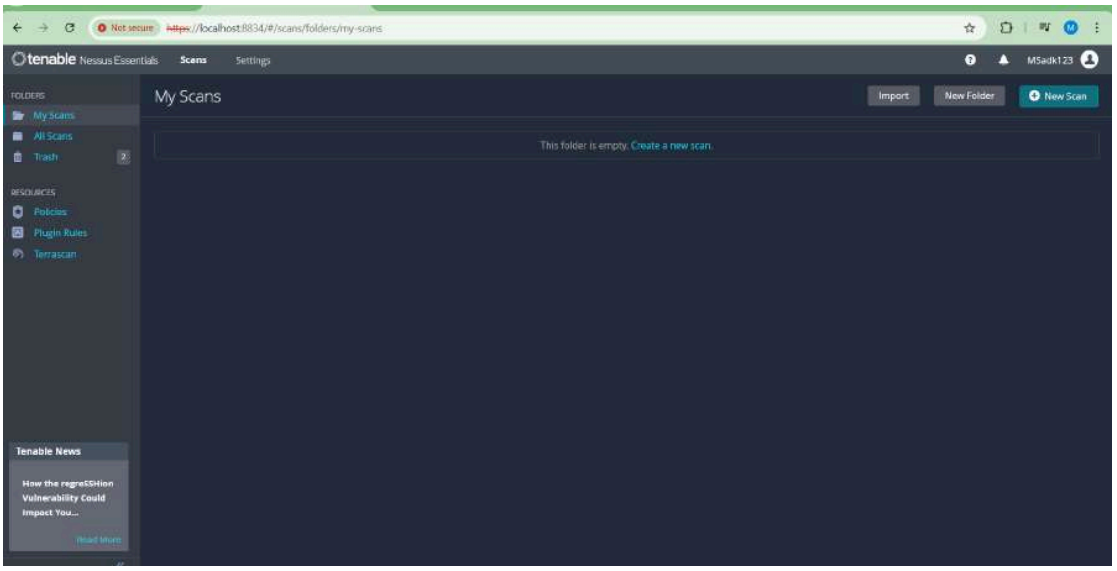
Hence we were able to do the host discovery scan and the basic network scan to find the vulnerabilities in the host's network using the Nessus tool.

B. Perform Web Application Tests Scan in the Nessus tool on the below targets:

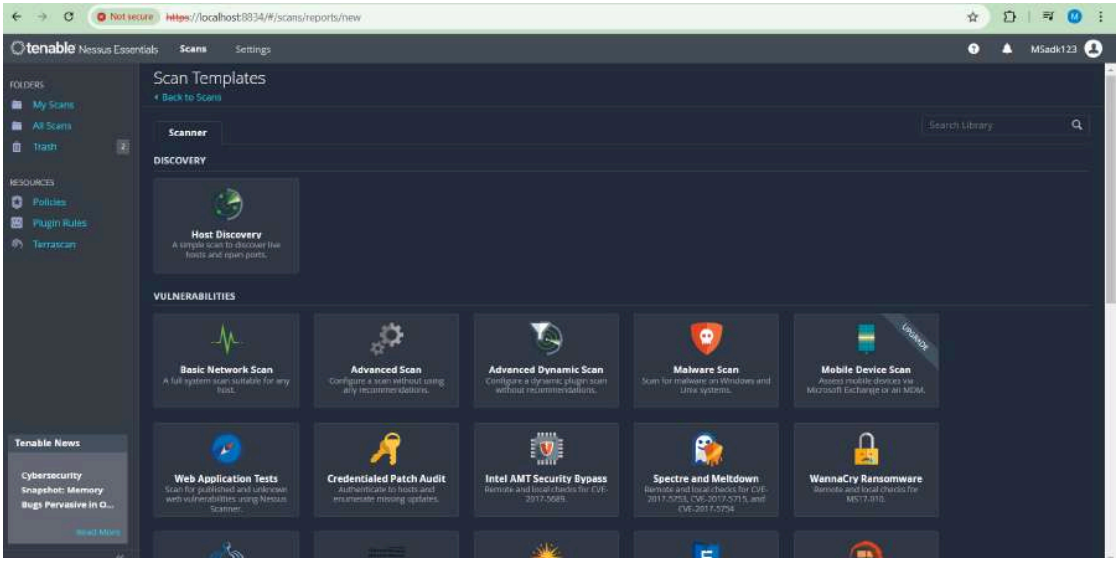
a) http://testasp.vulnweb.com/

b) https://www.shoppersstop.com/

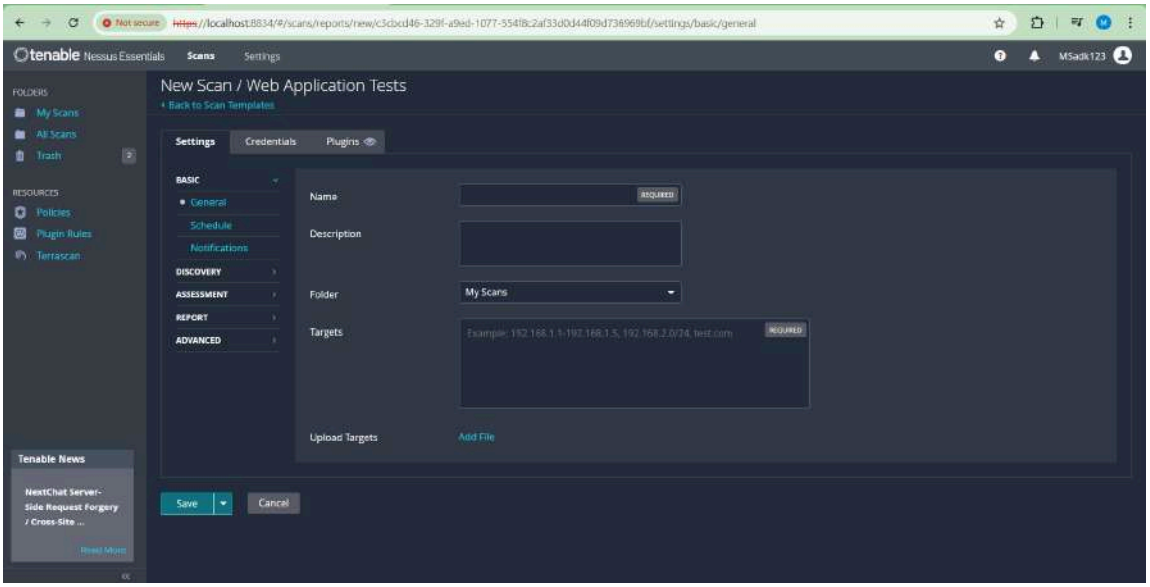
1. Open nessus



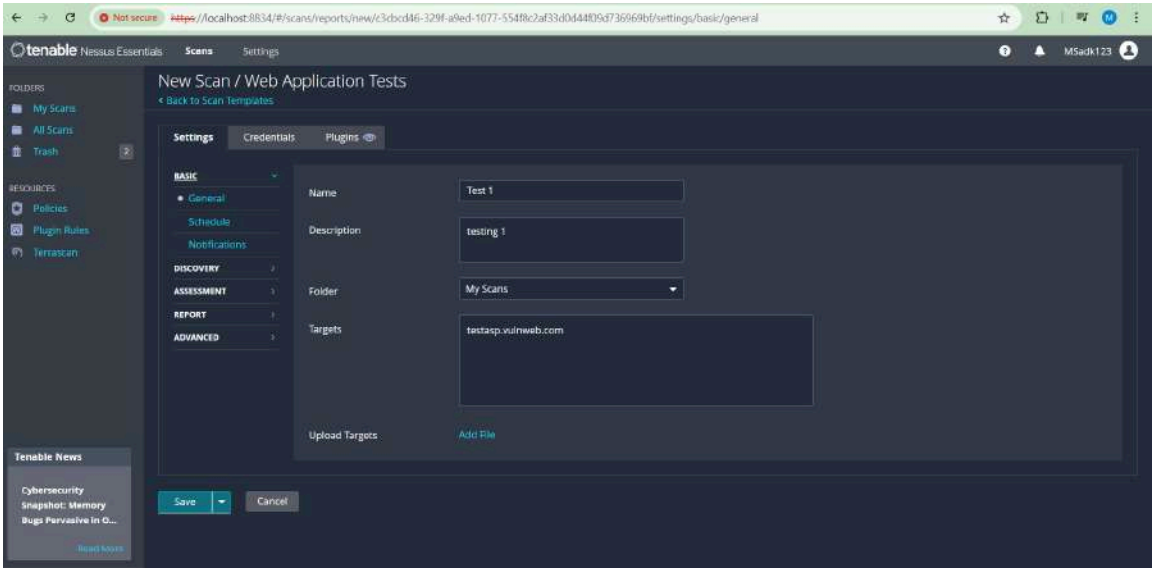
2. Click on new scan



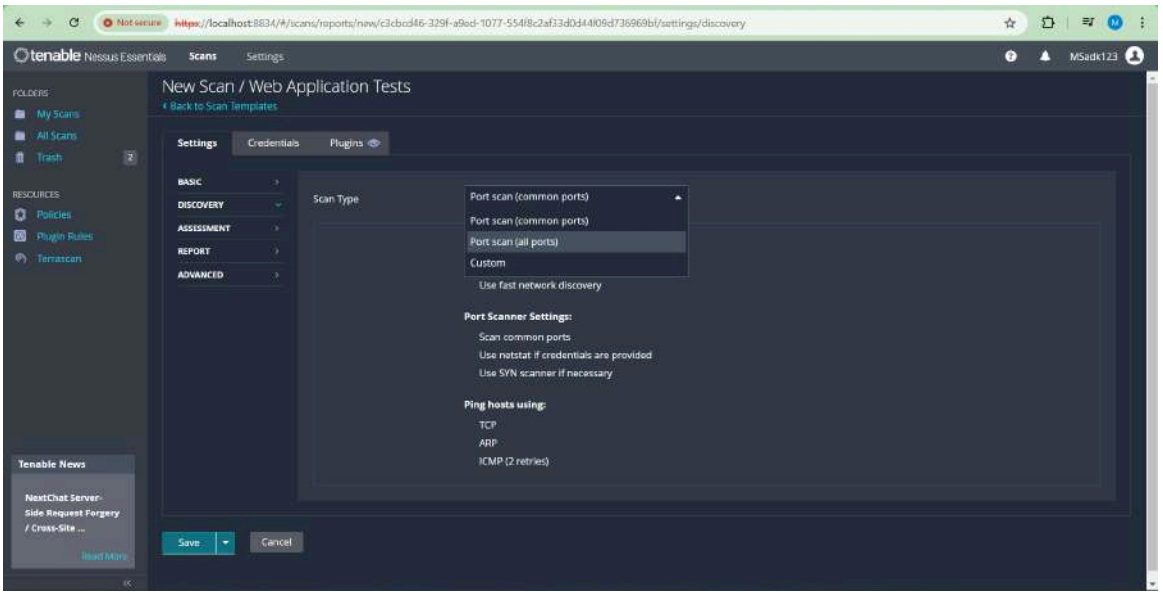
3. Click on web application tests



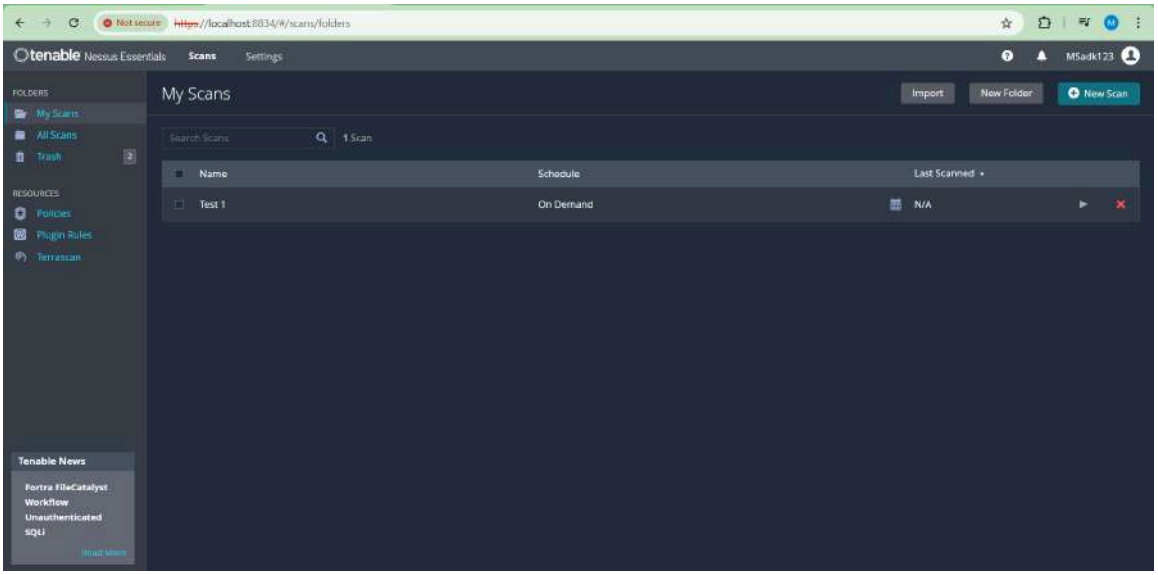
4. Enter the details of the target



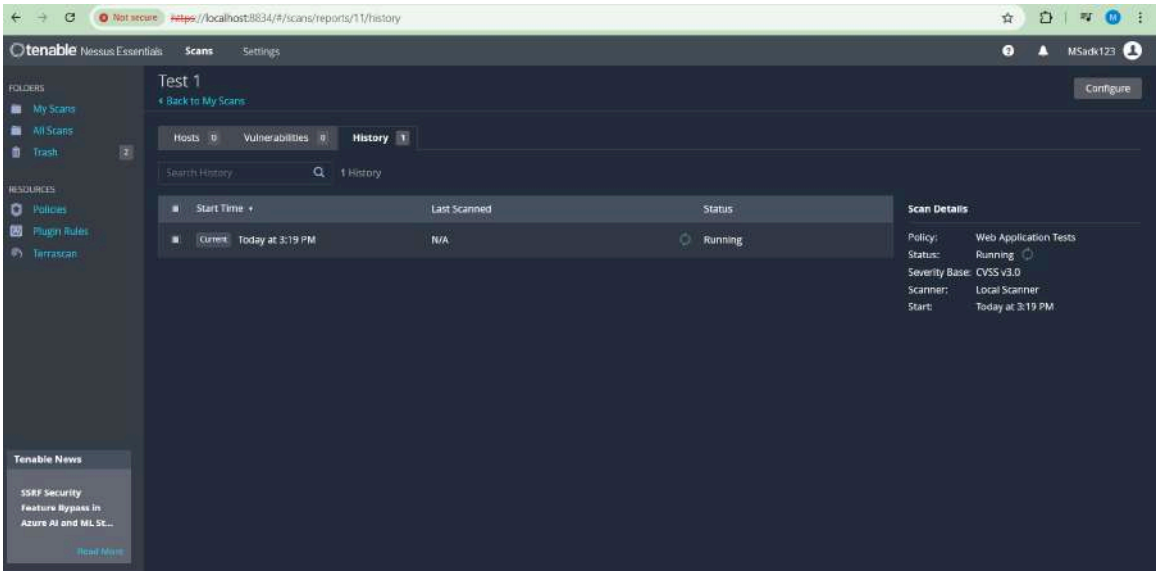
5. In discovery section change the scan type to all ports



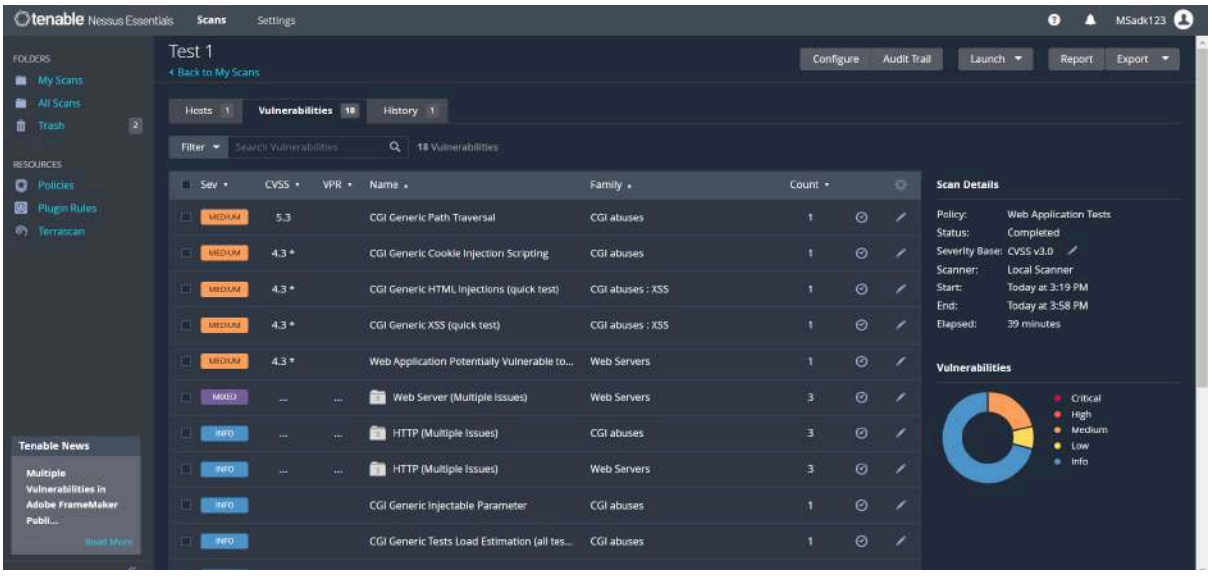
6. Save the scan



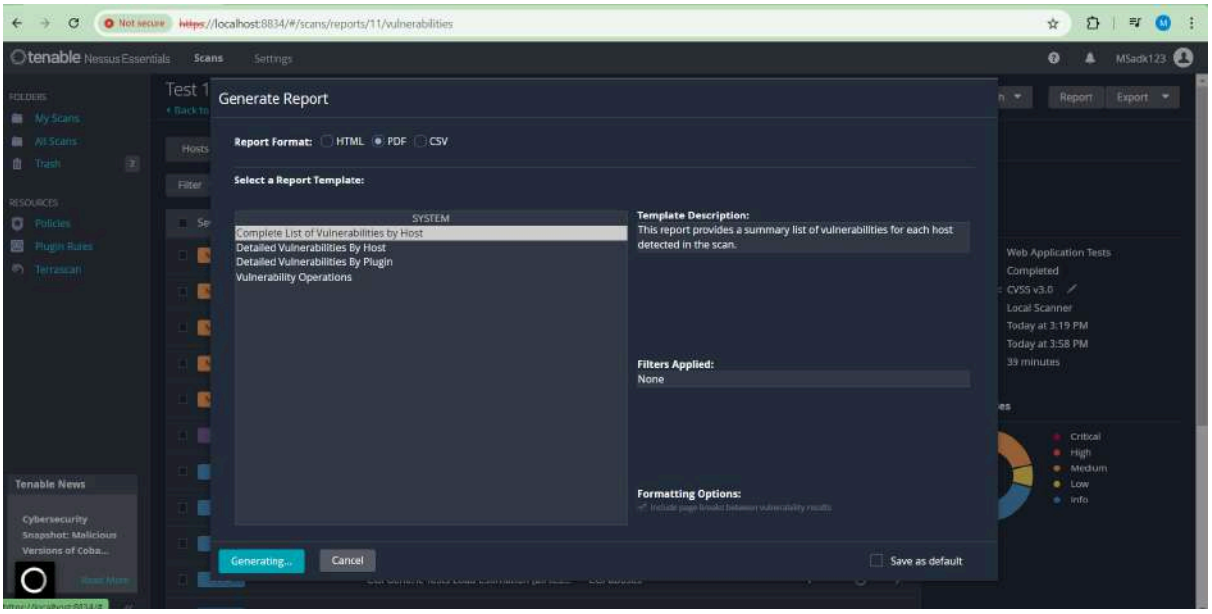
7. Click on the play button, the scan will be launched



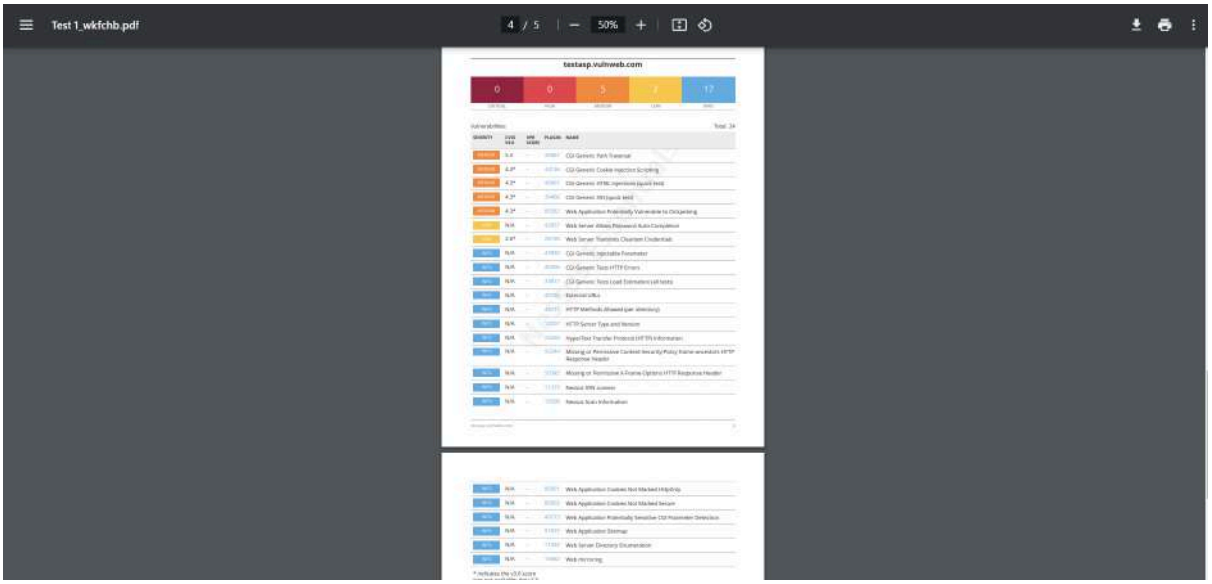
8. Scanning for test 1 was completed , here is the result



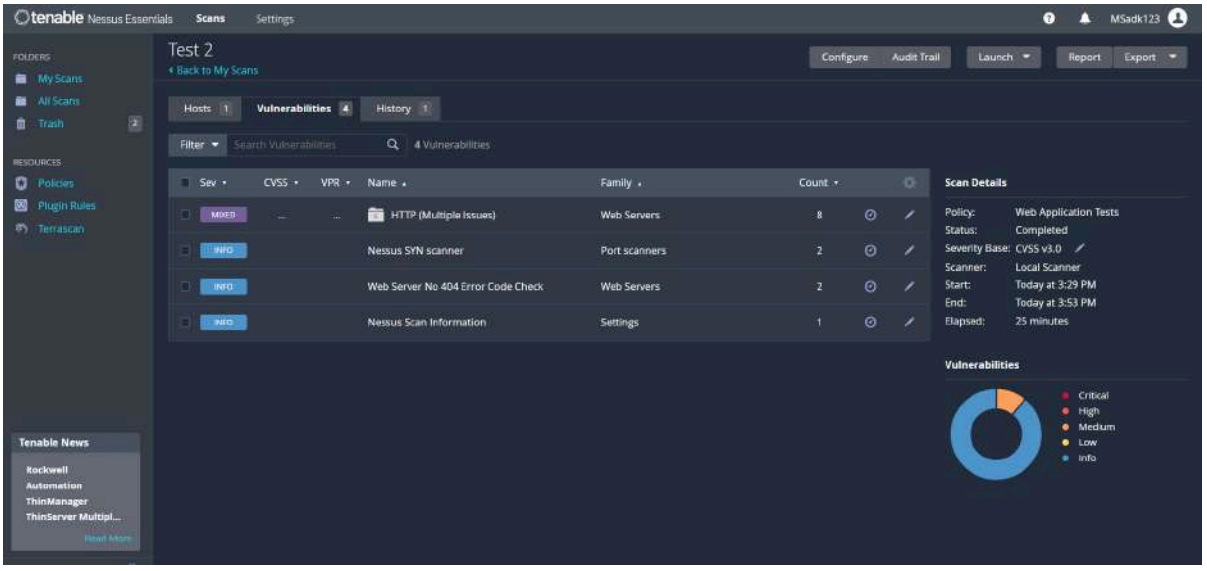
9. Generating report



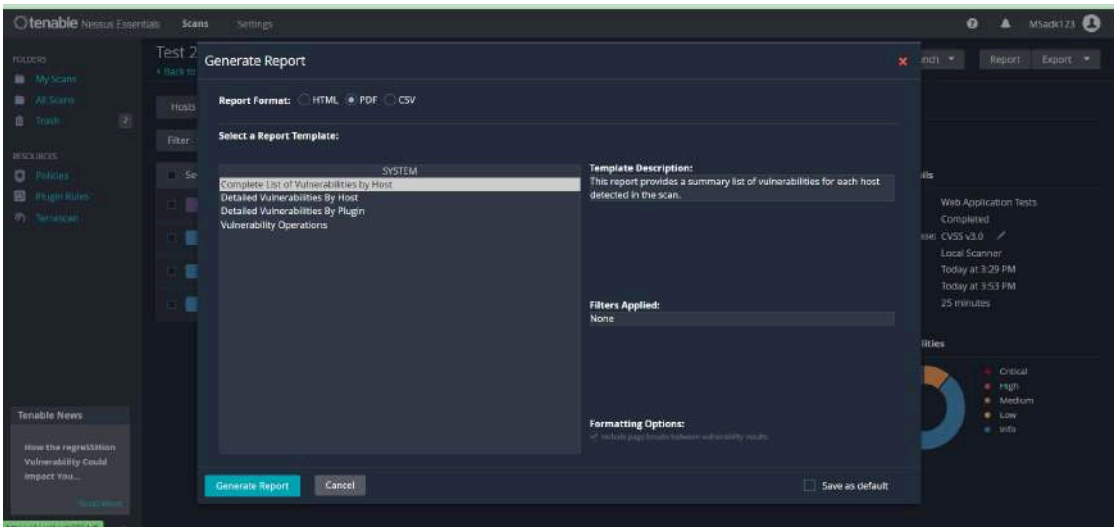
10. Here is the final report



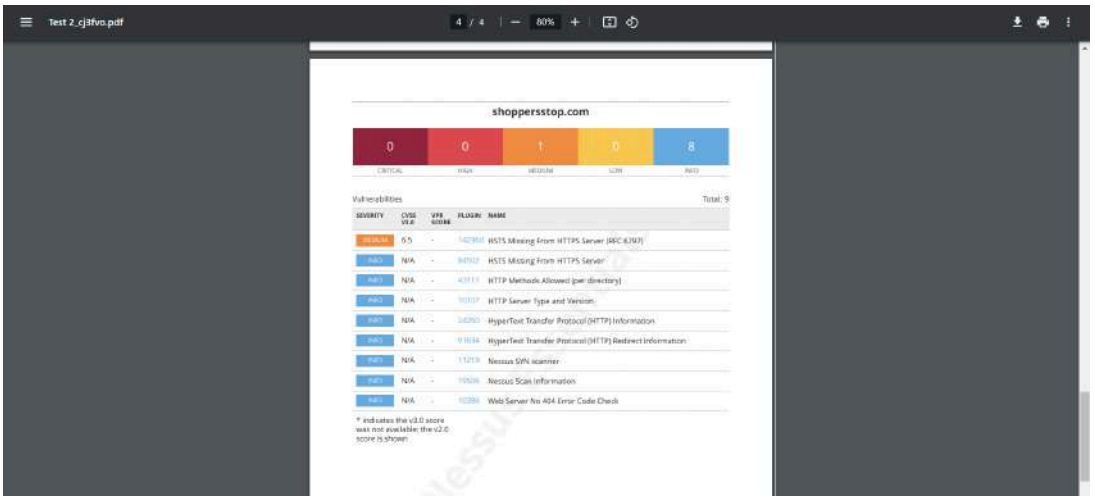
11. Scanning for test 2 also completed



12. Generating report



13. Here is the final report



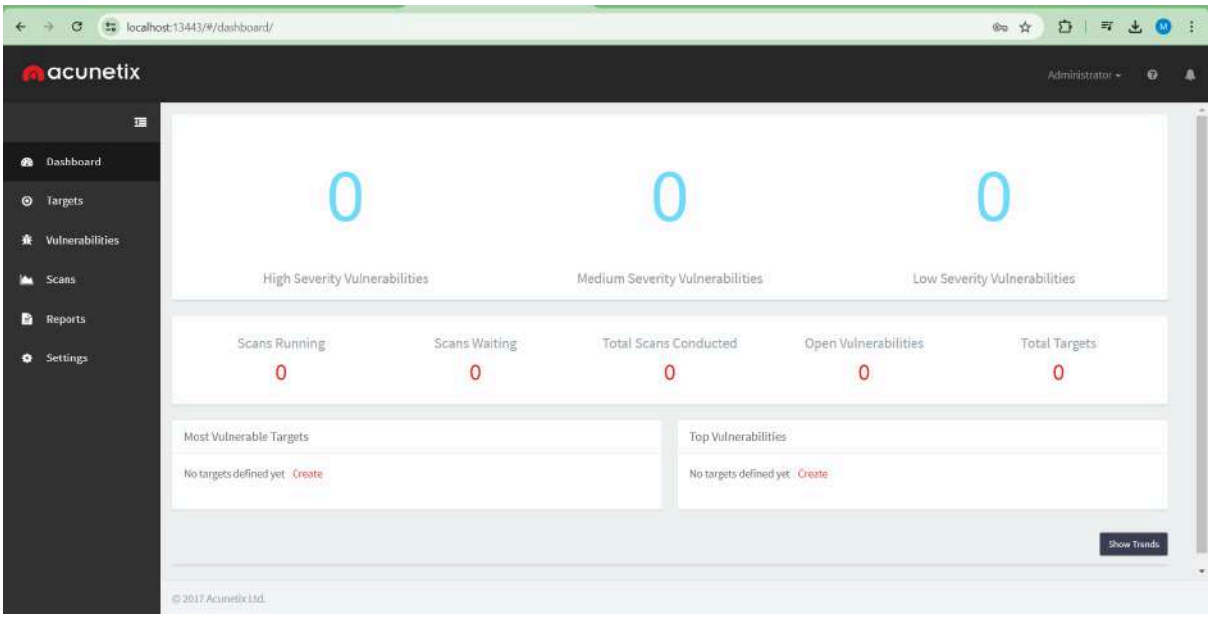
<u>CONCLUSION:</u>

Hence we were able to perform web-application testing on the following targets and retrieve a report of vulnerabilities present in the web-application.

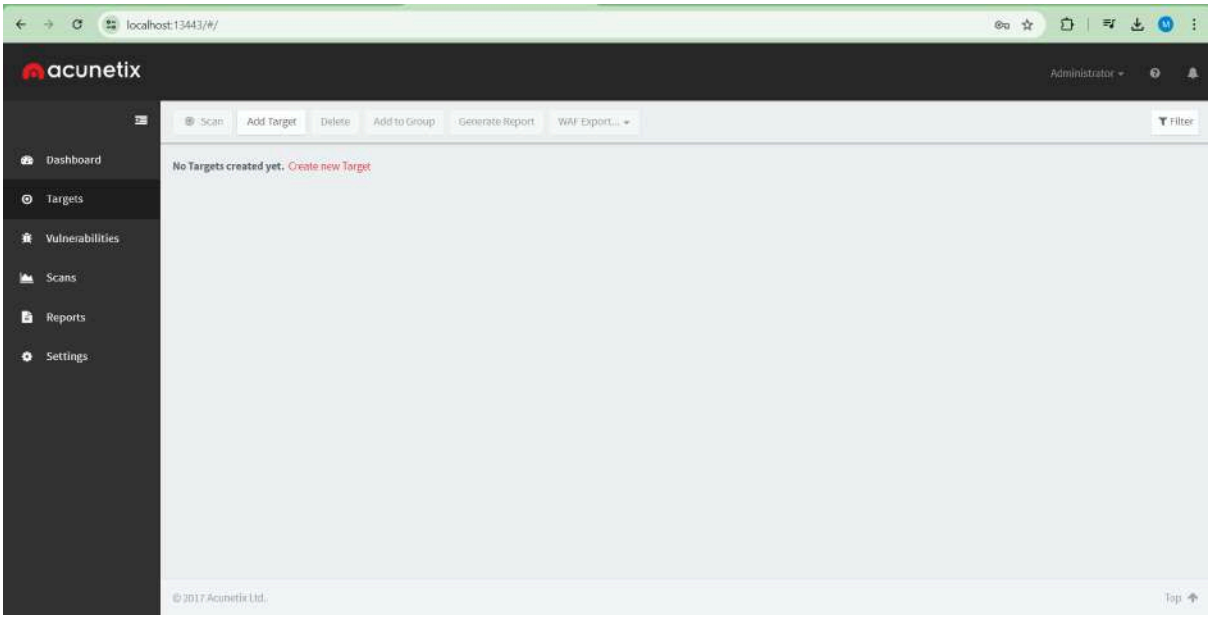
C. Scan the below-mentioned targets Using the Acunetix Vulnerability scanner:

a) https://www.ebay.com/

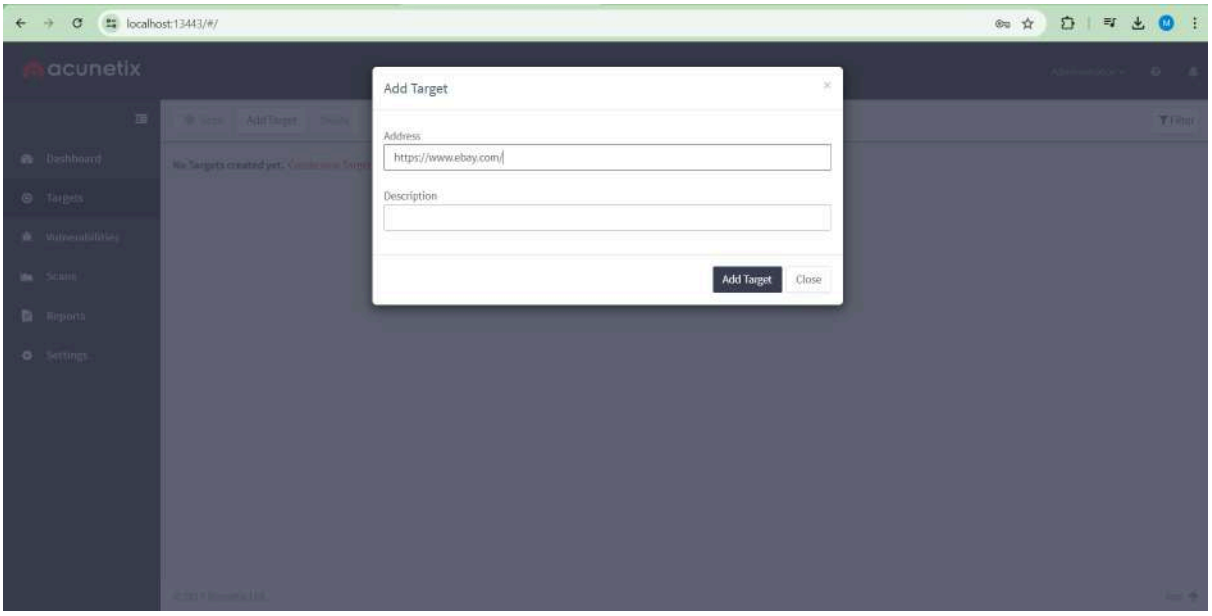
1. Open Acunetix web Application



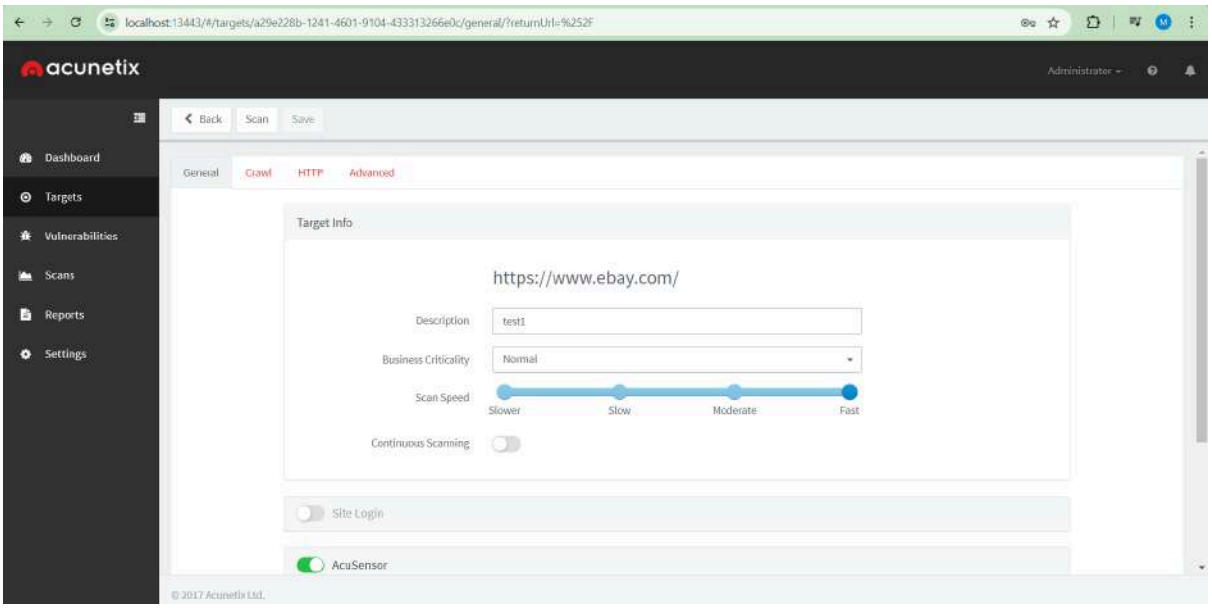
2. Go to targets



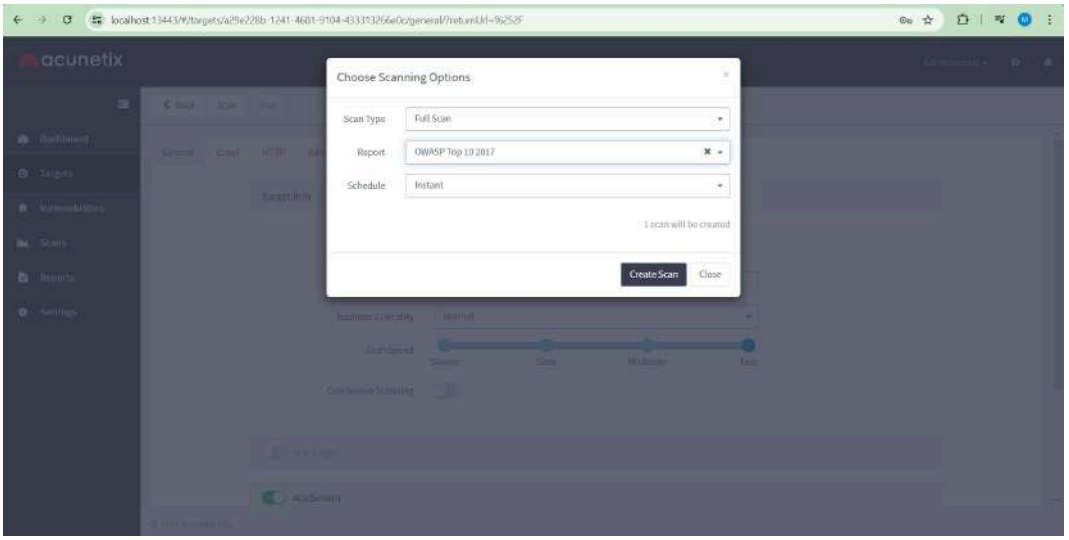
3. Click on add targets and Paste the target url



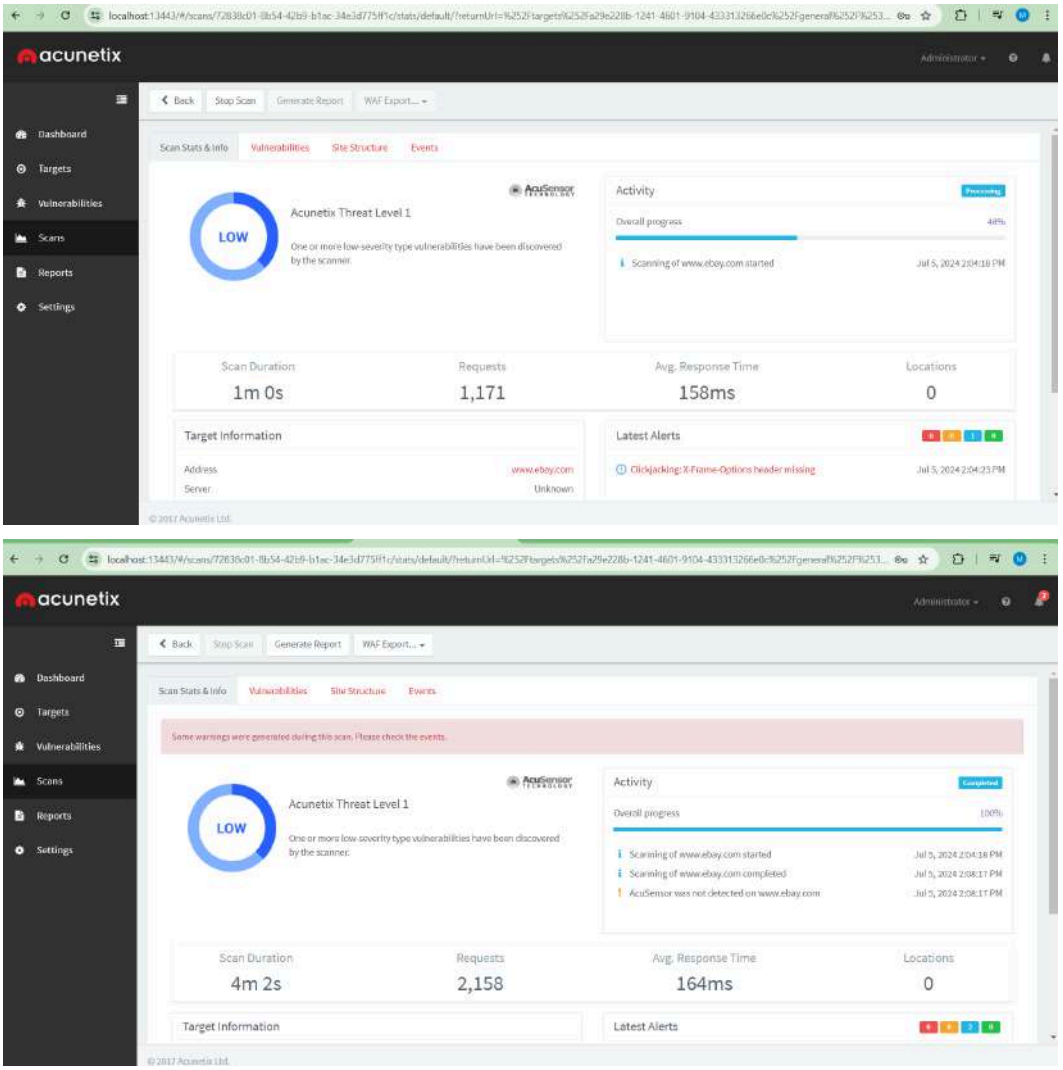
4. Click on add target



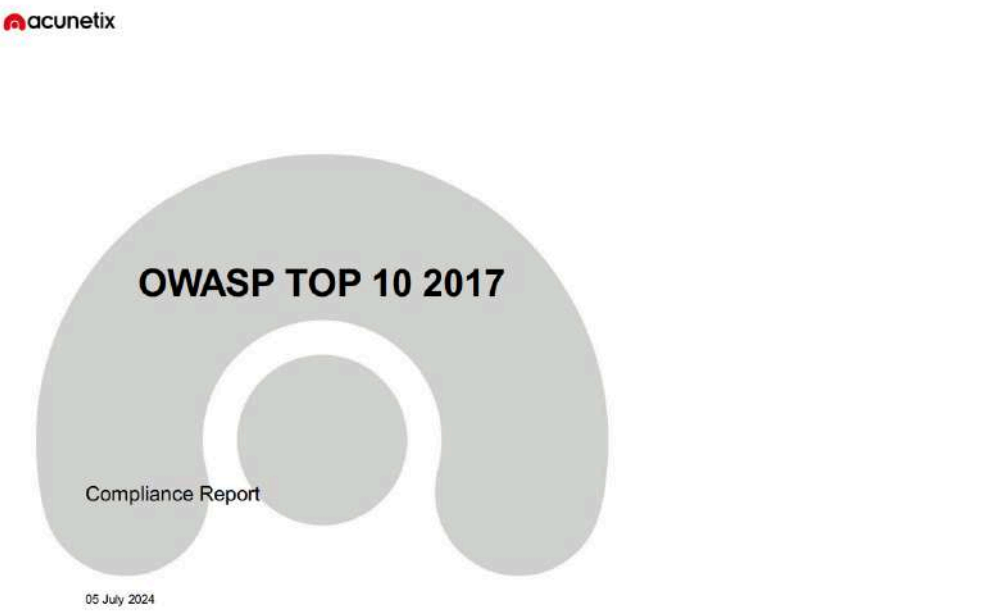
5. Click on Scan , and choose report type



6. Click on create scan



7. Click on generate report and you can download it



Scan

URL	https://www.ebay.com/
Scan date	05/07/2024, 14:04:16
Duration	4 minutes, 2 seconds
Profile	Full Scan

Compliance at a Glance

This section of the report is a summary and lists the number of alerts found according to individual compliance categories.

- Injection(A1)
No alerts in this category

- Broken Authentication(A2)
No alerts in this category

- Sensitive Data Exposure(A3)
Total number of alerts in this category: 2

- XML External Entity (XXE)(A4)
No alerts in this category

- Broken Access Control(A5)
Total number of alerts in this category: 1

- Security Misconfiguration(A6)
Total number of alerts in this category: 1

- Cross Site Scripting (XSS)(A7)
No alerts in this category

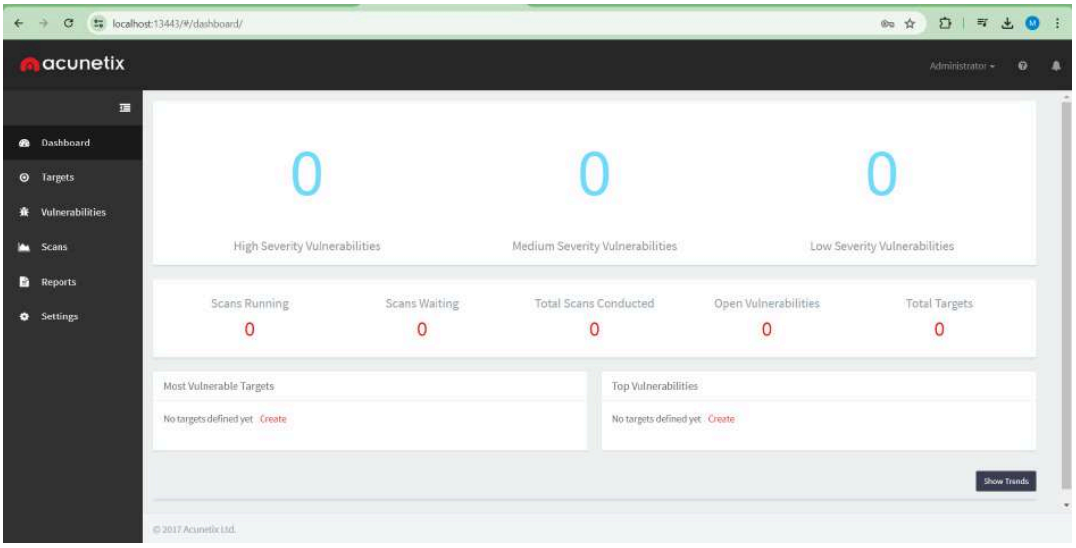
- Insecure Deserialization(A8)
No alerts in this category

- Using Components with Known Vulnerabilities(A9)
Total number of alerts in this category: 1

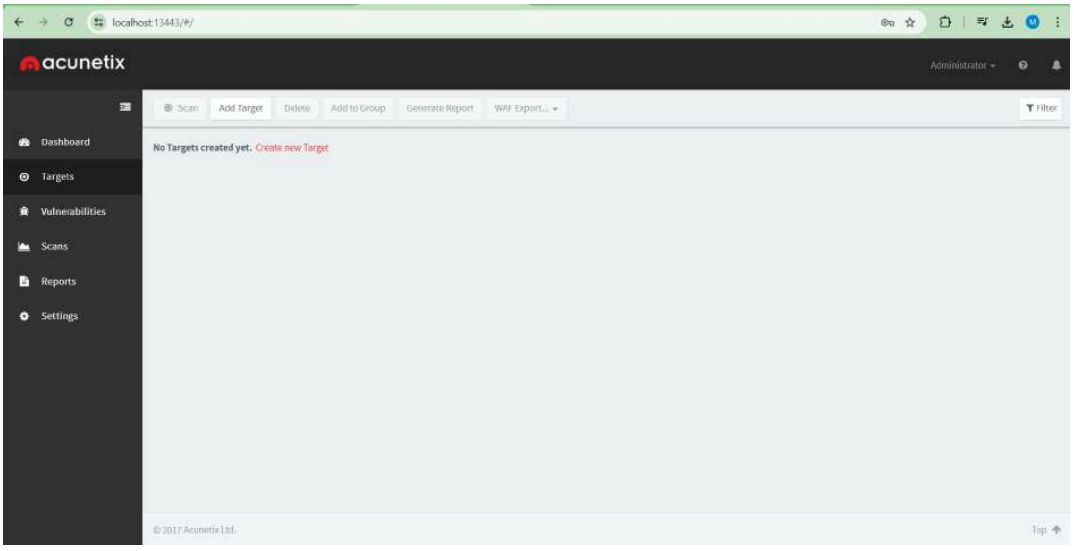
- Insufficient Logging and Monitoring(A10)
No alerts in this category

b) https://shopping.rediff.com/

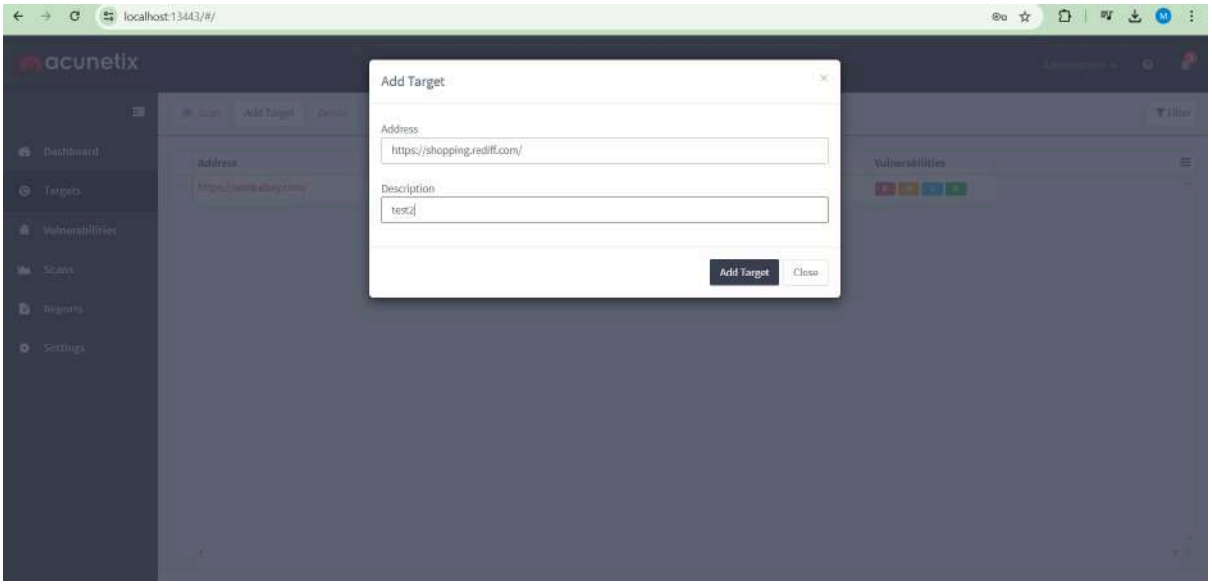
1. Open Acunetix web Application



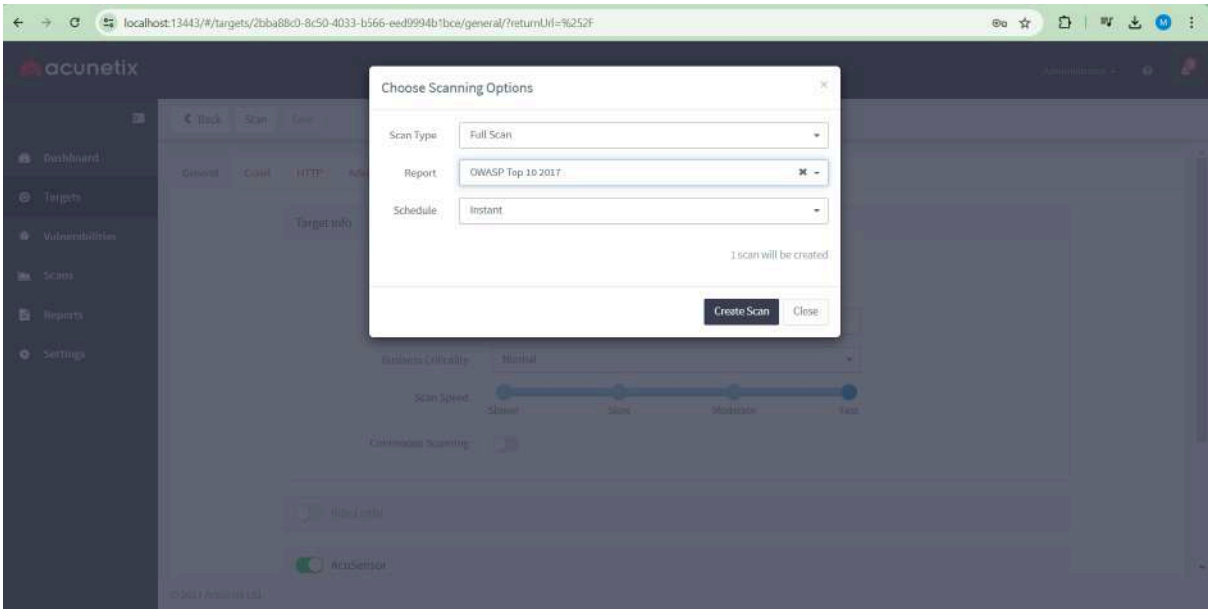
2. Go to targets



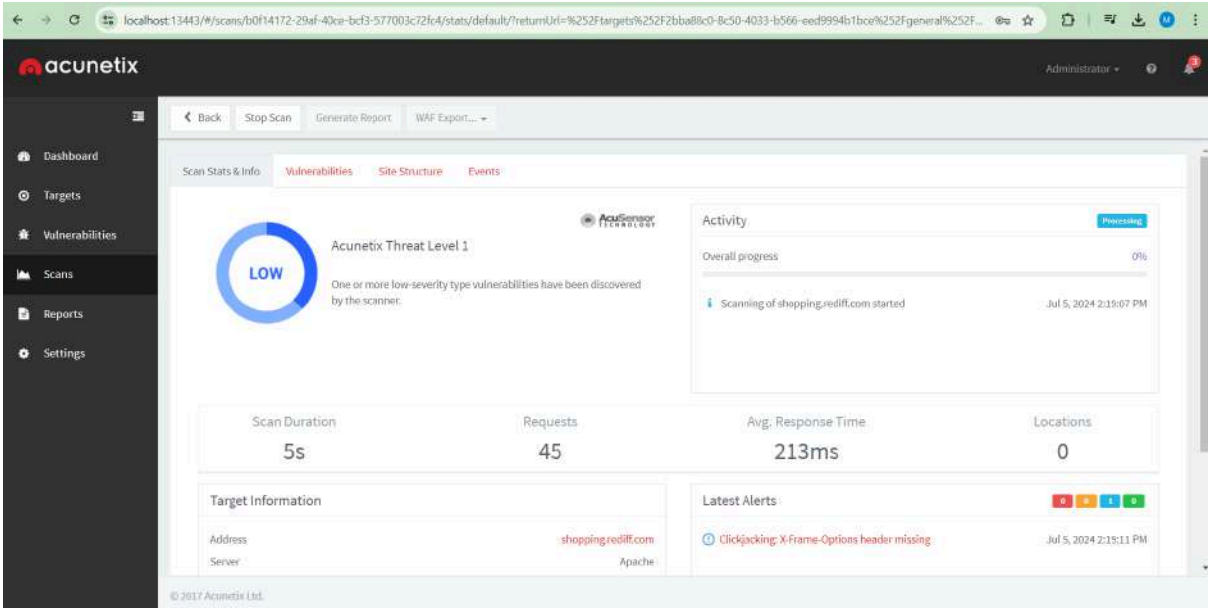
3. Click on add targets and Paste the target url

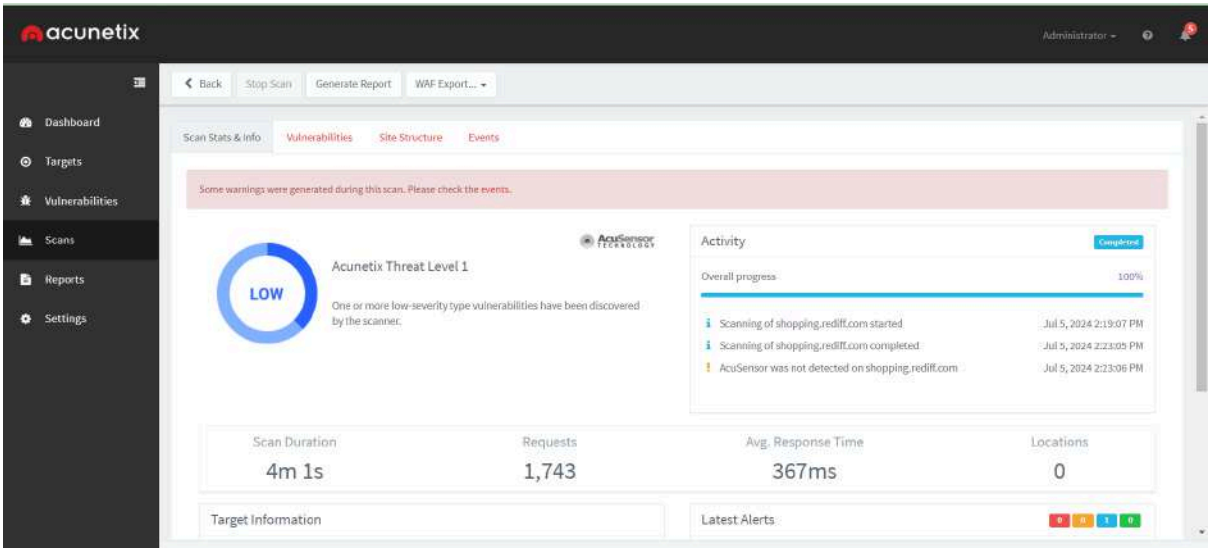


4. Click on Scan



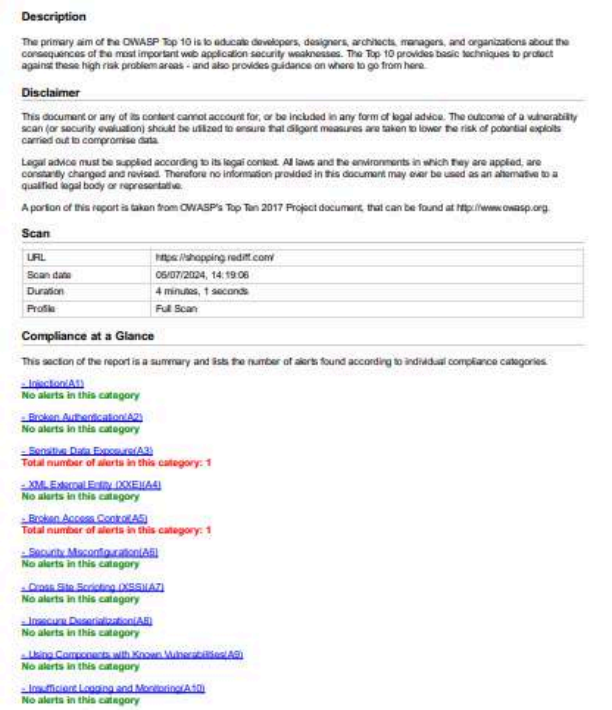
5. Click on create scan





Scan completed

6. Click on generate report and download it



Description

The primary aim of the OWASP Top 10 is to educate developers, designers, architects, managers, and organizations about the consequences of the most important web application security weaknesses. The Top 10 provides basic techniques to protect against these high risk problem areas - and also provides guidance on where to go from here.

Disclaimer

This document or any of its content cannot account for, or be included in any form of legal advice. The outcome of a vulnerability scan (or security evaluation) should be utilized to ensure that diligent measures are taken to lower the risk of potential exploits carried out to compromise data.

Legal advice must be supplied according to its legal context. All laws and the environments in which they are applied, are constantly changed and revised. Therefore no information provided in this document may ever be used as an alternative to a qualified legal body or representative.

A portion of this report is taken from OWASP's Top Ten 2017 Project document, that can be found at http://www.owasp.org.

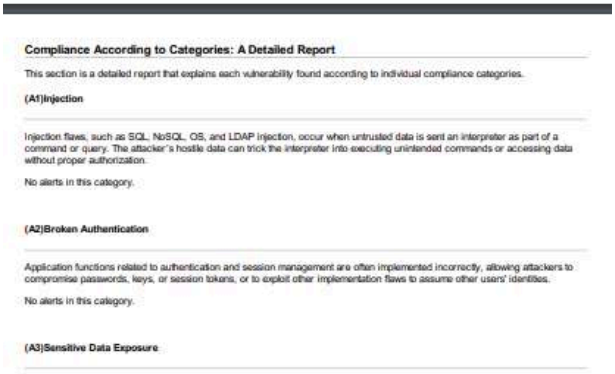
Scan

URL	https://shopping.rediff.com/
Scan date	05/07/2024, 14:19:06
Duration	4 minutes, 1 seconds
Profile	Full Scan

Compliance at a Glance

This section of the report is a summary and lists the number of alerts found according to individual compliance categories.

- Injection(A1)
  No alerts in this category
- Broken Authentication(A2)
  No alerts in this category
- Sensitive Data Exposure(A3)
  Total number of alerts in this category: 1
- XML External Entity (XXE)(A4)
  No alerts in this category
- Broken Access Control(A5)
  Total number of alerts in this category: 1
- Security Misconfiguration(A6)
  No alerts in this category
- Cross Site Scripting (XSS)(A7)
  No alerts in this category
- Insecure Deserialization(A8)
  No alerts in this category
- Using Components with Known Vulnerabilities(A9)
  No alerts in this category
- Insufficient Logging and Monitoring(A10)
  No alerts in this category



Compliance According to Categories: A Detailed Report

This section is a detailed report that explains each vulnerability found according to individual compliance categories.

(A1)Injection

Injection flaws, such as SQL, NoSQL, OS, and LDAP injection, occur when untrusted data is sent an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.

No alerts in this category.

(A2)Broken Authentication

Application functions related to authentication and session management are often implemented incorrectly, allowing attackers to compromise passwords, keys, or session tokens, or to exploit other implementation flaws to assume other users' identities.

No alerts in this category.

(A3)Sensitive Data Exposure

CONCLUSION:

Hence we were able to find the vulnerabilities of the following websites using the Acunetix vulnerability scanner by providing the website's domain .

Supraja Technologies

23E05-ST#IS#6653-TASK10

A. Perform No Rate Limiting on the login OTP page of the following websites mentioned below:

A. https://www.freshbus.com/

B. https://nuego.in/

C. https://yolobus.in/

No Rate Limiting Vulnerability

- **CVSS Score** - 5.3 (Medium)
- **Related OWASP Top 10** - A04:2021 - Insecure Design
- **Explanation**: A No Rate Limiting vulnerability occurs when an application fails to impose restrictions on the number or frequency of requests a user or IP address can make within a given time frame. This absence of rate limiting allows attackers to make an excessive number of requests to the application, potentially leading to various security issues and performance problems.
- **Impact**:

1. Denial of Service (DoS)
2. Brute Force Attacks
3. Scraping and Data Harvesting
4. API Abuse
5. Account Lockouts
6. Resource Exhaustion
7. Increased Infrastructure Costs

- **Recommendations**:

1. Implement Rate Limiting: Set appropriate limits on the number of requests allowed per user/IP within a specific time frame.
2. Graduated Rate Limiting
3. IP-based Rate Limiting
4. User-based Rate Limiting

5. API Key Rate LimitingImplement CAPTCHA

❖ **References**:

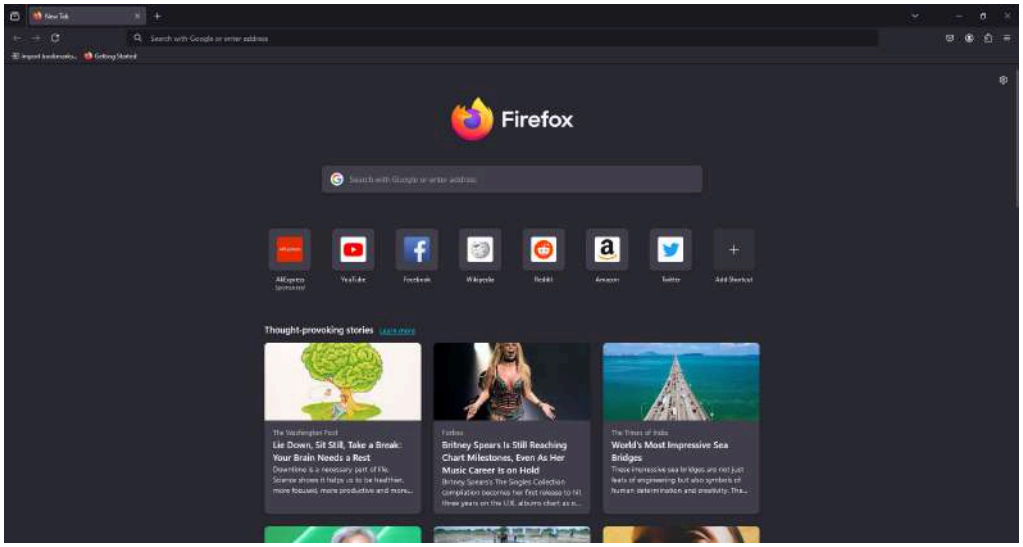
1. [OWASP API Security Top 10 2019 - API4:2019 Lack of Resources & Rate Limiting:]
2. [PortSwigger Web Security Academy: Rate limiting and other request-based defenses:]

❖ **Procedure:**

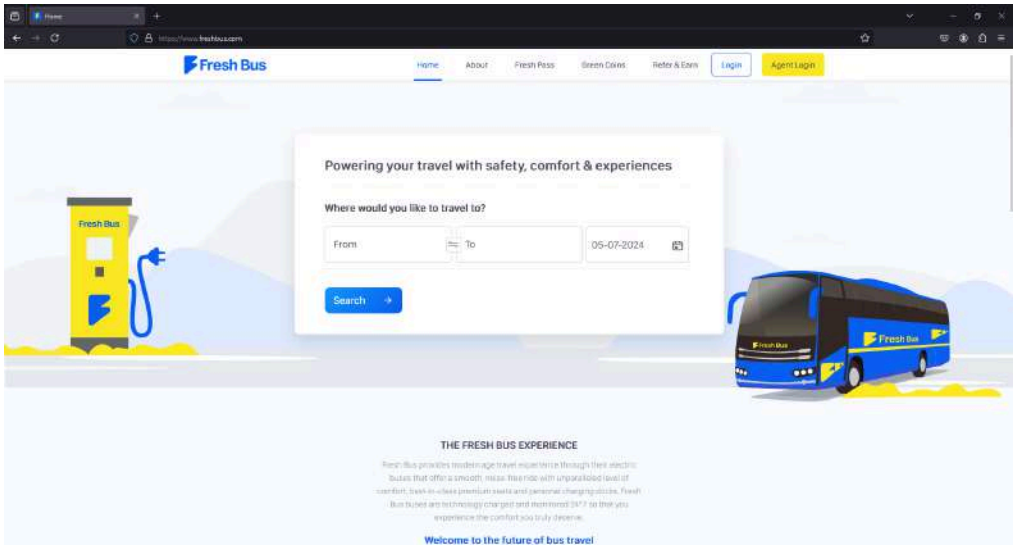
➢ **Target Website -** [https://www.freshbus.com]

➢ **Steps -**

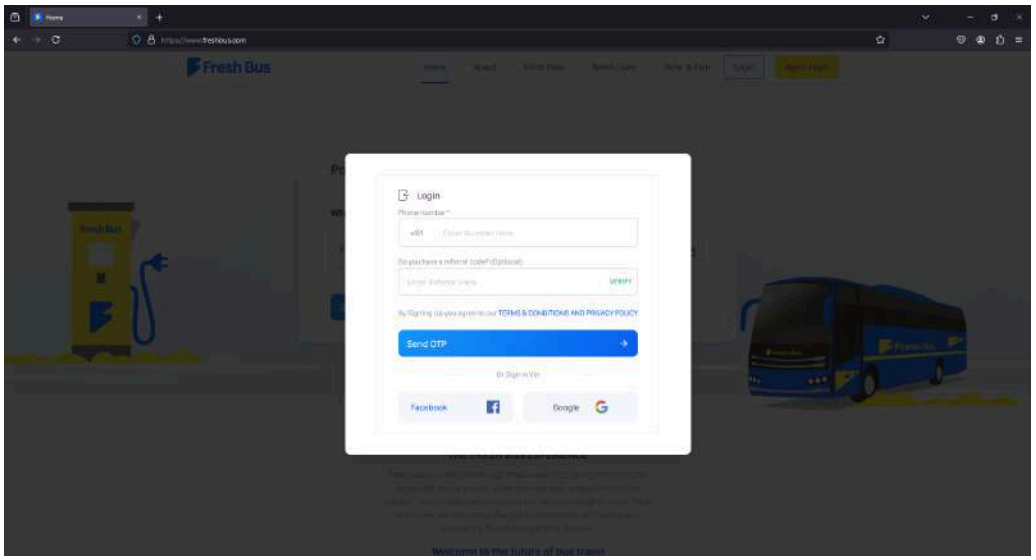
STEP-1: Open firefox browser, and make sure the proxy setting is configured with BurpSuite.



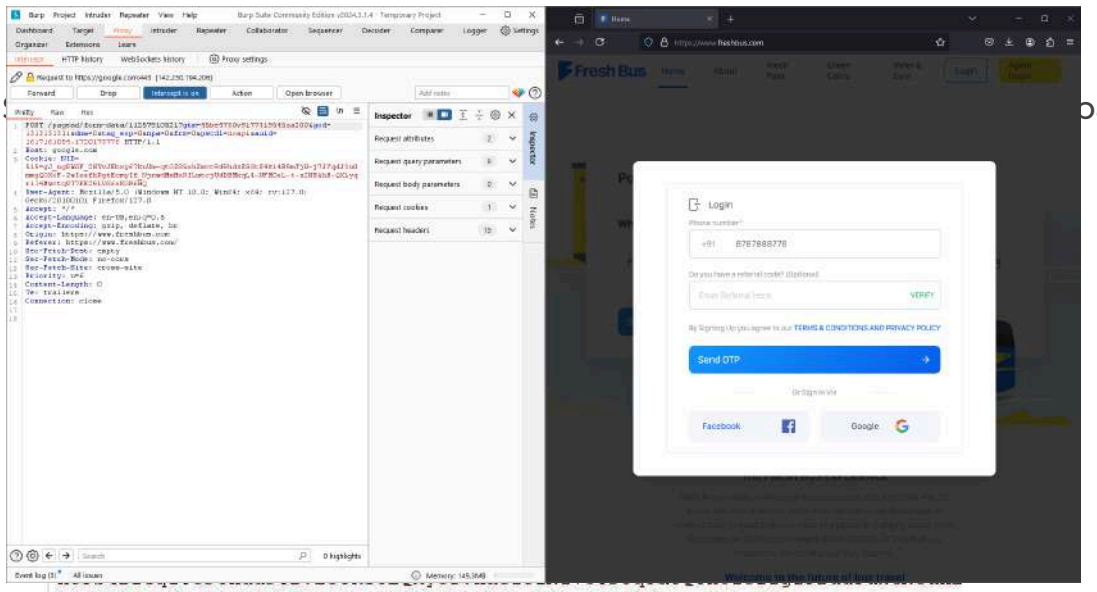
STEP-2: Search for https://www.freshbus.com/



STEP-3: Now, find and go to the login page of the website.



STEP-5: Now enter some random credential in the entries required. Simultaneously open Burp Suite and turn on the intercept.



bage.

```
bBxIwl8xXK9eEgf2gZDYzlcYBkTzsbmokISivGHqtqzBMLojvT; _gcl_au=
1.1.1617161054.1720173776; ci_session=mlpplidtd6un41ld4gvh9vb85kesah9r;
_ga_03S3LDTVT4=GS1.1.1720173776.1.1.1720176152.33.0.0; _ga=
GA1.1.442763591.1720173777; _clck=1r2puuu%7C2%7Cfn7%7C0%7C1647; _fbp=
fb.1.1720173778420.520129386960856578; G_ENABLED_IDPS=google
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:127.0)
Gecko/20100101 Firefox/127.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/json;charset=utf-8
Content-Length: 43
Origin: https://www.freshbus.com
Referer: https://www.freshbus.com/
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Priority: u=1
Te: trailers

{
  "username":"8787888778",
  "referralCode":""
}
```

As we can see it is the same as entered in the login page.

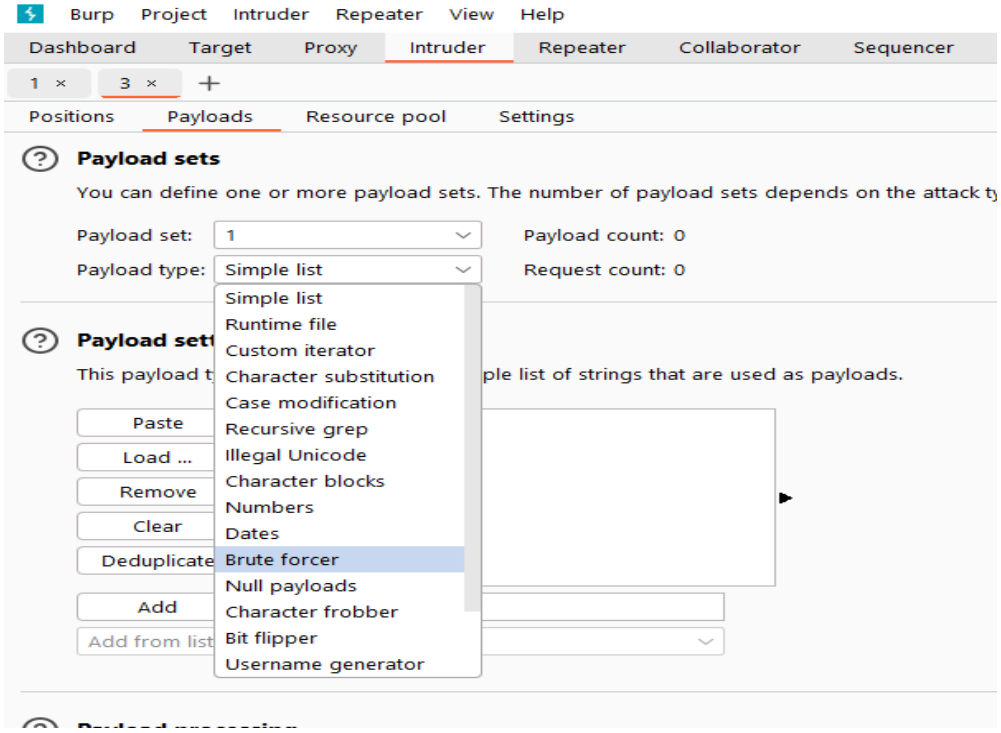
STEP-7: Now look for Accept-Language:en-US,en: q=0.5. Select 0.5 then click on add.

```
POST /api//payment/send_otp_login HTTP/2
Host: www.freshbus.com
Cookie: AWSALB=
MtJS4BfwqlYG9UHdd9IBvLctK5ULQXyGOVHaeZ6fNz+CIS
YBkTzsbmokISivGHqtqzBMLojvT; AWSALBCORS=
MtJS4BfwqlYG9UHdd9IBvLctK5ULQXyGOVHaeZ6fNz+CIS
YBkTzsbmokISivGHqtqzBMLojvT; _gcl_au=1.1.1617
mlpplidtd6un411d4gvh9vb85kesah9r; _ga_03S3LDTV
GA1.1.442763591.1720173777; _clck=1r2puuu%7C2%
fb.1.1720173778420.520129386960856578; G_ENABI
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.$5$
Accept-Encoding: gzip, deflate, br
Content-Type: application/json;charset=utf-8
Content-Length: 43
Origin: https://www.freshbus.com
Referer: https://www.freshbus.com/
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Priority: u=1
Te: trailers

{"username":"8787888778","referralCode":""}
```

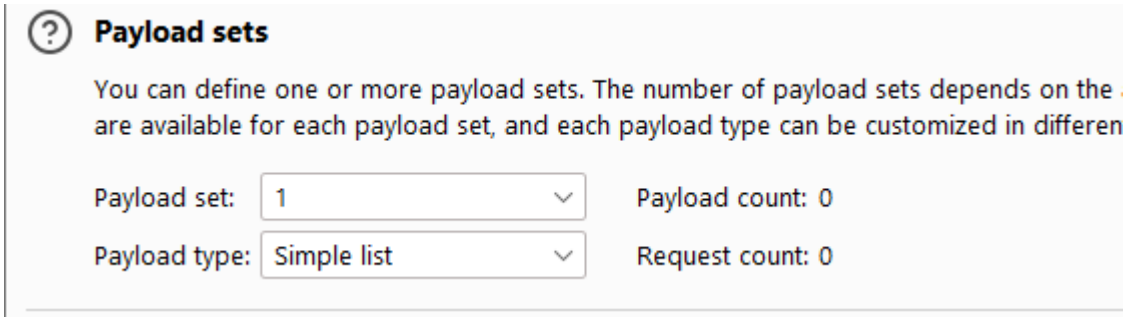
This will automatically add $ before and after 0.5, resulting in q=0.5

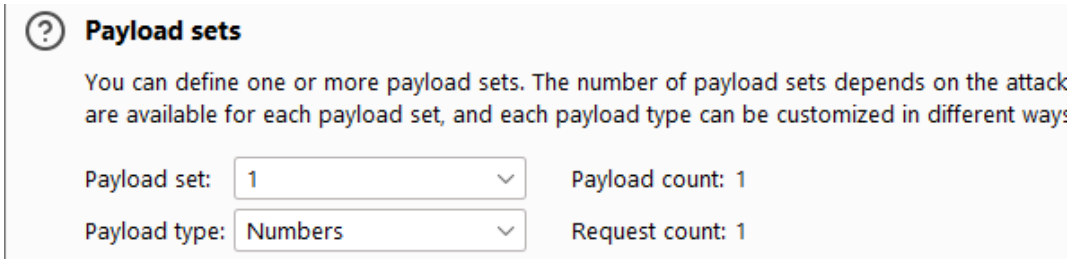
NOTE: Ensure to click on clear before this process.

STEP-8: Turn off the intercept and go to the payload section.

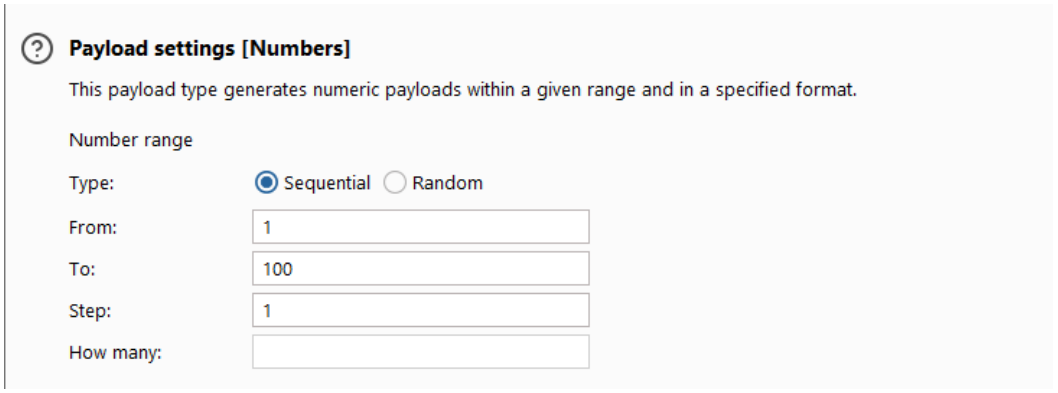


STEP-09: Here you will find payload type as simple list as default .Change it to Numbers.



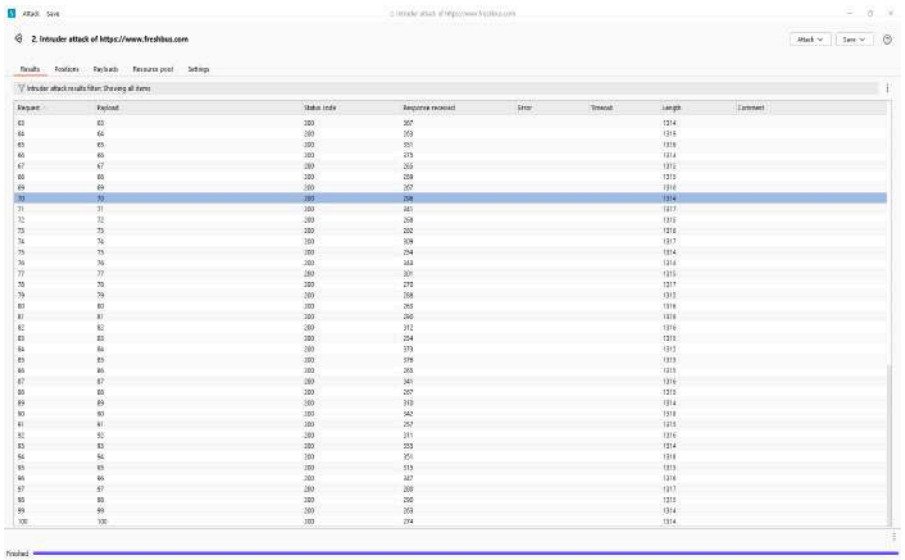


STEP-10: Also alter the payload settings to the one's given below.



- From: 1
- To: 100
- Step: 1

STEP-11: After alteration of changes. Click on start attack.

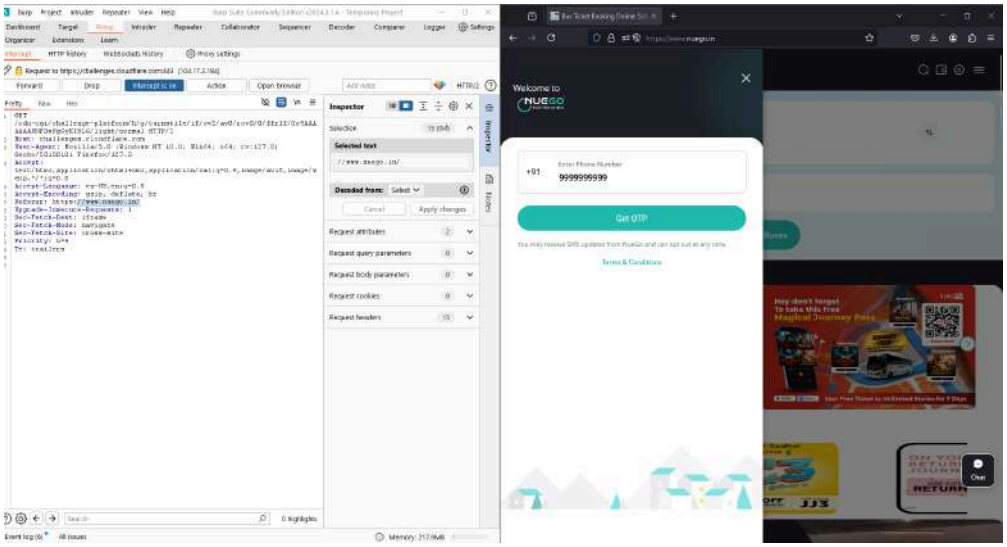


As we can see burp suite will start sending 100 requests.

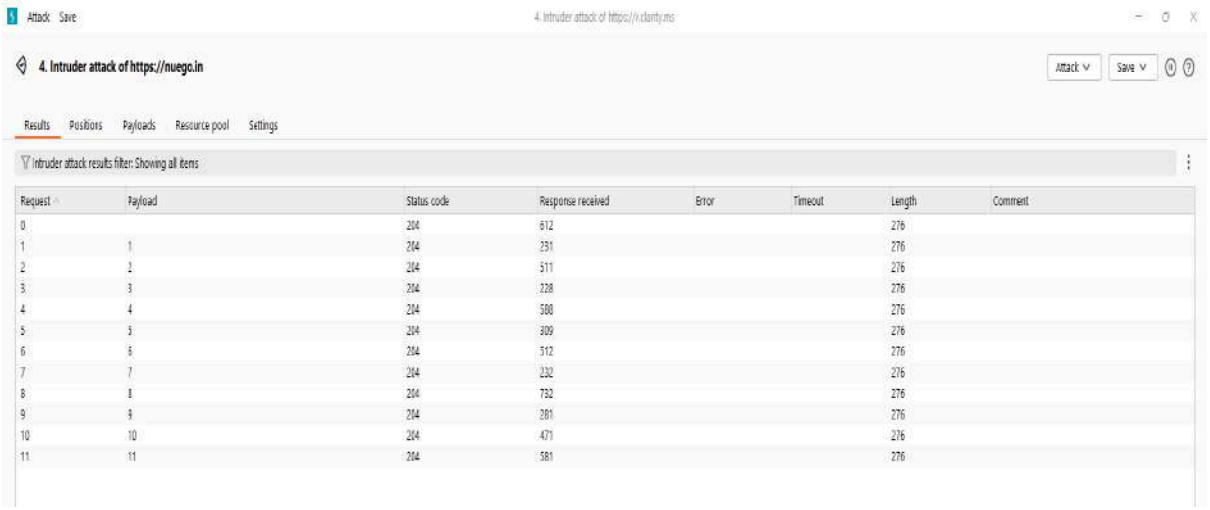
CONCLUSION: if performing on any target if the set request matches with the actual request sent during the attack then the target is said to be vulnerable.This website is vulnerable.

TARGET-B:(https://nuego.in/)

We have to follow the same steps as done for TARGET A.



Result:

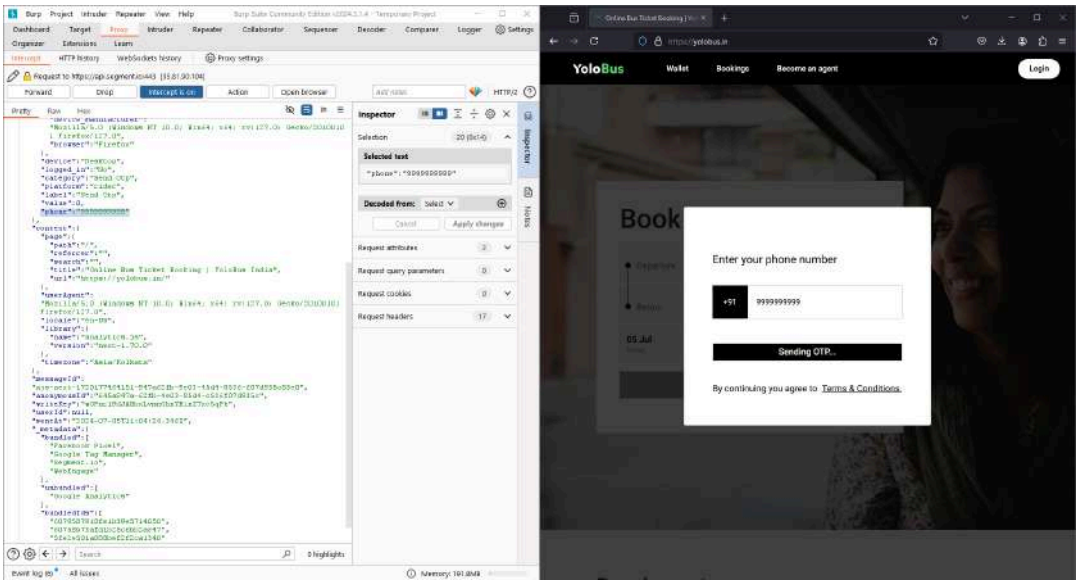


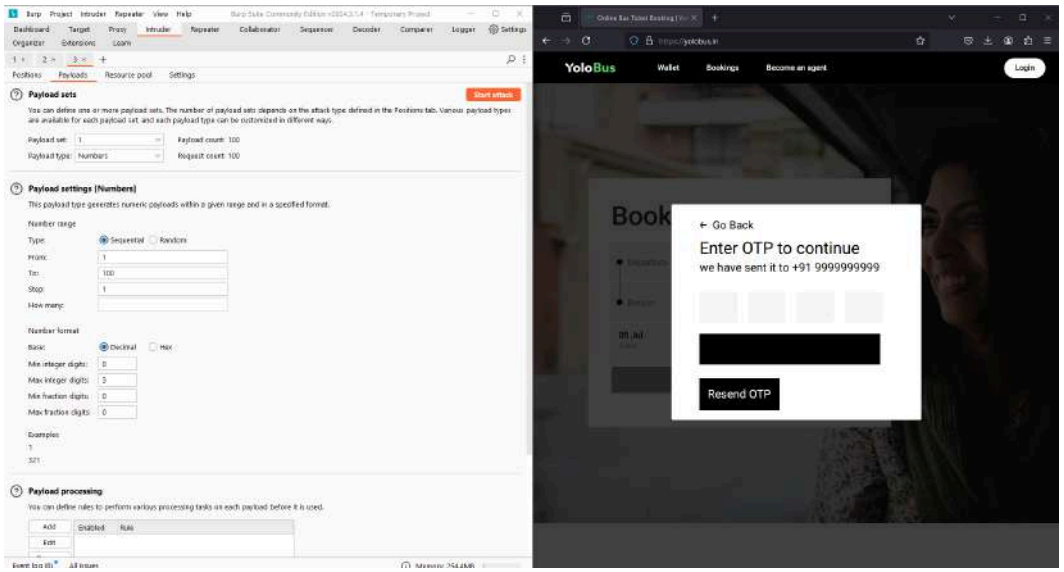
Request	Payload	Status code	Response received	Error	Timeout	Length	Comment
0		204	612			276	
1	1	204	231			276	
2	2	204	511			276	
3	3	204	228			276	
4	4	204	588			276	
5	5	204	309			276	
6	6	204	512			276	
7	7	204	232			276	
8	8	204	732			276	
9	9	204	281			276	
10	10	204	471			276	
11	11	204	581			276	

CONCLUSION: This website is vulnerable.

TARGET-C:(https://https://yolobus.in/.in/)

Repeating the same steps as done for TARGET A.





Result:

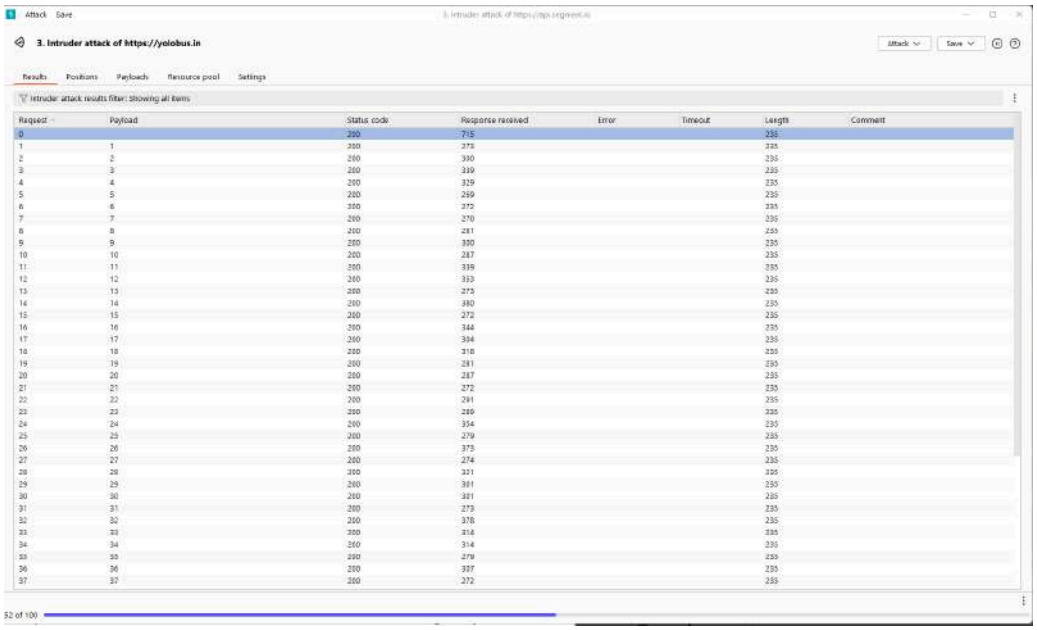
Attack Save

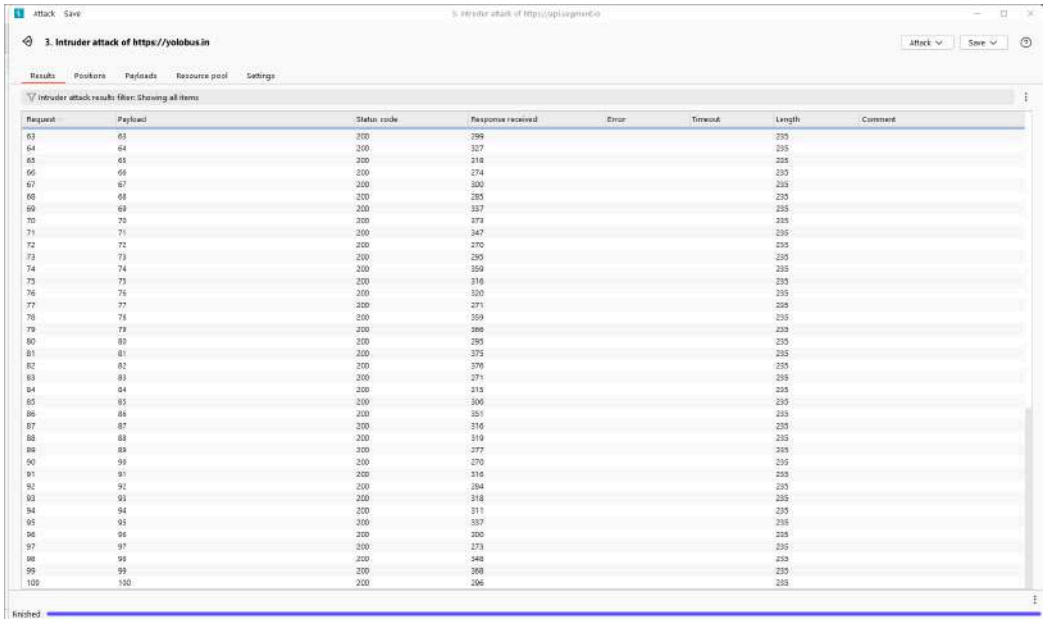
3. Intruder attack of https://yolobus.in

Results Positions Payloads Resource pool Settings

Intruder attack results filter: Showing all items

Request	Payload
0	
1	1
2	2
3	3





CONCLUSION: All 100 requests were sent so the website is vulnerable.

B. Perform a Parameter (price) tampering on any 2 websites and Prepare clear Documentation.

PARAMETER TAMPERING VULNERABILITY

- **CVSS Score** - 6.5 (Medium)
- **Related OWASP Top 10** - A03:2021 - Injection
- **Explanation**: Parameter tampering is a vulnerability that occurs when an application does not properly validate, filter, or sanitise user-supplied input parameters. Attackers can manipulate these parameters (such as form fields, URL parameters, or HTTP headers) to alter the application's behaviour, bypass security controls, or access unauthorised data.
- **Impact**:

1. Unauthorized data access
2. Data manipulation
3. Business logic bypassing
4. Privilege escalation
5. Injection attacks
6. Information disclosure
7. Session hijacking
8. Financial fraud

- **Recommendations**:

1. Input validation: Implement strict server-side input validation for all user-supplied data.
2. Parameterized queries: Use parameterized queries or prepared statements to prevent SQL injection.
3. Whitelist validation
4. Avoid client-side validation only: Never rely solely on client-side validation; always validate on the server-side.
5. Principle of least privilege: Ensure application processes run with minimal necessary privileges.
6. Use strong session management

- **References**:

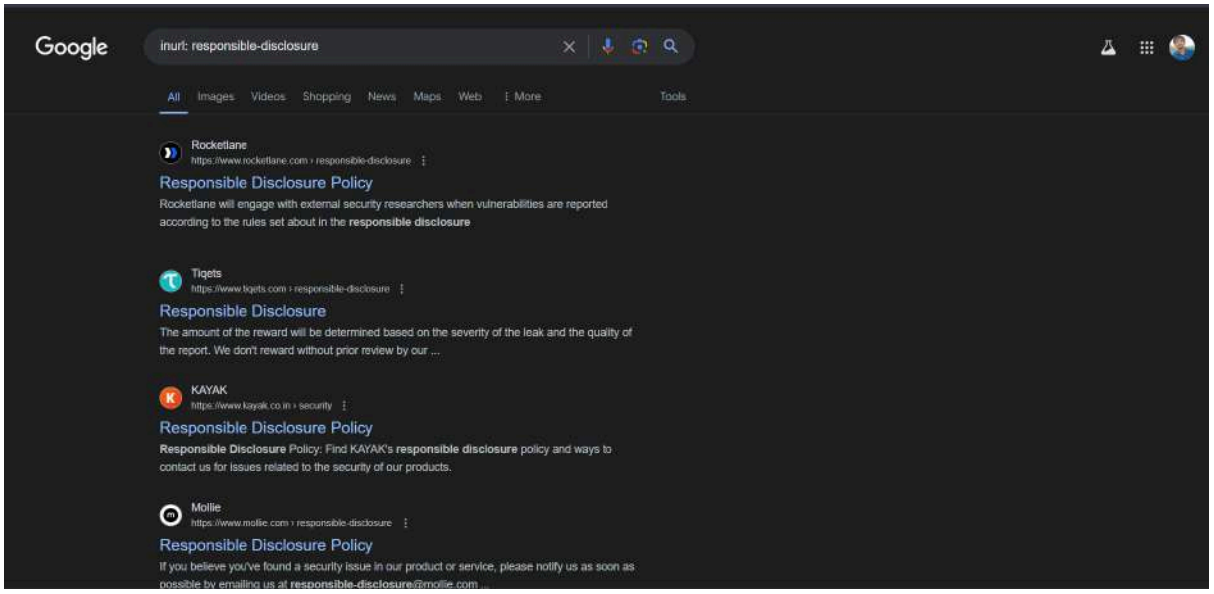
1. OWASP Parameter Tampering
2. PortSwigger Web Security Academy

- **Procedure:**
    - **Target Website -** https://www.justbake.in/
    - **Payload -** Host: www.facebook.com
    - **Steps -**

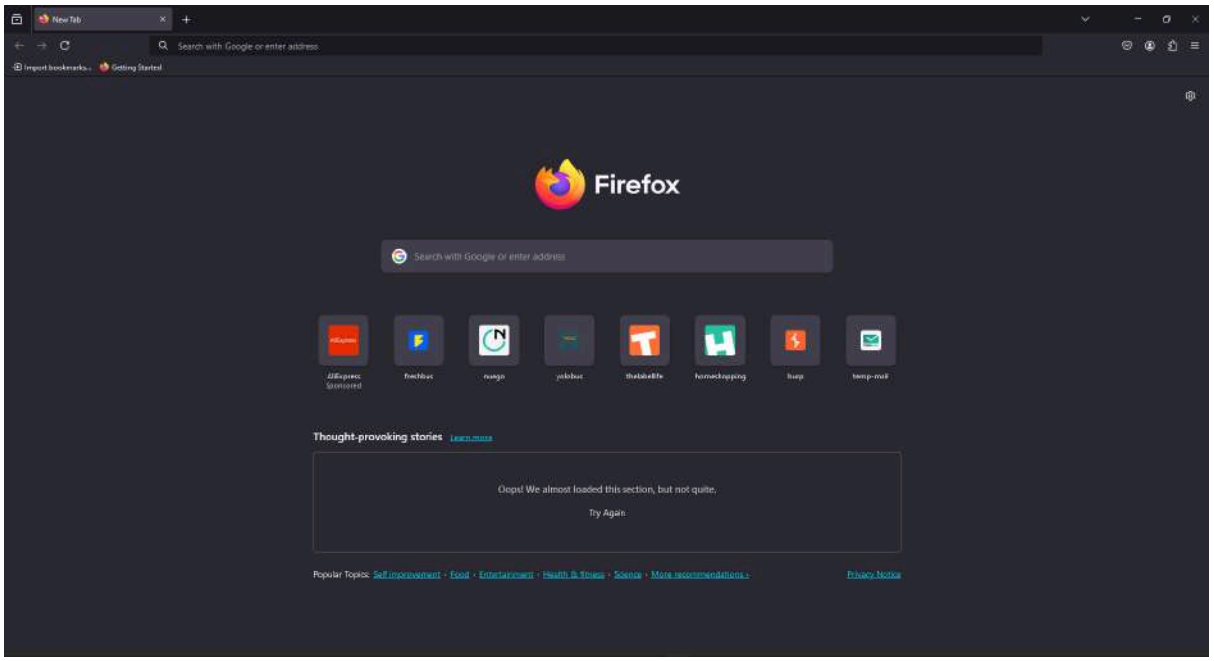
TARGET-1(https://www.justbake.in/)

Step-1: Open chrome and use the following google dork

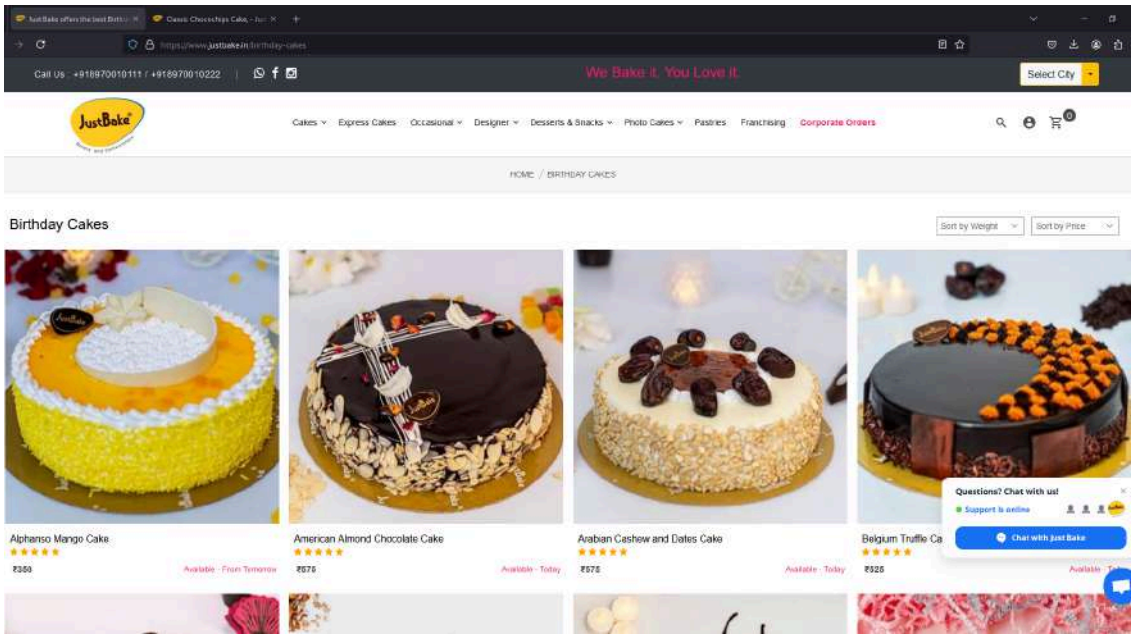
(inurl:Responsible-Disclosure)



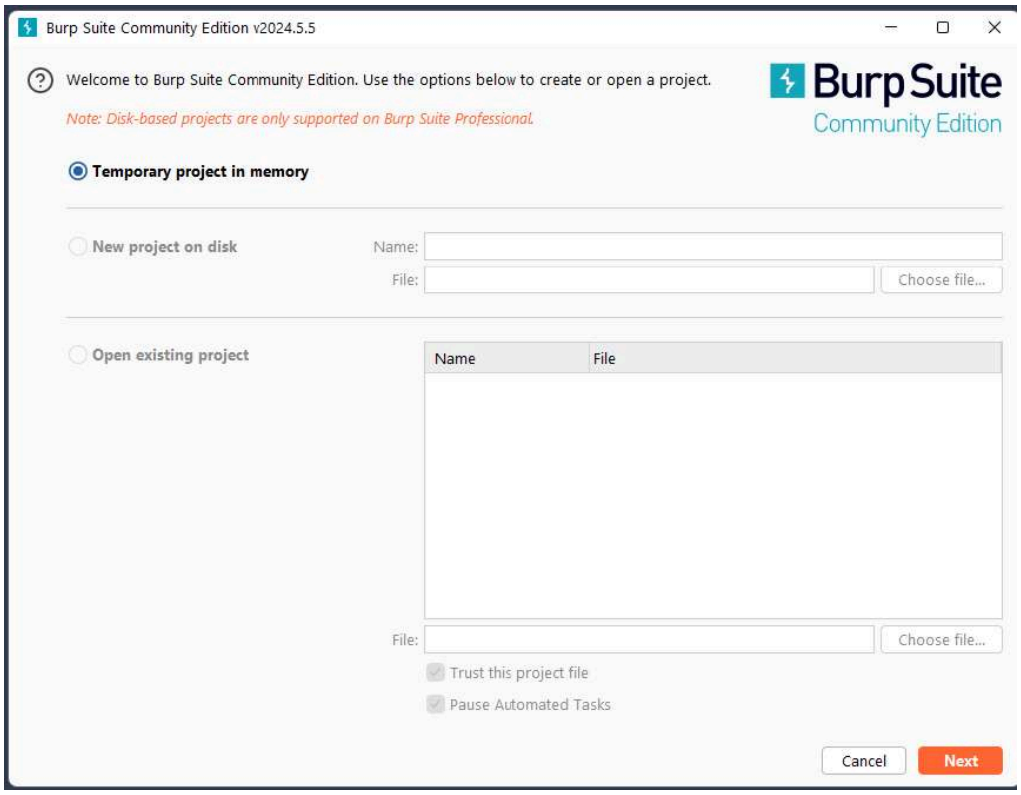
Step-2: Open new tab firefox browser.



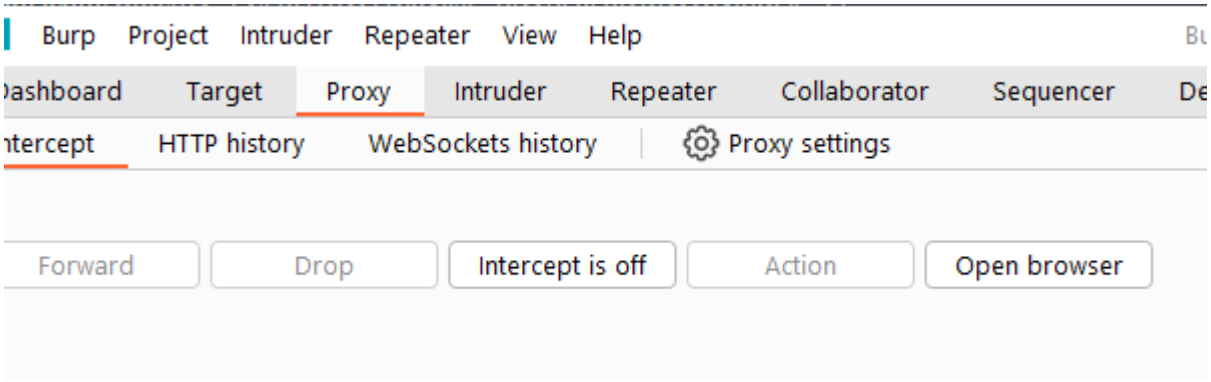
Step-3: Now go to https://www.justbake.in/



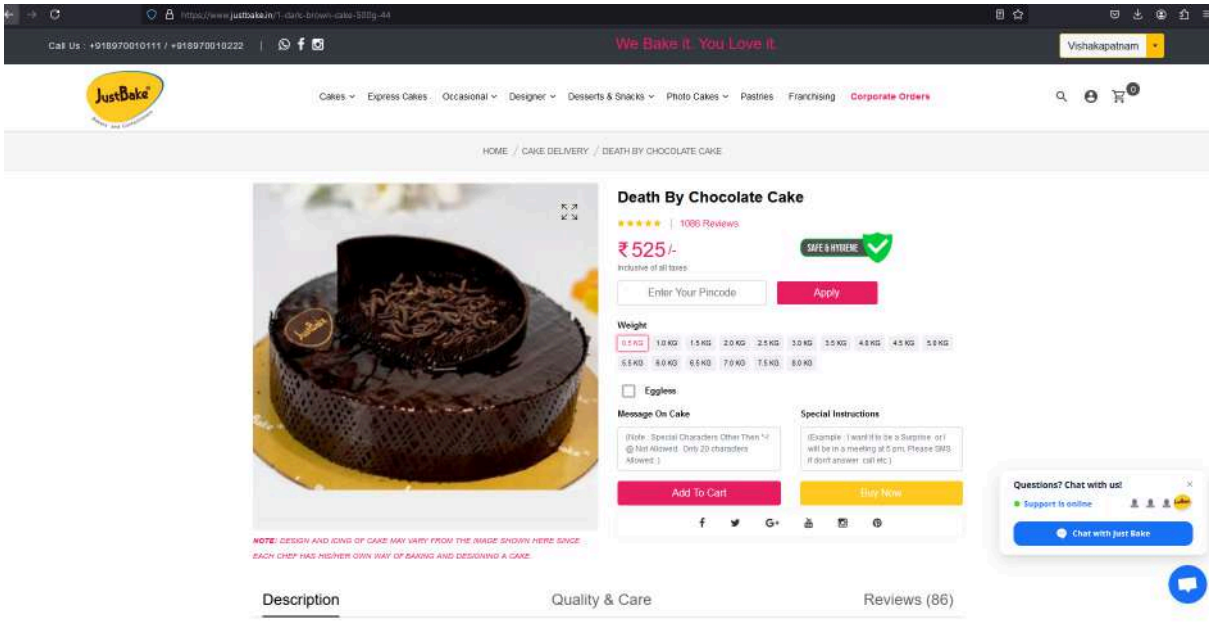
Step-4: Open burp suite community edition



Step-5: Now, go to the proxy tab in burp suite.

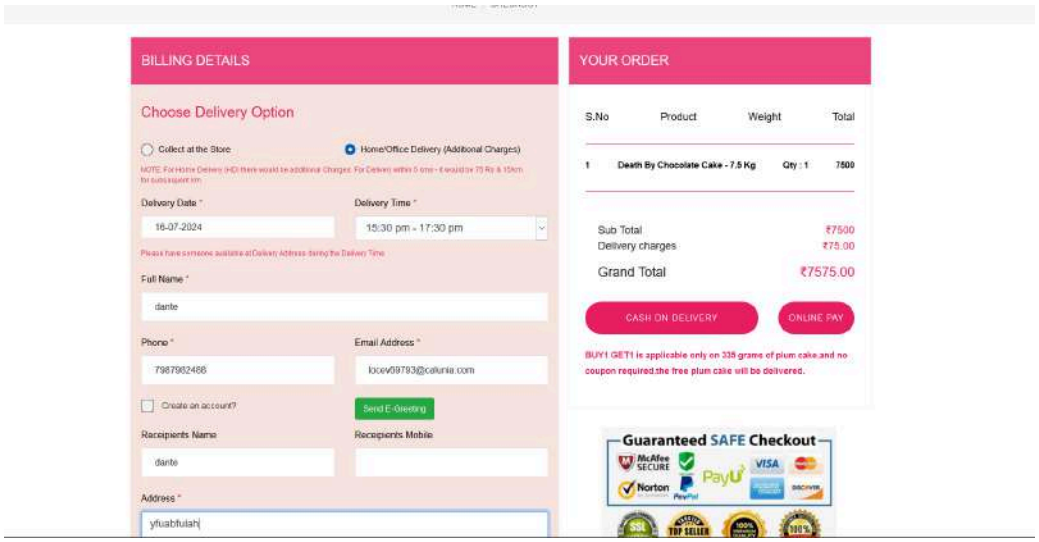


Step-6: Now go to the target website and add any product in the cart.



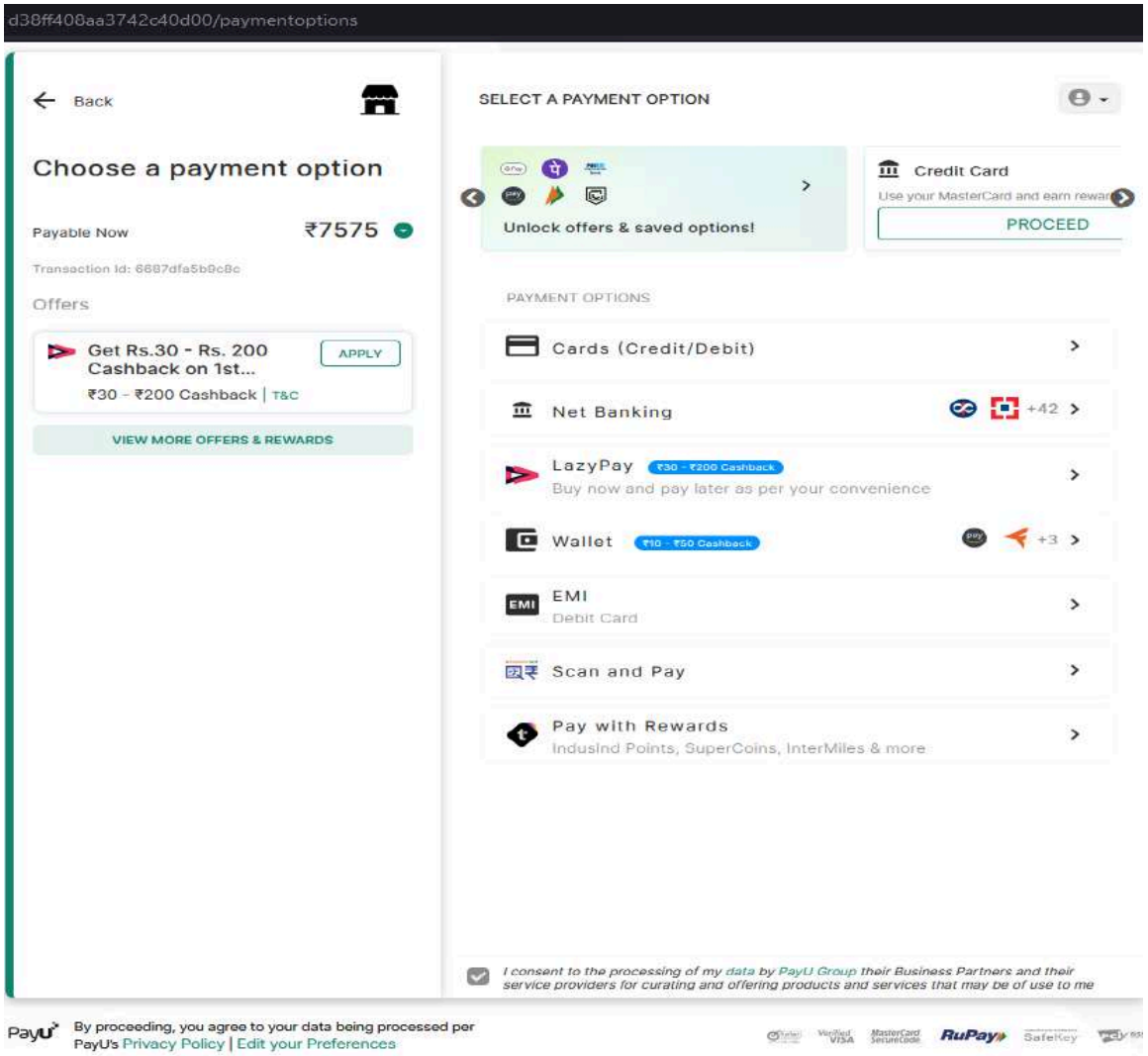
Step-7: To proceed, **use fake credentials except the phone number.**

https://temp-mail.org/en/ can be used for fake and temporary mail id.



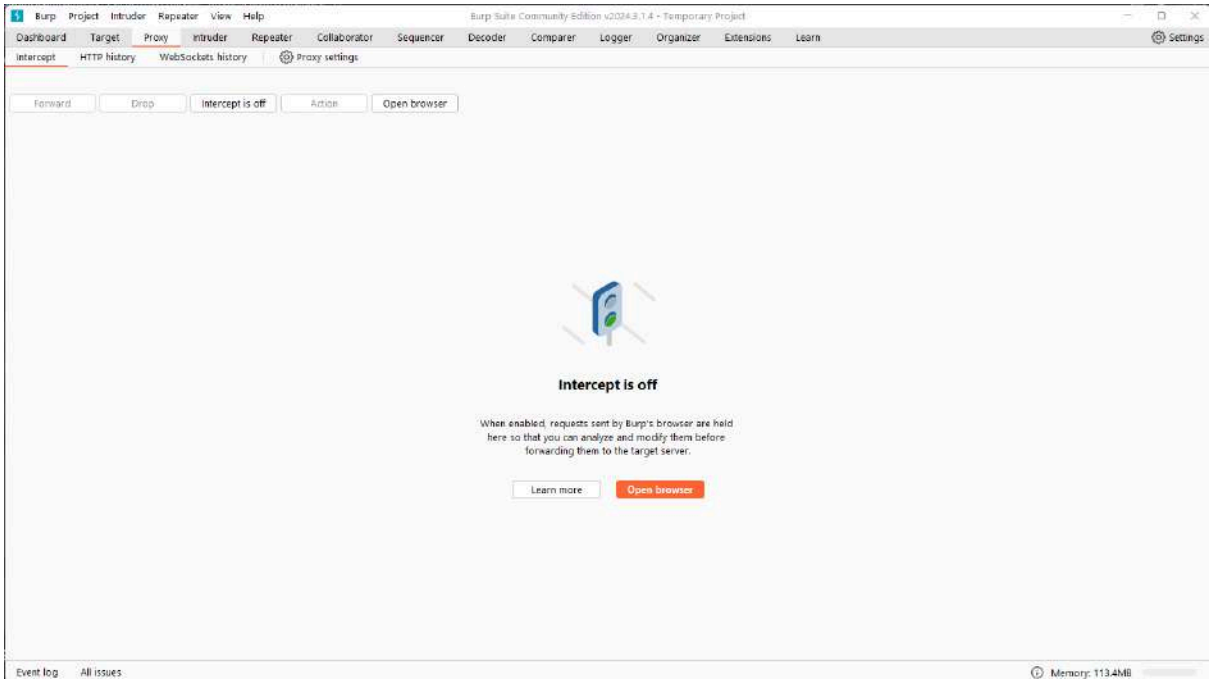
We can use any credentials but need an actual phone number to obtain the message containing the tampered price.

Step-8: Now proceed and go to the checkout page.

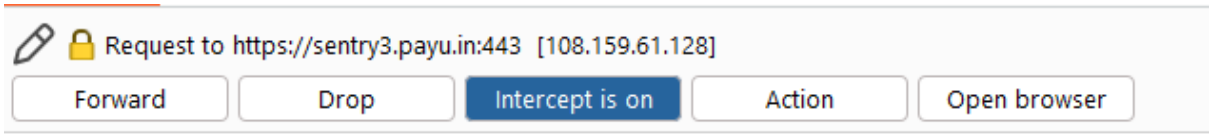


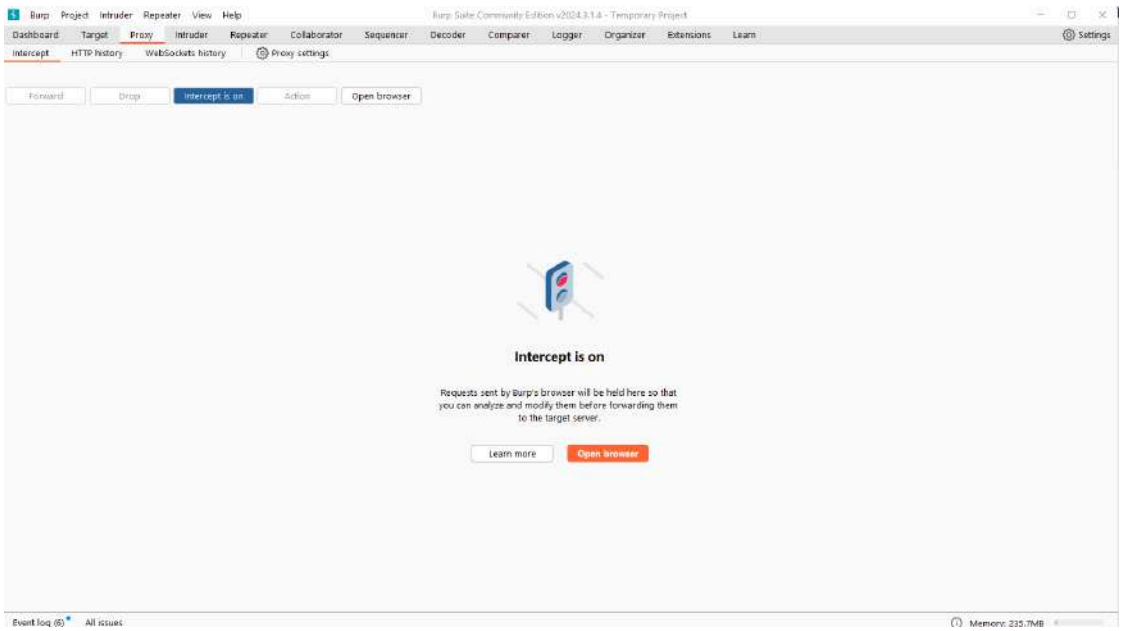
Note that the actually that was supposed to be paid is 7575 INR.

Step-9: Now go to burp suite

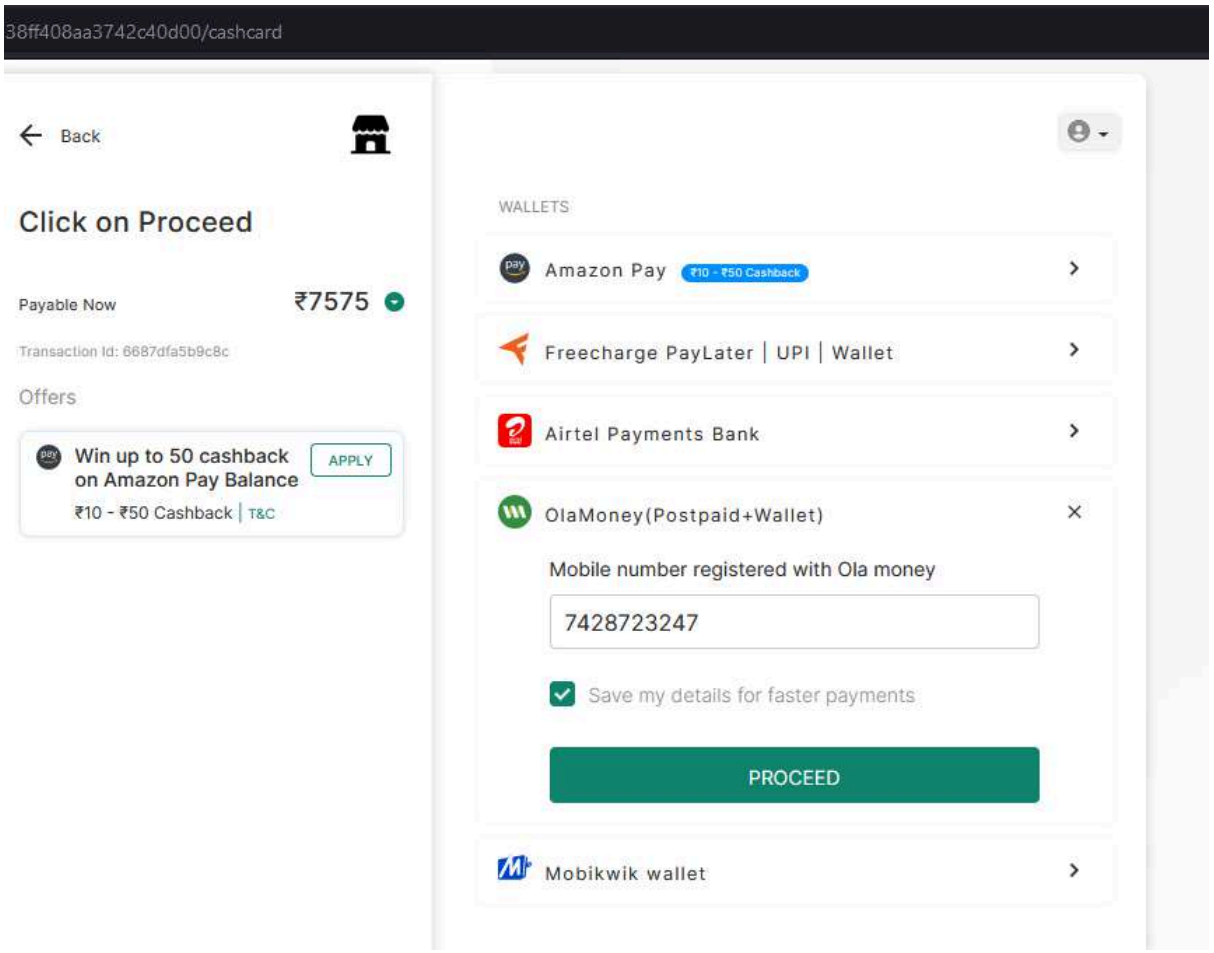


Step-10: Turn on the intercept



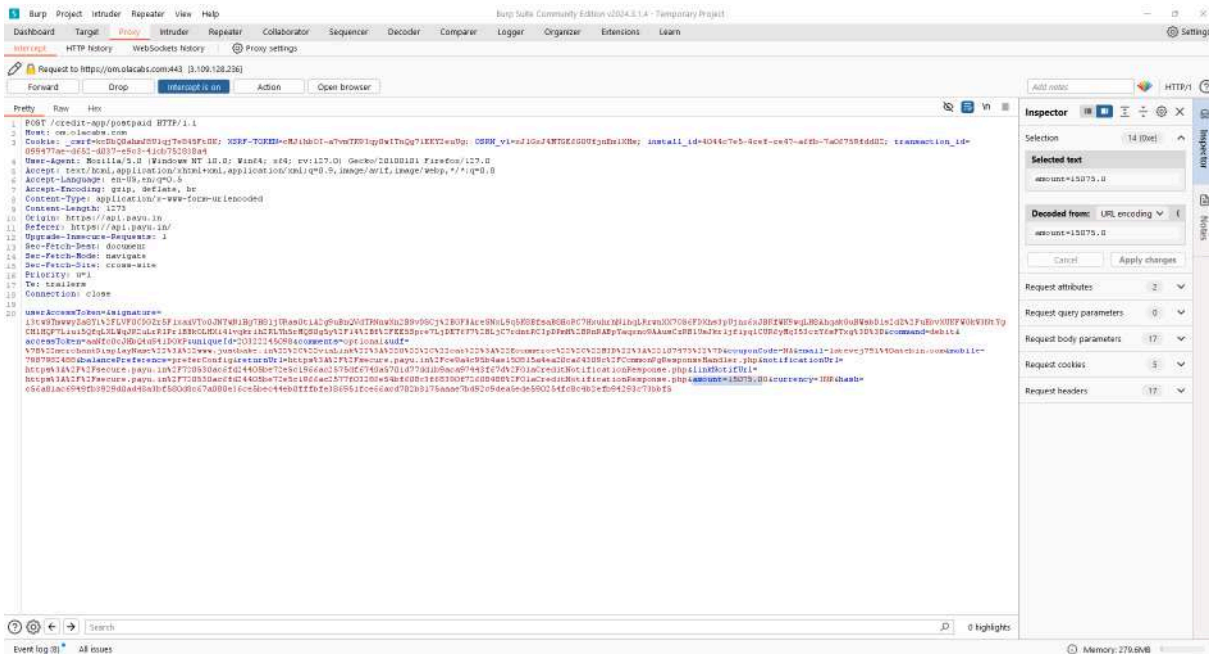


Step-11: Simultaneously select any payment option on the website.



For this i have used ola money

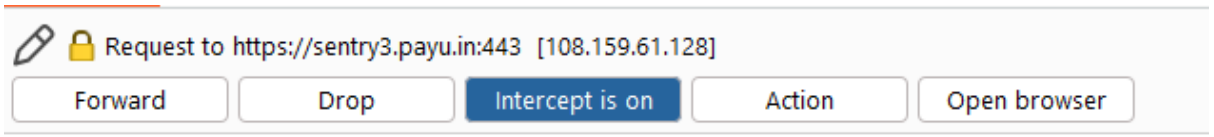
Step-12: Now if we go to the burp suite now we will see entries.



Step-13: Here find the section where the amount that is supposed to be paid is mentioned and change it to a desired value(here I have updated it to 1 INR). If not found click on forward and check again.

```
%22%3A%22107473%22%7D&couponCode=NA&email=lat
FCommonPgResponseHandler.php&notificationUrl=
onResponse.php&linkNotifUrl=
onResponse.php&amount=1.00&currency=INR&hash=
4293c73bbf5
```

Step-14: once done again click on “intercept is on” to turn off the intercept.



PROOF OF CONCEPT

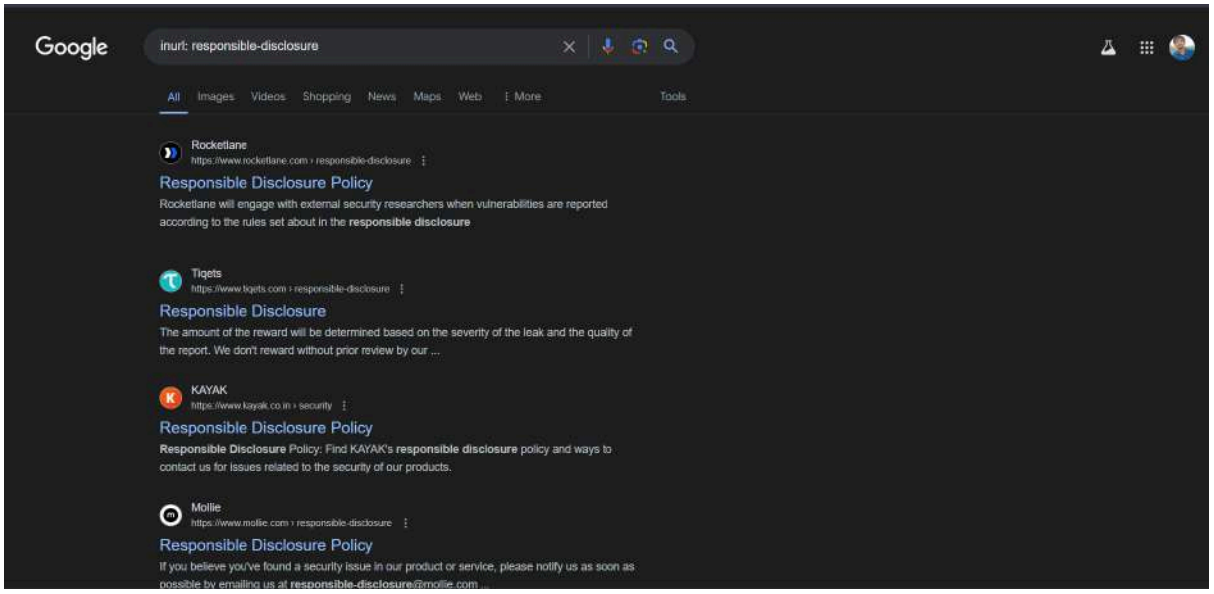
As soon as you turn off the intercept (step-14)you will receive a message on the phone number which was used during the cart checkout (refer to step-7).

We can see that the transaction is altered to Rs. 1.0 instead of Rs. 7575 during the cart checkout (refer to step 8).

TARGET-2(https://store.thelabellife.com/)

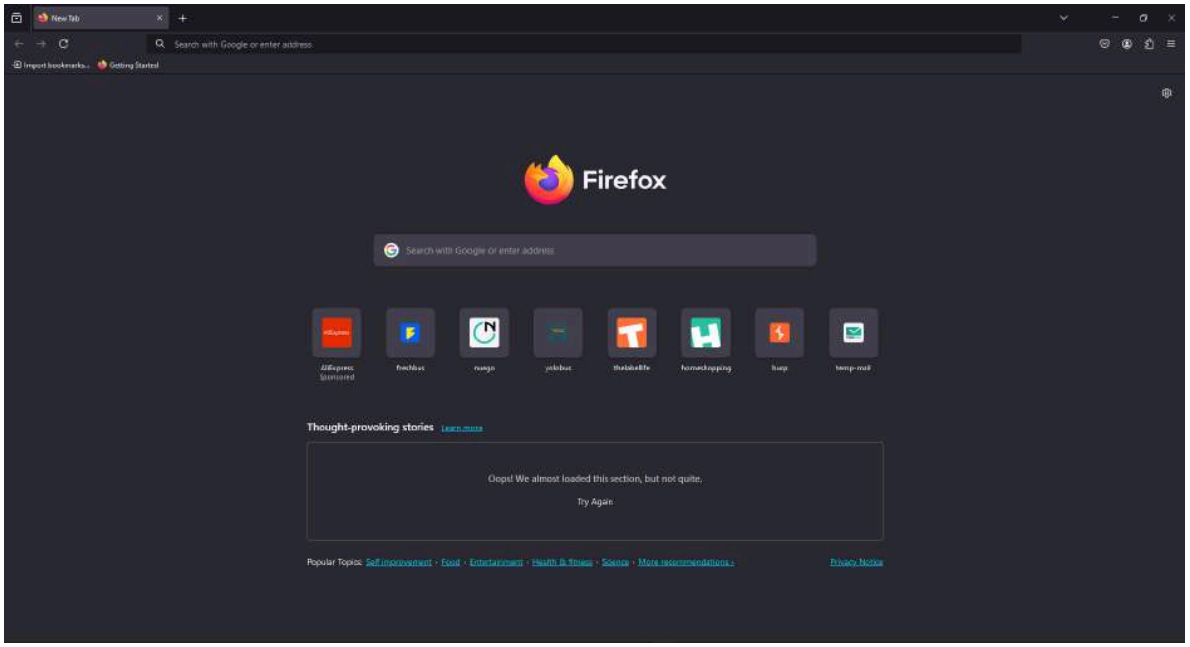
Step-1: Open chrome and use the following google dork

(inurl:Responsible-Disclosure)

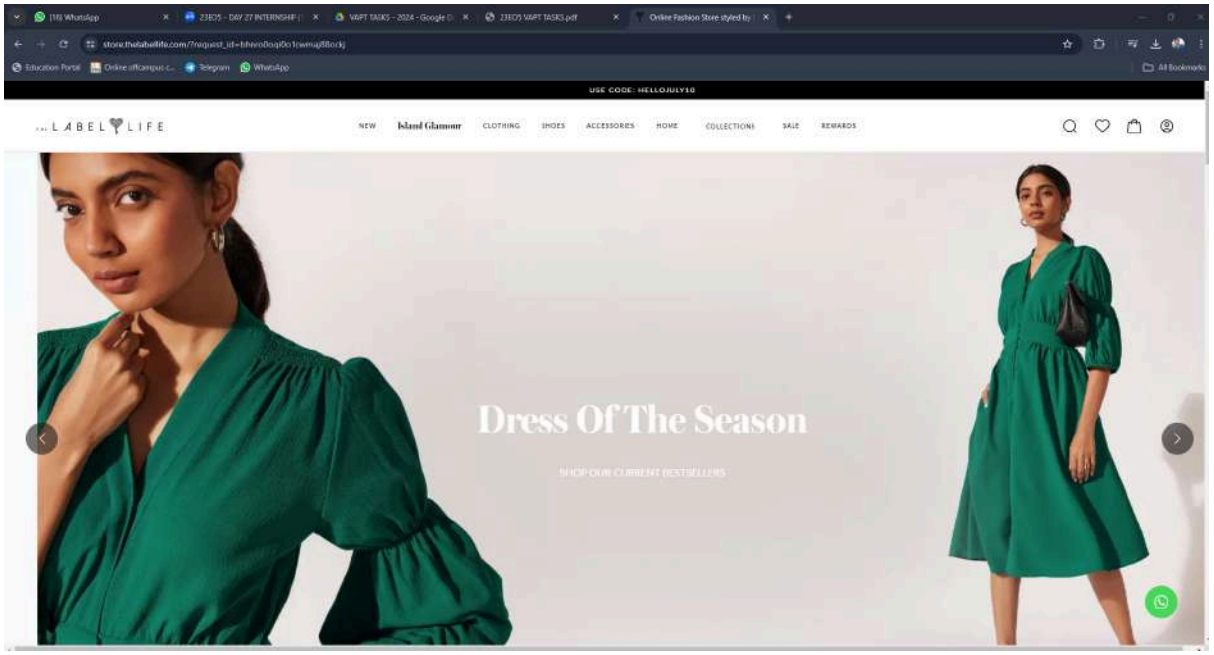


This will give websites which are vulnerable. This will give vulnerable websites.

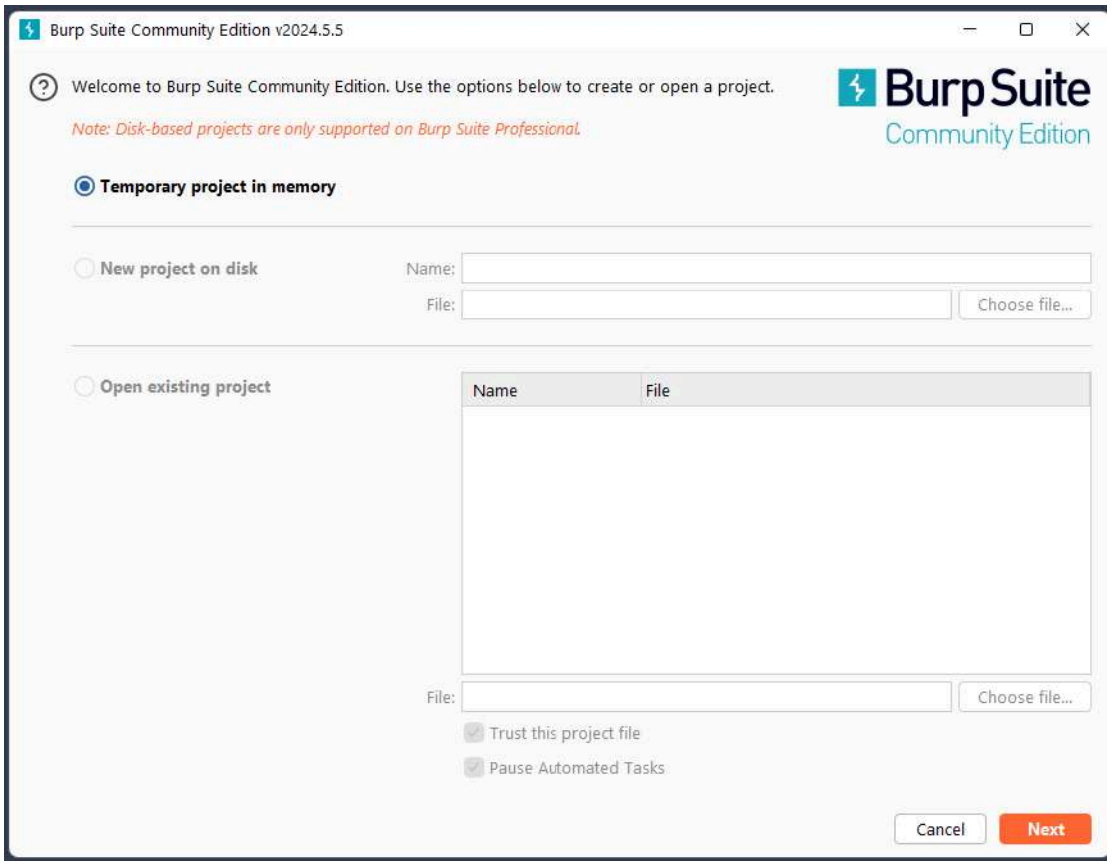
Step-2: Open new tab firefox browser.



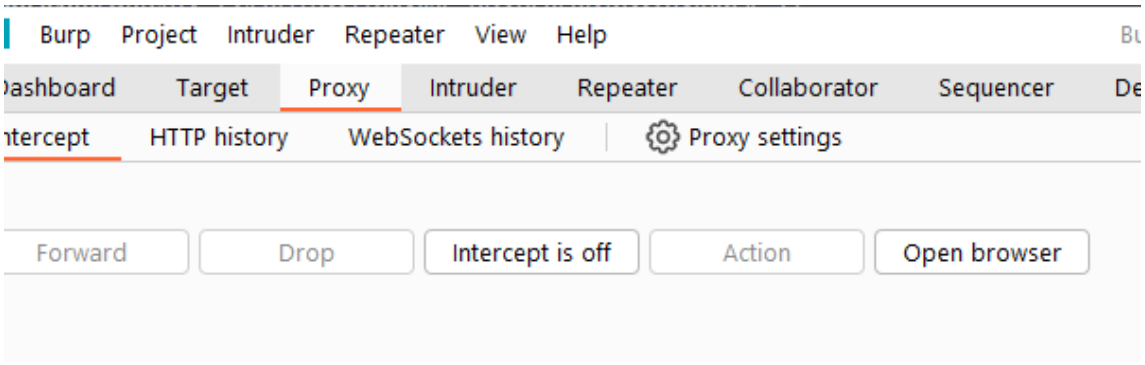
STEP-3: Now, go to https://store.thelabellife.com/.



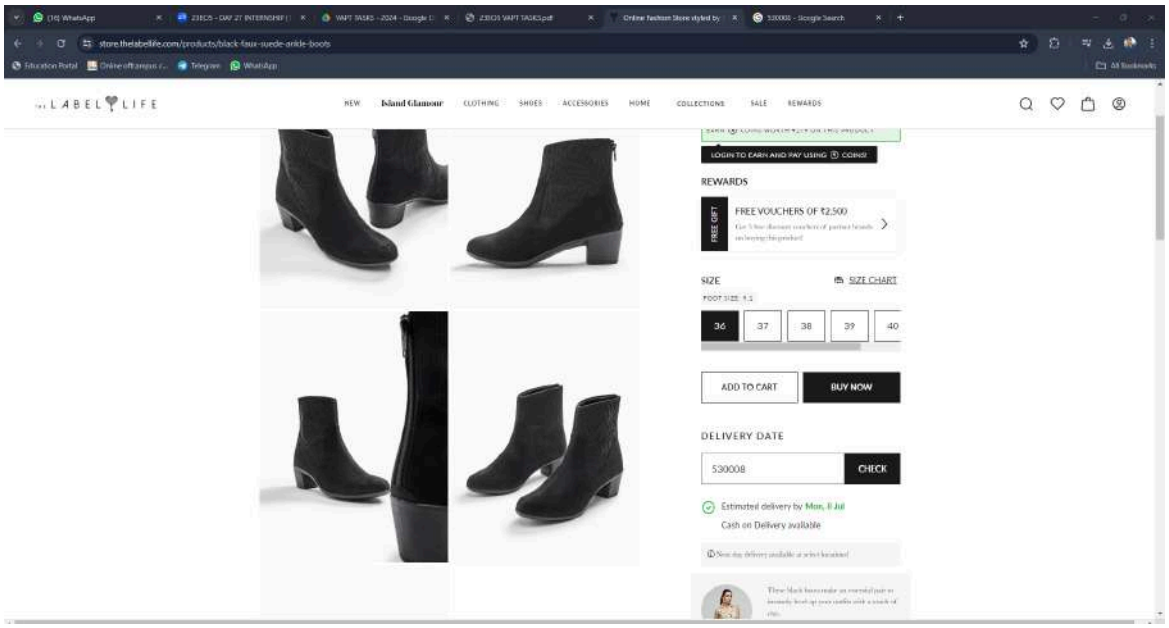
Step-4: Open burp suite community edition



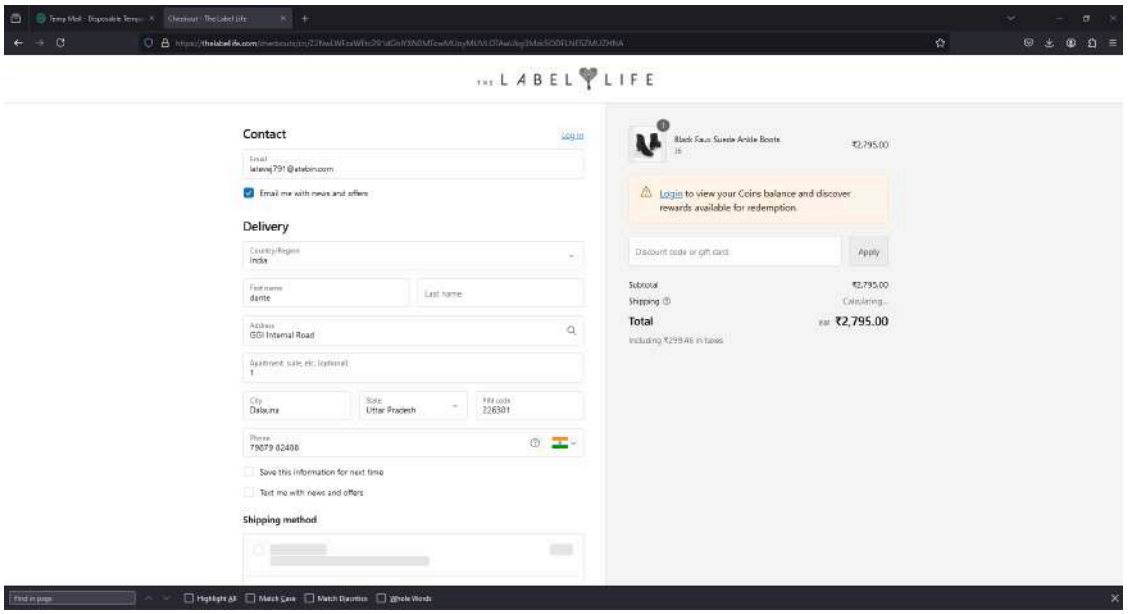
Step-5: Now, go to the proxy tab in burp suite.



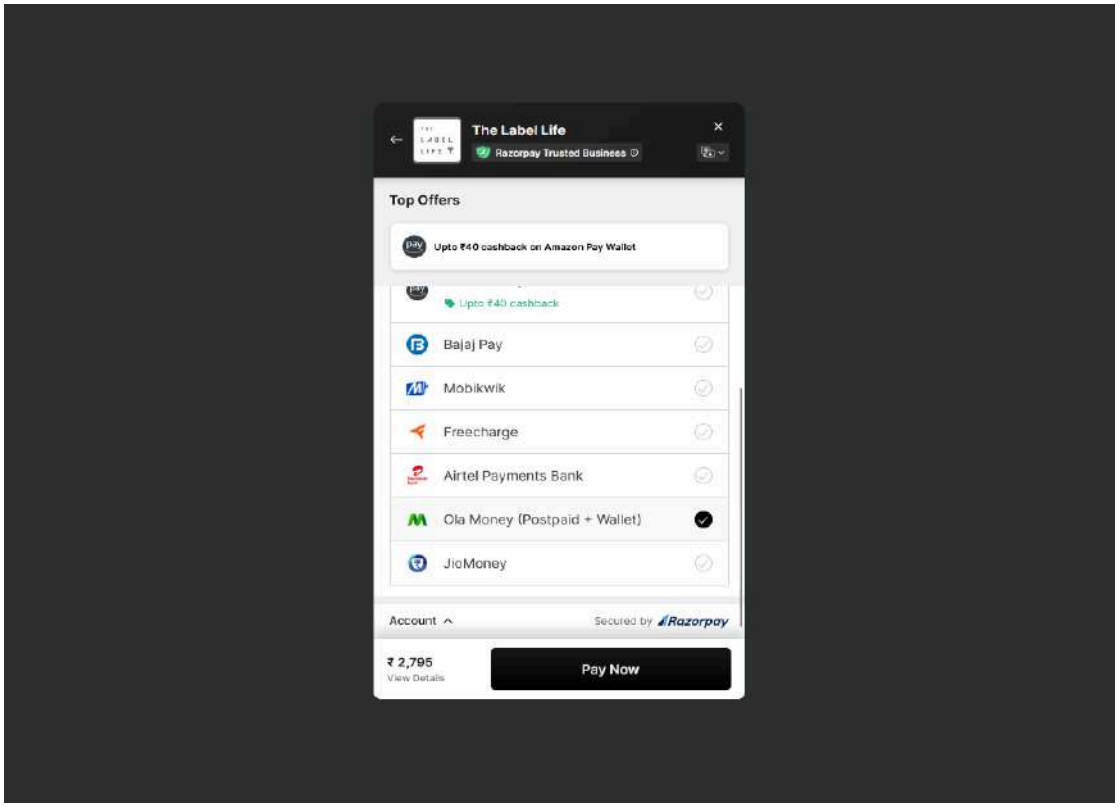
Step-6: Now add any product in the cart.



Step-7: Give **random credentials except phone number** to get the message containing the tampered value.

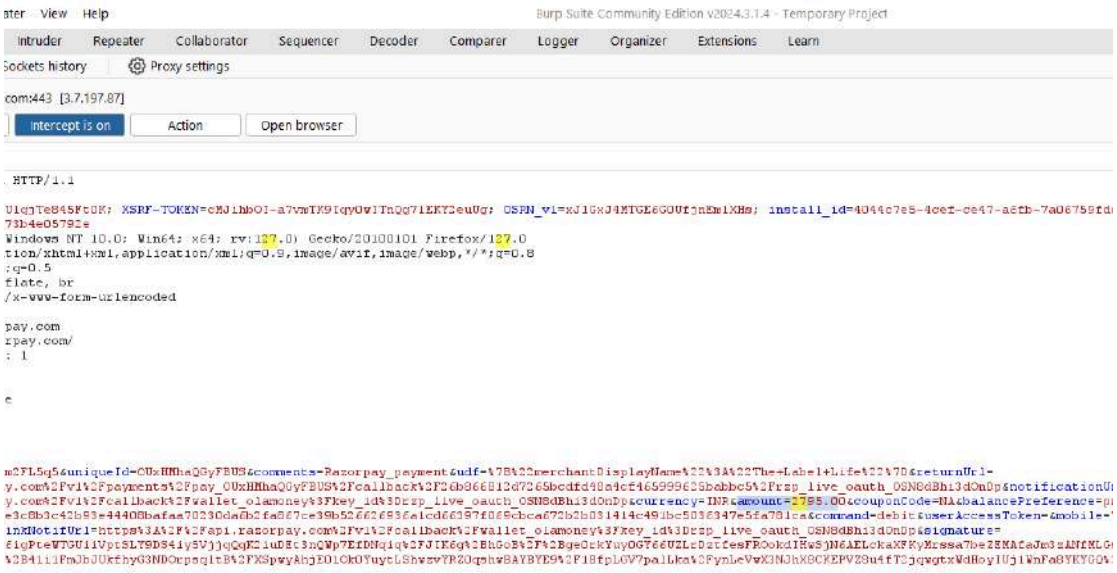


Step-8: Proceed to payment page

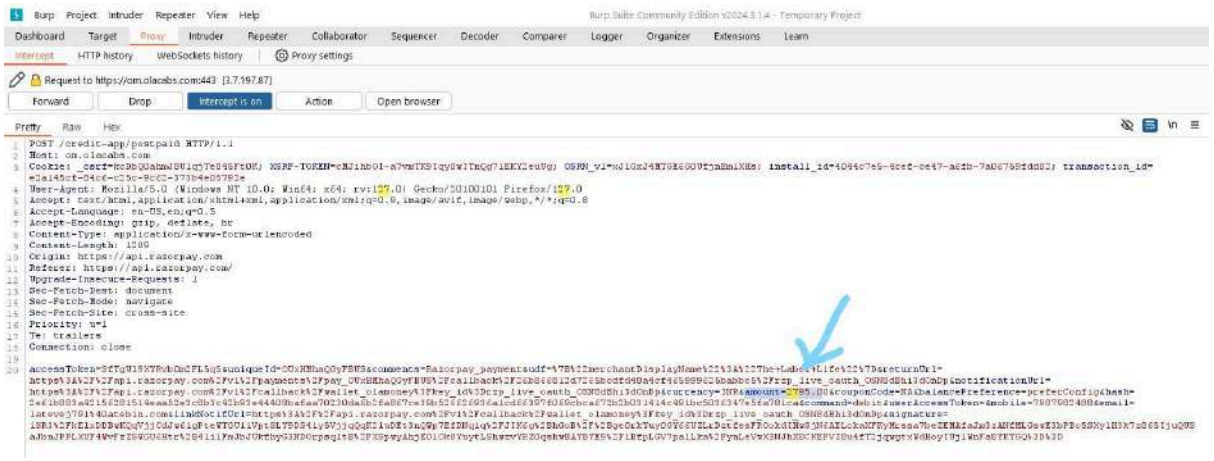


Here I have selected ola money as payment gateway.

Step-09:Simultaneously turn on intercept in burp suite.



Step-10: Look for the amount which is the same as the amount to be paid(Rs. 2785 in this case). If not shown click on forward and check again.

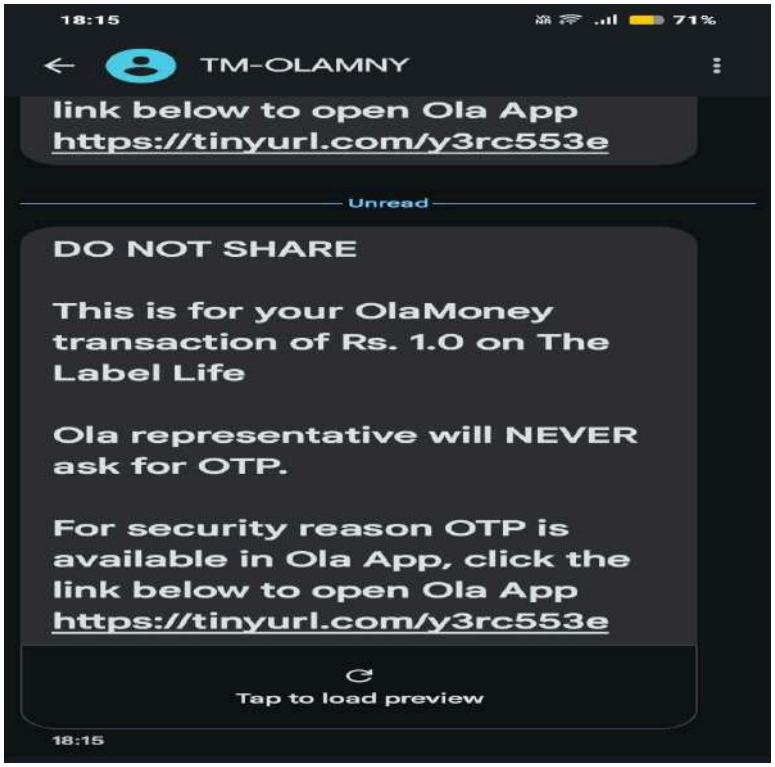


Step-11: Once the amount value is in the sight, select and edit to the desired value you want. In this case i have altered the amount value to Rs.1

```
22merchantDisplayName%22%3A%22The+Label+Life%22%7D&returnUrl=
7265bcdfd48a4cf465999625babbc5%2Frzp_live_oauth_OSN8dBhi3dOnDp&notificationUrl=
OSN8dBhi3dOnDp&currency=INR&amount=1.00&couponCode=NA&balancePreference=preferConfig&hash=
bca672b2b031414c491bc5036347e5fa781ca&command=debit&userAccessToken=&mobile=7987982488&emai
t_olamoney%3Fkey_id%3Drzp_live_oauth_OSN8dBhi3dOnDp&signature=
%2F%2BgeOrkYuyOGY66UZLrDztfesFROokdIHwSjN6AELckaXFKyMrssa7beZEMAfaJm3zANfMLGswE3bPBc5SXylH3
YBYE9%2F1BfpLGV7palLka%2FynLeVwX3NJhX8CKEPVZ8u4fT2jqwgtxWdHoyIUj1WnFa8YKYGQ%3D%3D
```

PROOF OF CONCEPT

Once the value is altered, turn off the intercept and check for the phone number provided in step-7 for the message.



C. Perform Authentication Bypass Exploitation on any website and Prepare clear Documentation.

OTP Bypassing

AUTHENTICATION BYPASS EXPLOITATION

- **CVSS Score** - 9.1 (Critical)
- **Related OWASP Top 10** - A07:2021 - Identification and Authentication Failures
- **Explanation**: Authentication Bypass is a critical security vulnerability that allows attackers to gain unauthorized access to protected resources or functionality without providing valid credentials. This vulnerability occurs when the authentication mechanism in an application can be circumvented, tricked, or compromised. Attackers exploit weaknesses in the authentication process to assume the identity of legitimate users, potentially gaining access to sensitive data or administrative functions.
- **Impact**:

1. Unauthorised access
2. Data breach
3. Privilege escalation: Attackers may gain administrative or elevated user privileges.
4. Identity theft: Impersonation of legitimate users, leading to potential fraud or misuse of accounts.
5. Regulatory non-compliance: Violation of data protection regulations (e.g., GDPR, HIPAA) due to unauthorised access.

- **Recommendations**:

1. Implement strong authentication mechanisms: Use multi-factor authentication (MFA) where possible.
2. Secure session management: Implement secure session handling and token generation.
3. Use secure protocols: Employ HTTPS for all authentication-related traffic.
4. Implement proper password policies: Enforce strong password requirements and secure password storage (e.g., using bcrypt or Argon2).
5. Account lockout mechanisms: Implement account lockout after a certain number of failed login attempts.

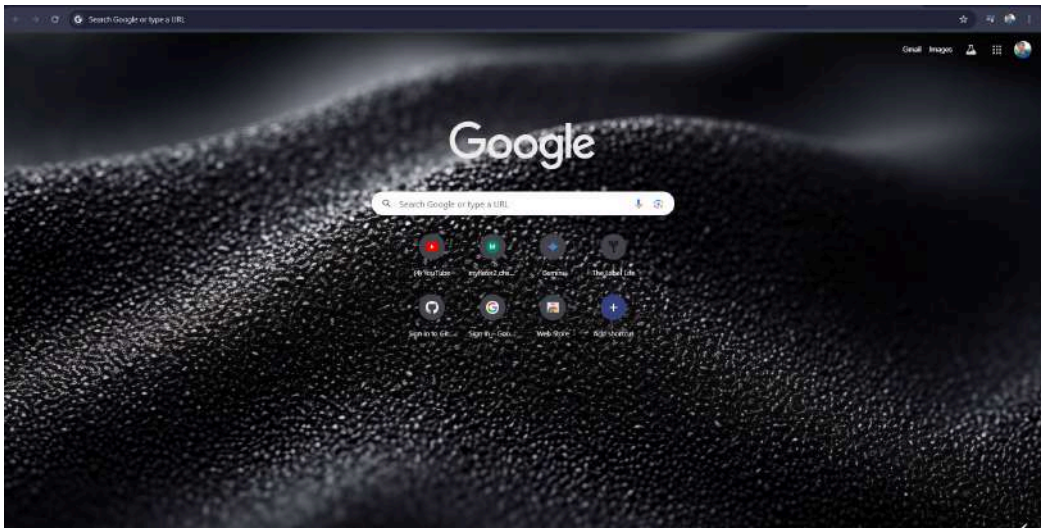
- **References**:

1. OWASP Authentication Cheatsheet

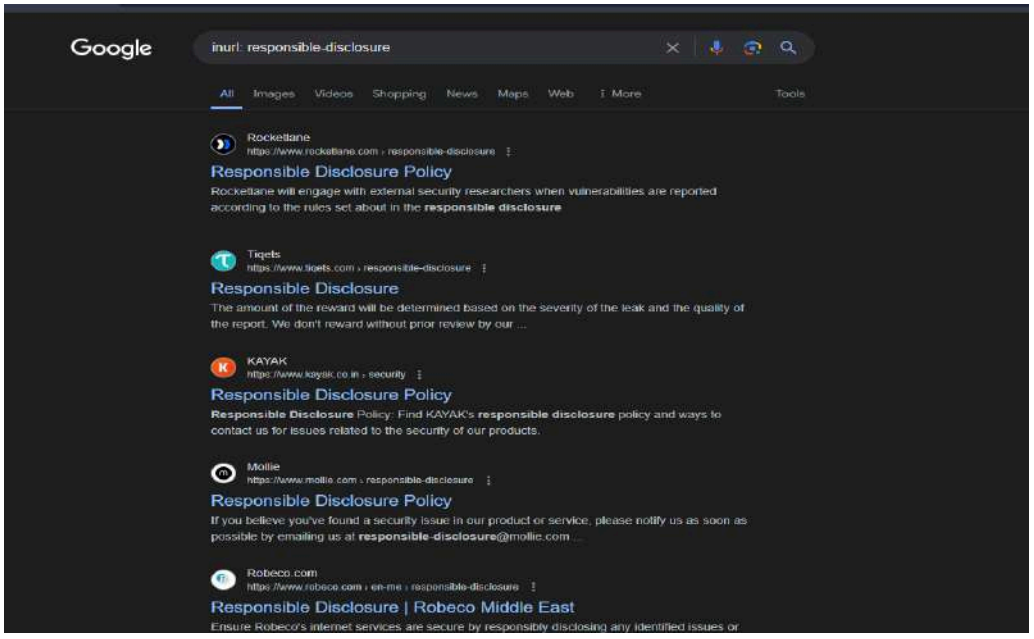
- **Procedure:**
    - **Target Website -** https://www.justbake.in/
    - **Payload -** Host: www.facebook.com
    - **Steps -**

PROCEDURE

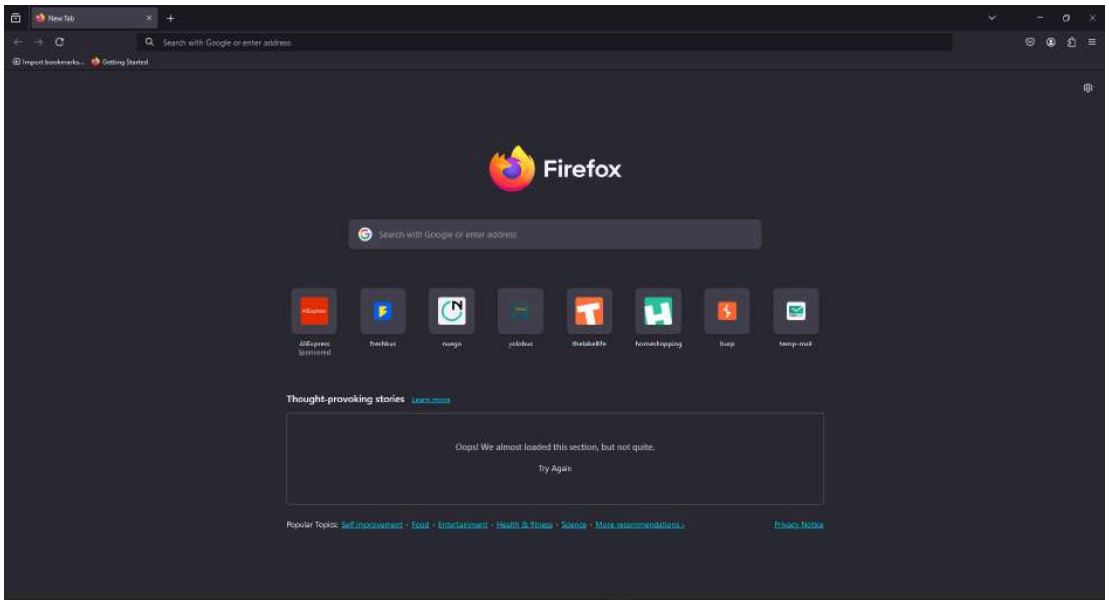
STEP-1: Open google chrome.



Step-2: Search for “inurl:Responsible-Disclosure”.



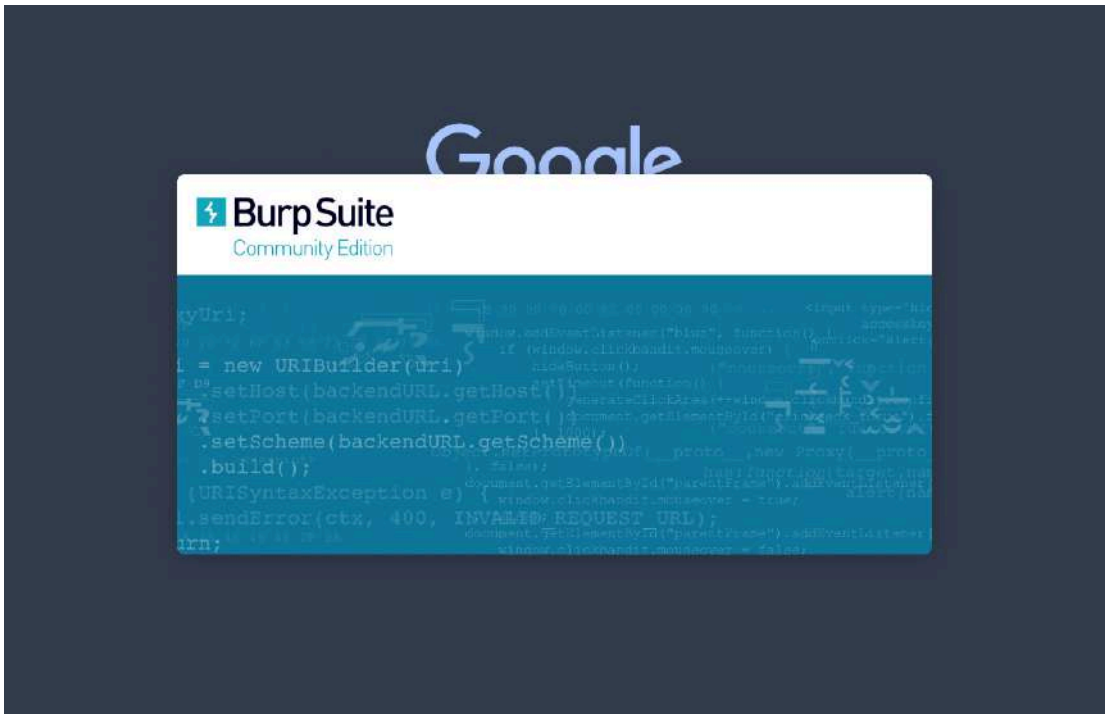
Step-03: Now open firefox browser.



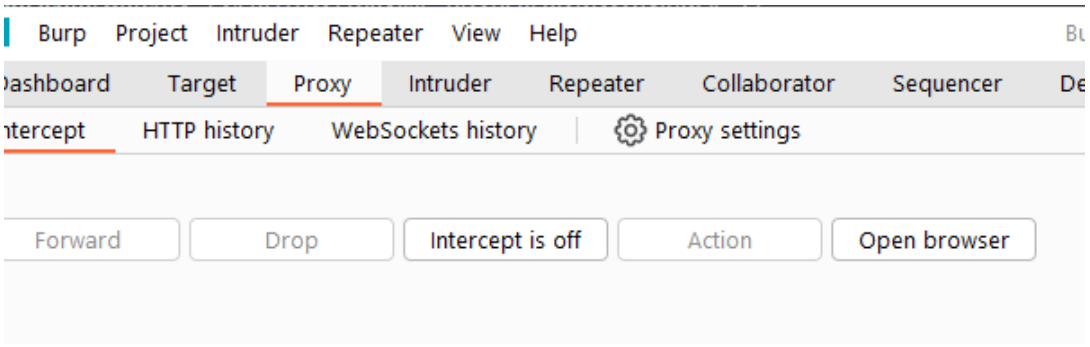
Step-4: Here search the vulnerable website obtained from google chrome.

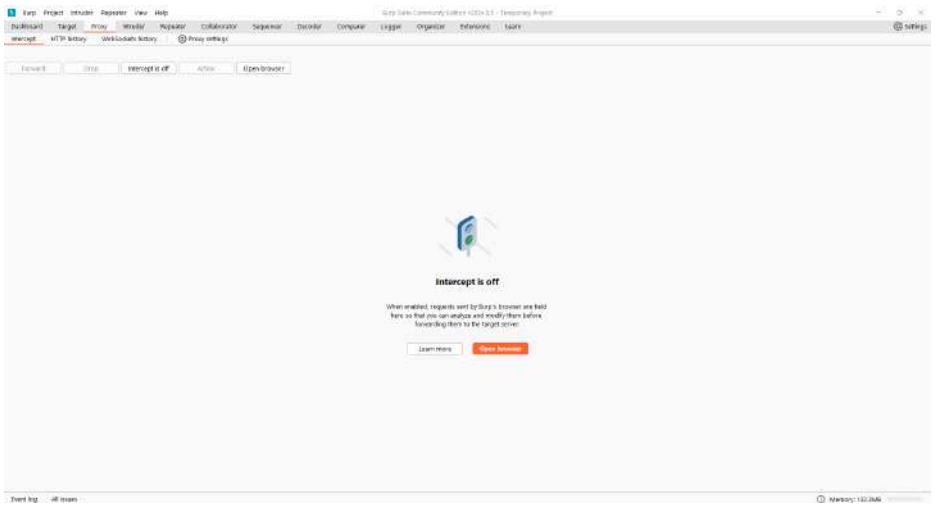
(Here i am performing the practical on https://www.justbake.in/)

Step-5: Open burp suite community edition

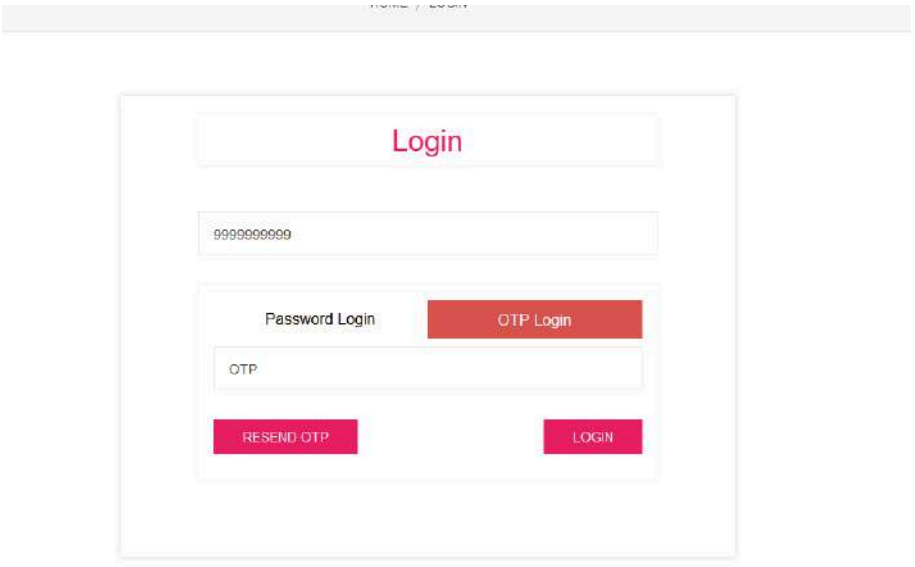


Step-6: Now, go to the proxy tab in burp suite.



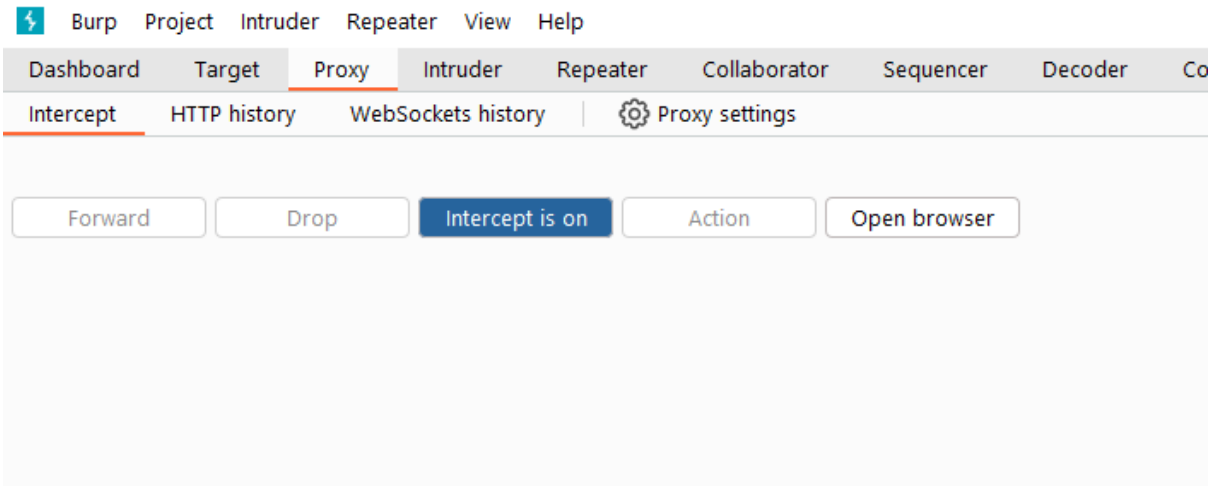


Step-7: Now go to the login page of the website and enter any random phone number and login otp.



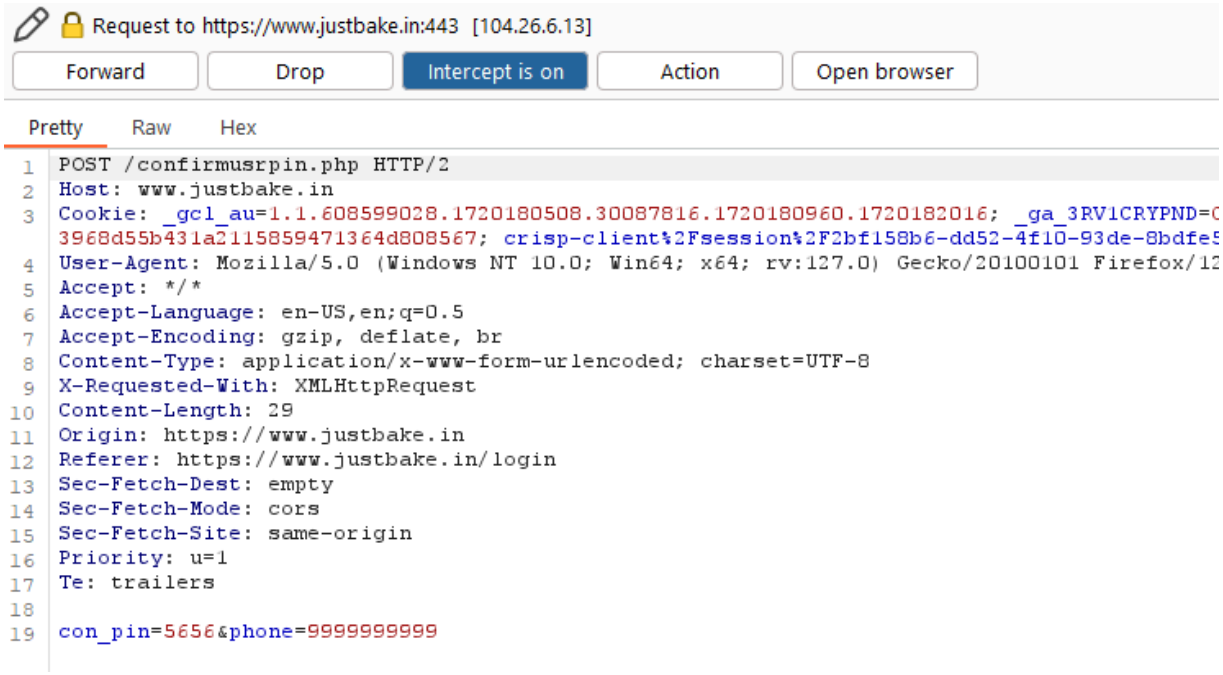
Don’t click on LOGIN.

Step-8: before clicking on the login button, Turn on the intercept in the burp suite tool.

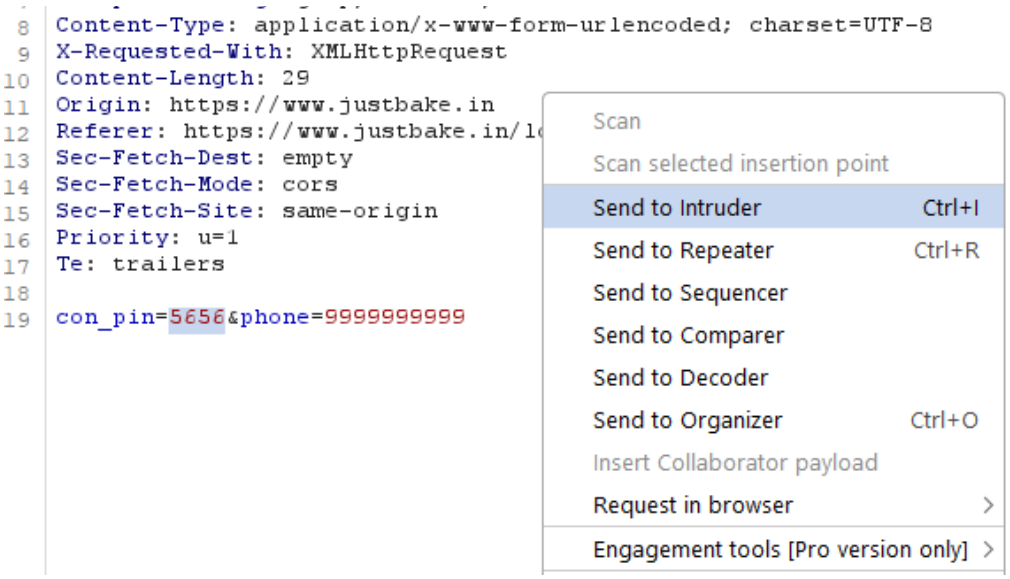


After turning on the intercept click on the login button mentioned in the above step.

Step-9: The entries in the burp suite will start to appear.

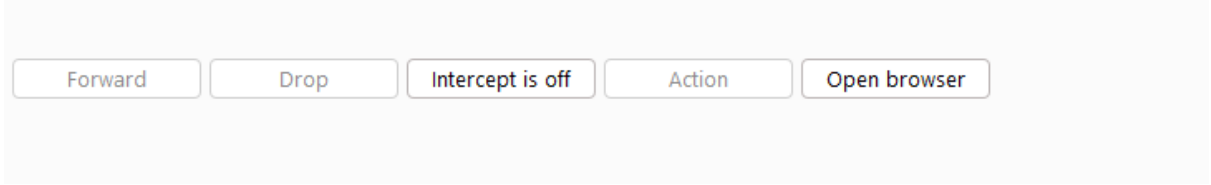


Step-10: Now look for the credentials used in step-7. To be precise, find the otp that was entered and select it.

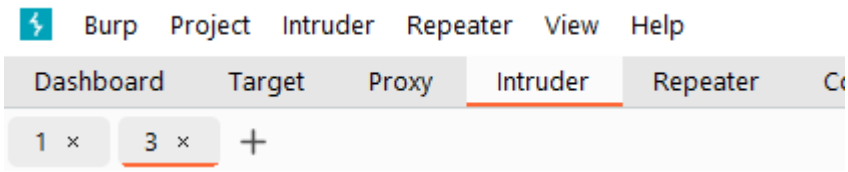


Right click on it and click on “Send to intruder”.

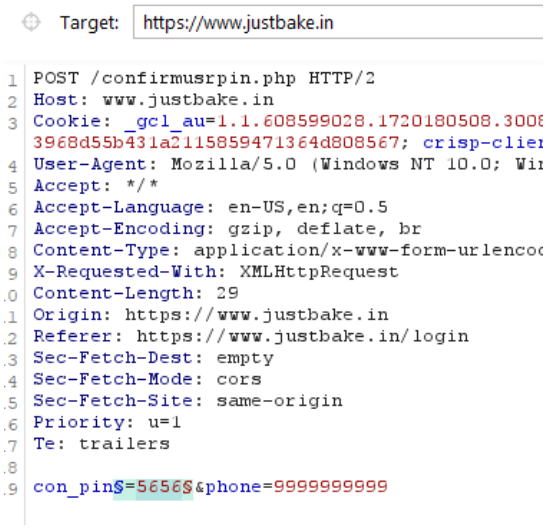
Step-11: turn off the intercept.



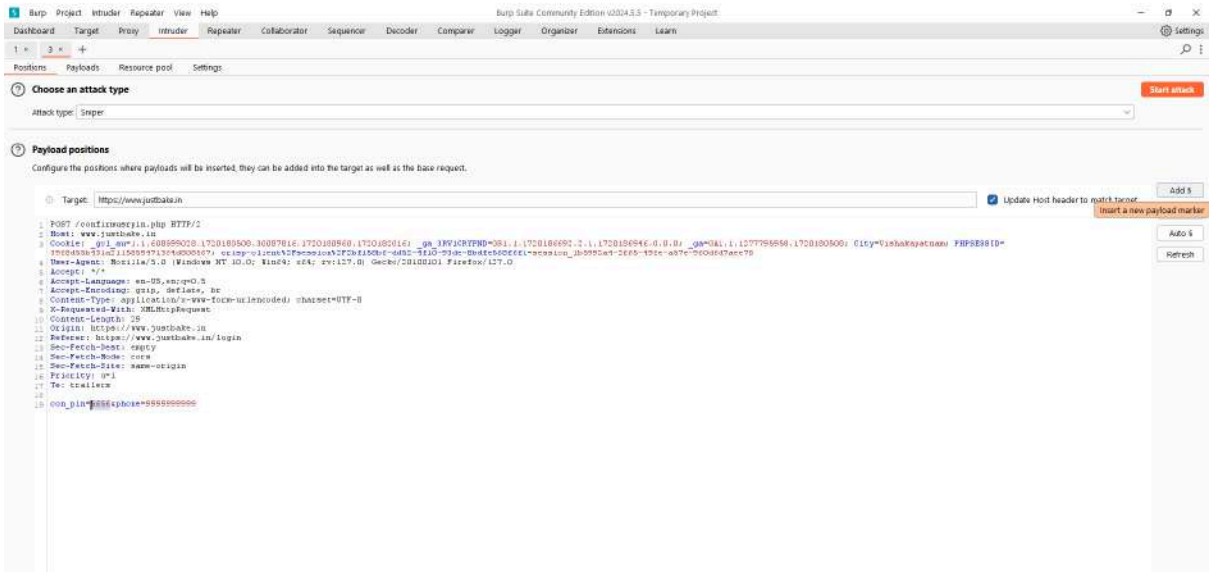
Step-12: Go to the intruder section of the burp suite.



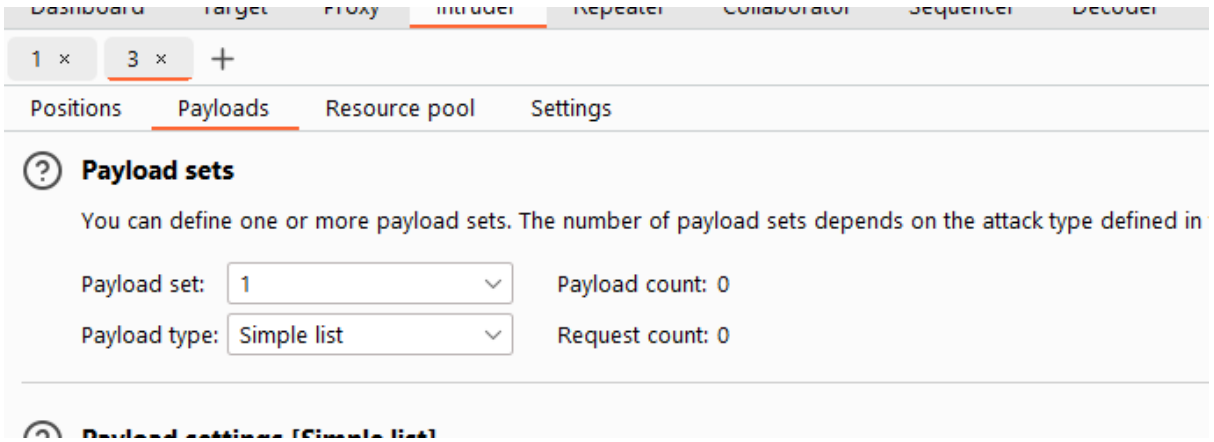
Step-13: Again look for the otp entered.



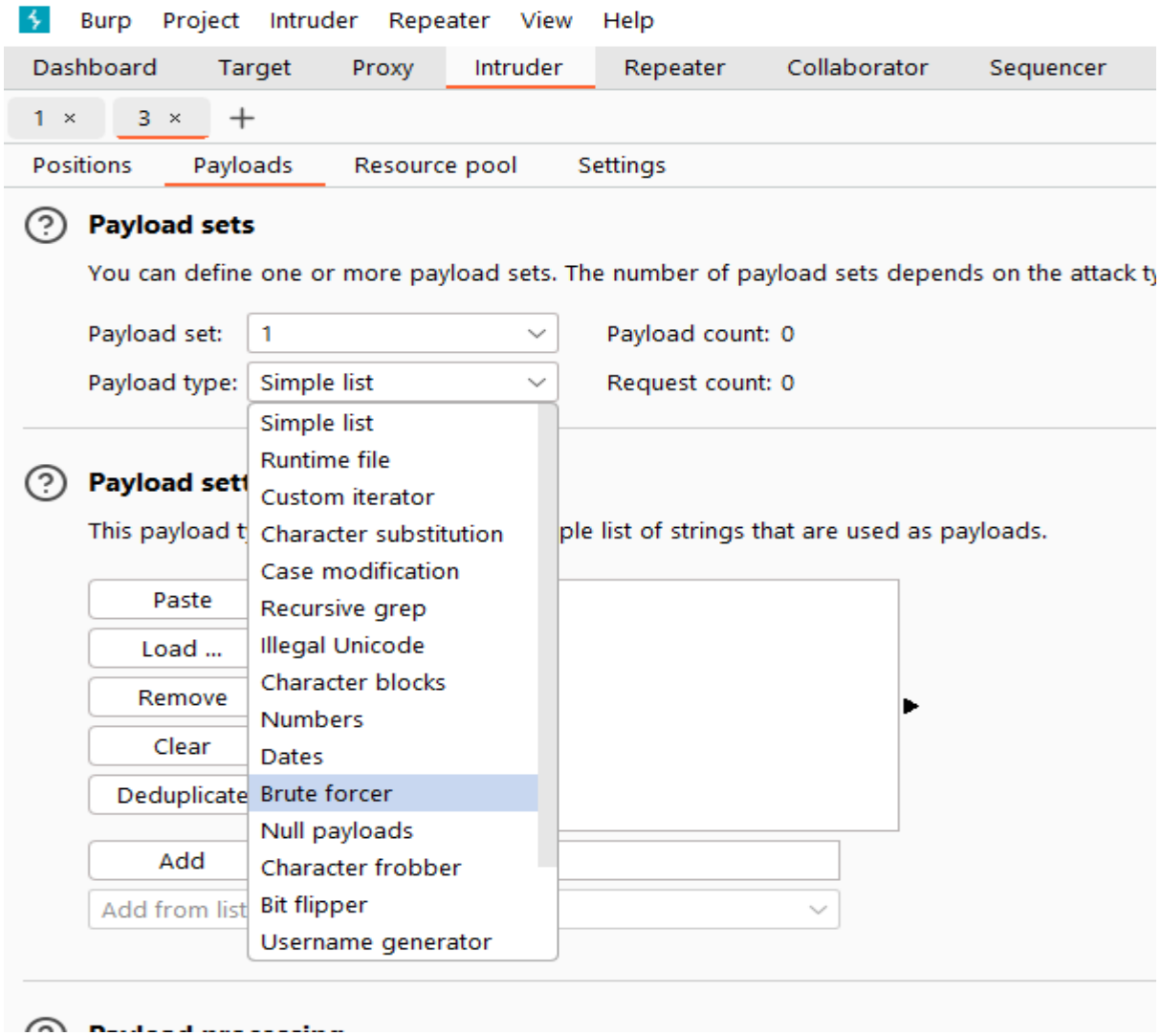
Step-14: Select the otp and click on add button on the right side of screen.



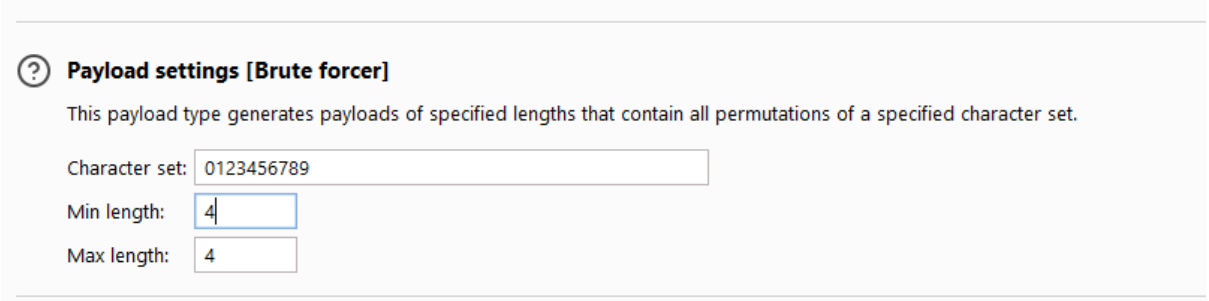
Step1: Now go to the payload section in intruder.



Step-16: Change the payload set to “brute force”.



Step-17: Change the payload settings same as given below.



Character set: 0123456789

Min Length: 4

Max length: 4

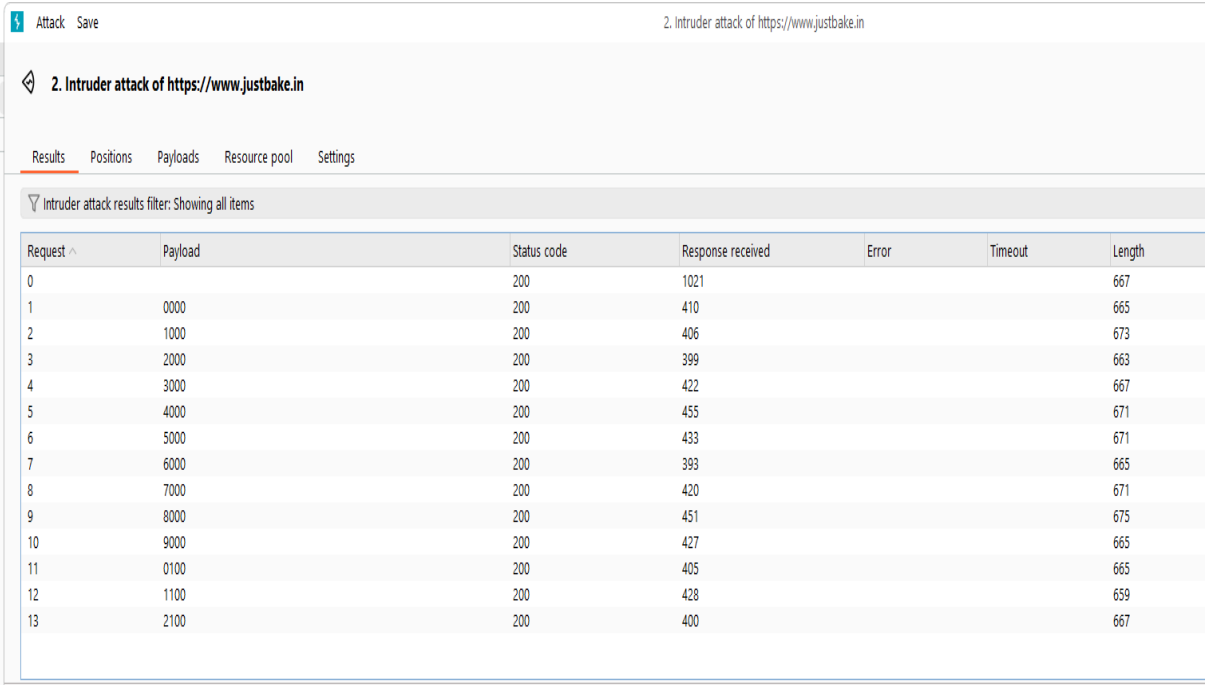
The min length and max length depends on the target web application. If the target requires a 6 digit otp then set the values as follows.

Character set: 0123456789

Min Length:6

Max Length:6

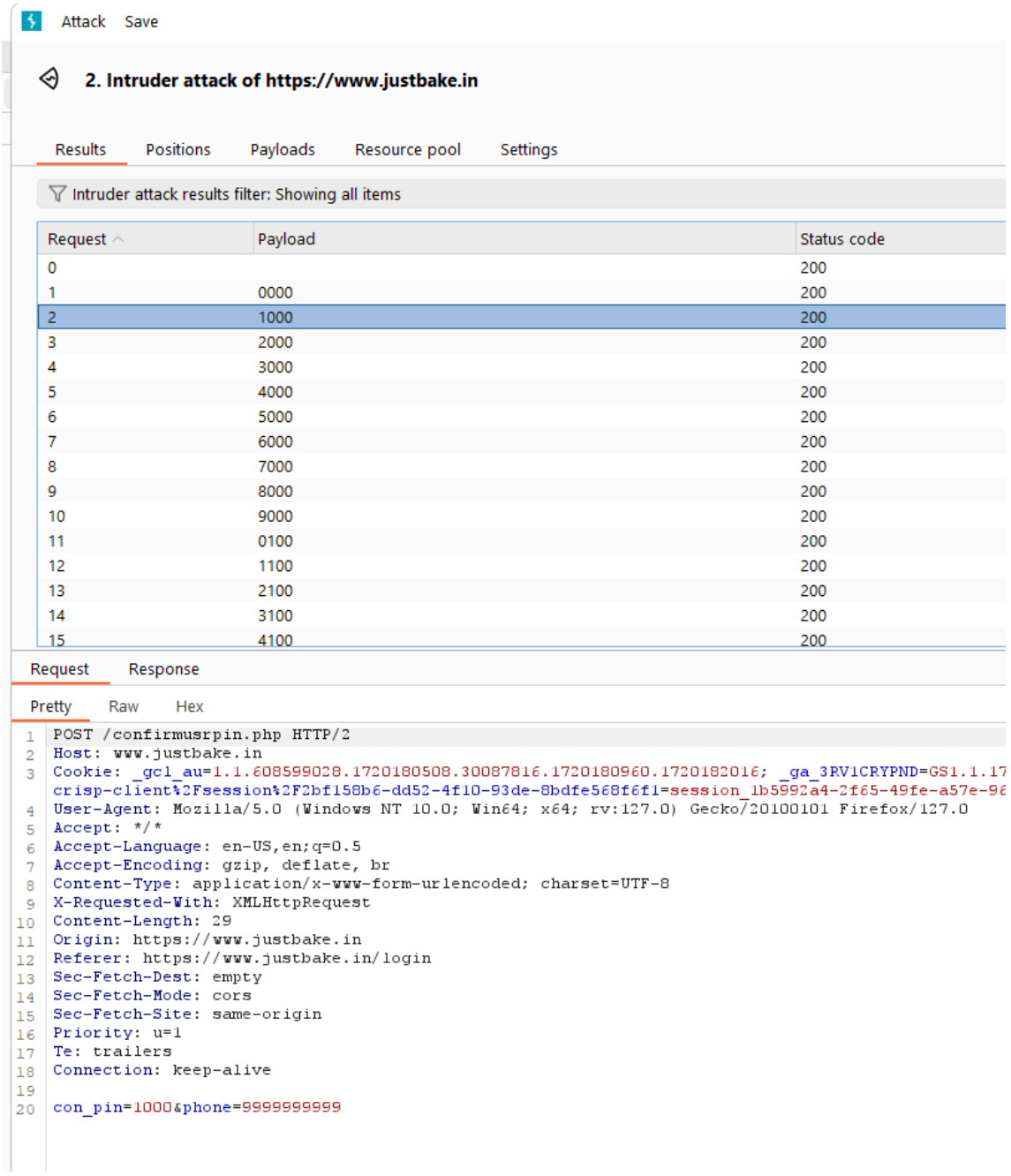
Step-18: Once done click on the “start attack” button on the top right corner.



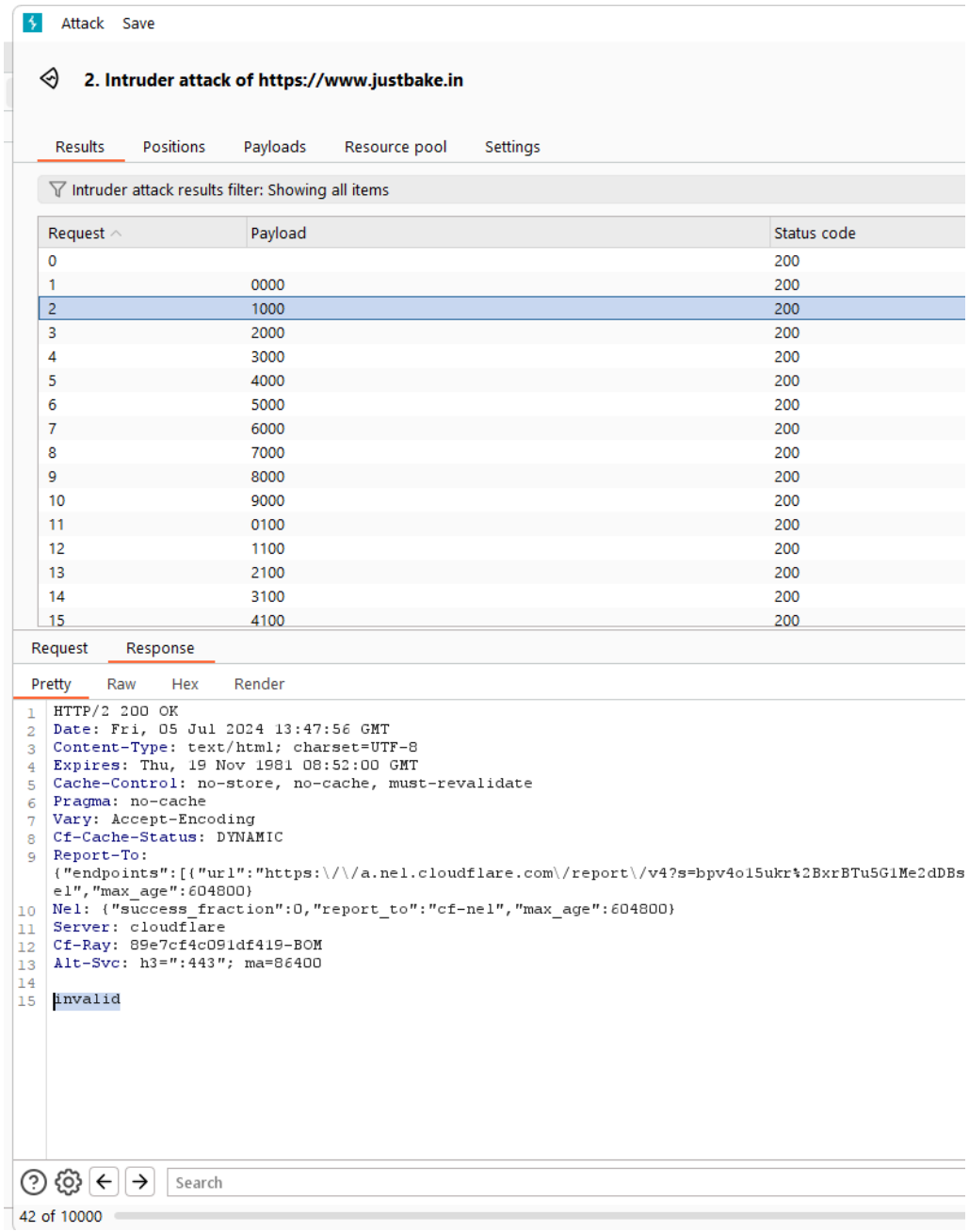
Request	Payload	Status code	Response received	Error	Timeout	Length
0		200	1021			667
1	0000	200	410			665
2	1000	200	406			673
3	2000	200	399			663
4	3000	200	422			667
5	4000	200	455			671
6	5000	200	433			671
7	6000	200	393			665
8	7000	200	420			671
9	8000	200	451			675
10	9000	200	427			665
11	0100	200	405			665
12	1100	200	428			659
13	2100	200	400			667

It will start generating requests.

Step-19: Now click on a request.

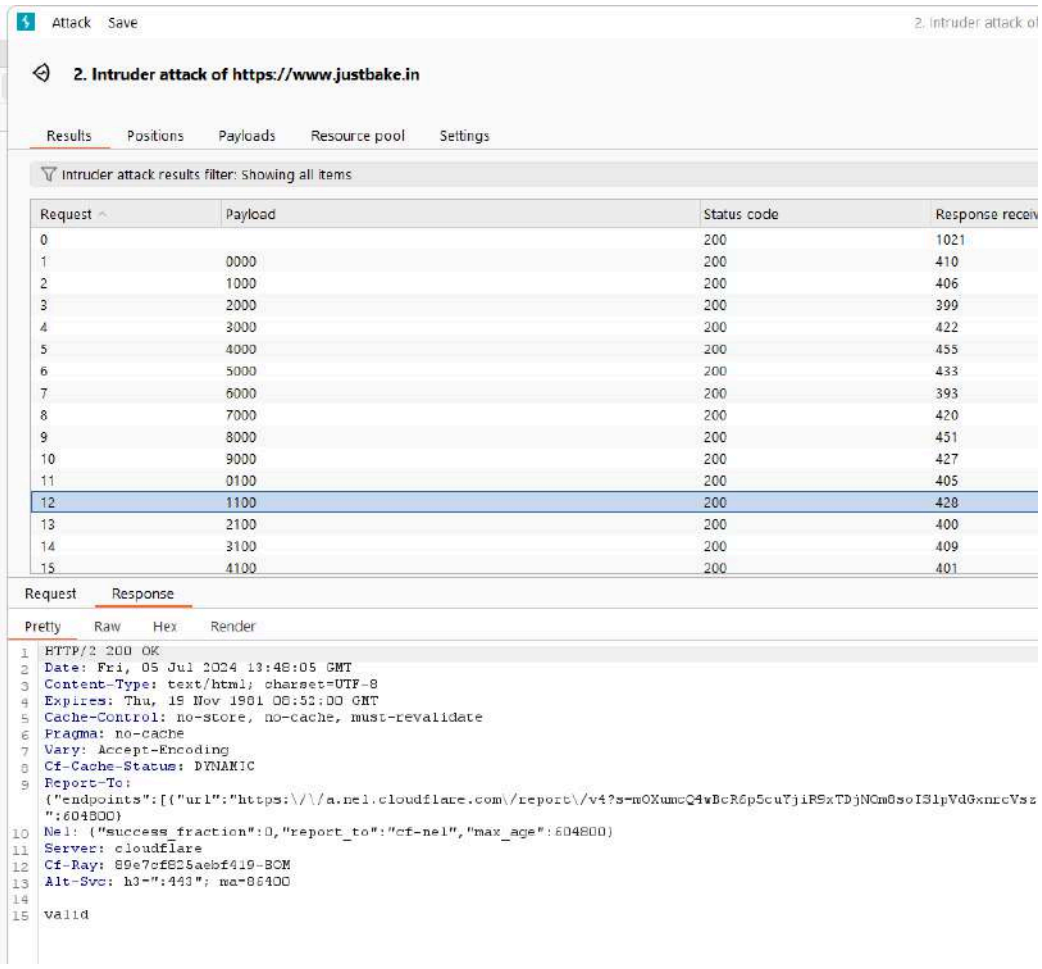


Step-20: Check the response tab for the validity of the request.



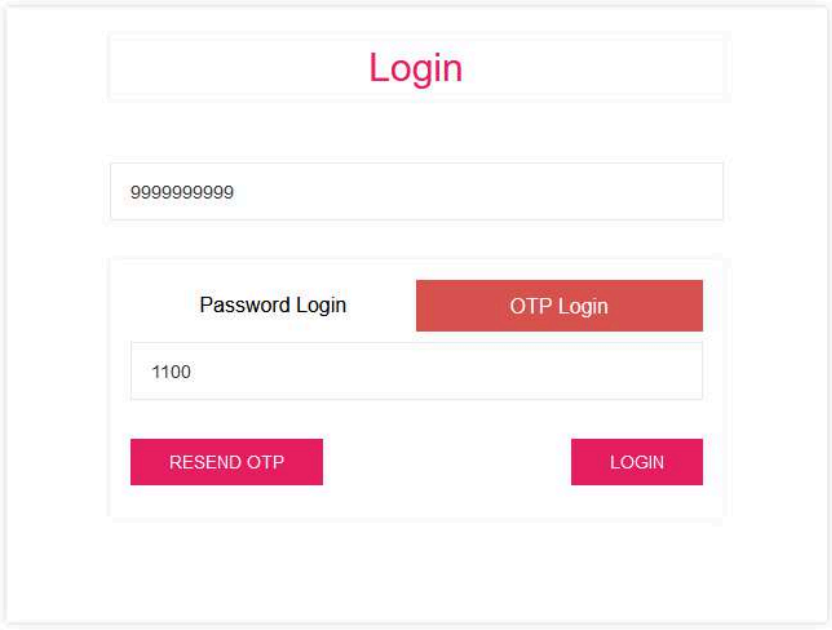
This request is invalid.

Step-21: Repeat steps 19-20 for every request until a valid request is obtained.



Here request 12 is valid.

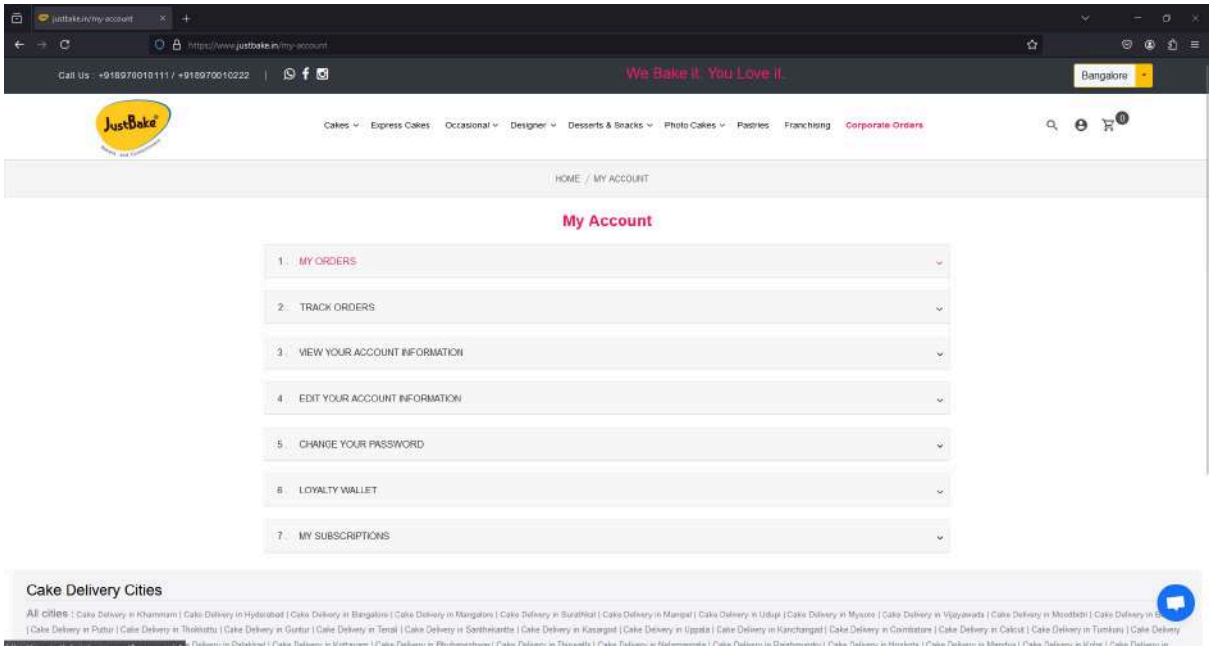
Step-22: Copy the payload corresponding to a valid request and paste it in the otp section of the target (refer to step 7) and click on login.



In this case request 12 was valid so we copied the payload corresponding to (1100) and pasted it in the login page.

PROOF OF CONCEPT

After Step-22 the login would be successful.



Here the payload 1100 was valid and we used it to login to the web application.

Supraja Technologies

23E05-ST#IS#6653-TASK11

A. Find a website vulnerable to Host Header Injection Vulnerability.

Host Header Injection Vulnerability

- **CVSS Score** - 7.5 (High)
- **Related OWASP Top 10** - A5:2021 - Security Misconfiguration
- **Explanation**: This vulnerability occurs when an application uses the HTTP Host header value to generate links, import resources, or make security decisions without properly validating or sanitizing it.
- **Impact**:

1. Cache Poisoning: Injecting malicious content into the cache, affecting multiple users.
2. Password Reset Poisoning: Manipulating password reset links to direct users to attacker-controlled sites.
3. SSRF: Server side request forgery.
4. Phishing

- **Recommendations**:

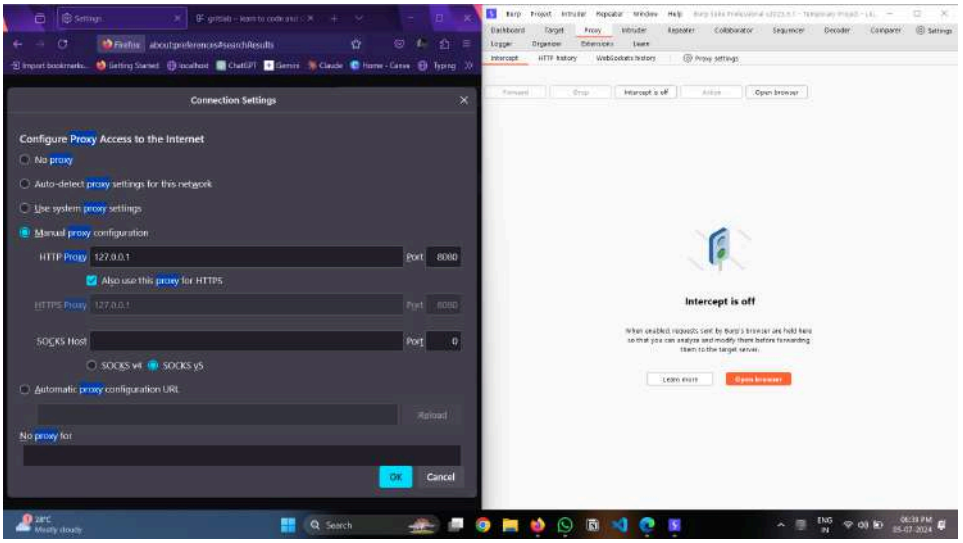
1. Validate and sanitize the Host header
2. Use allowlists
3. Avoid using user-supplied Host headers
4. Implement proper URL parsing
5. Use HTTPS
6. Implement additional security headers like X-Frame-Options, Content-Security-Policy, and Strict-Transport-Security.

- **References**:

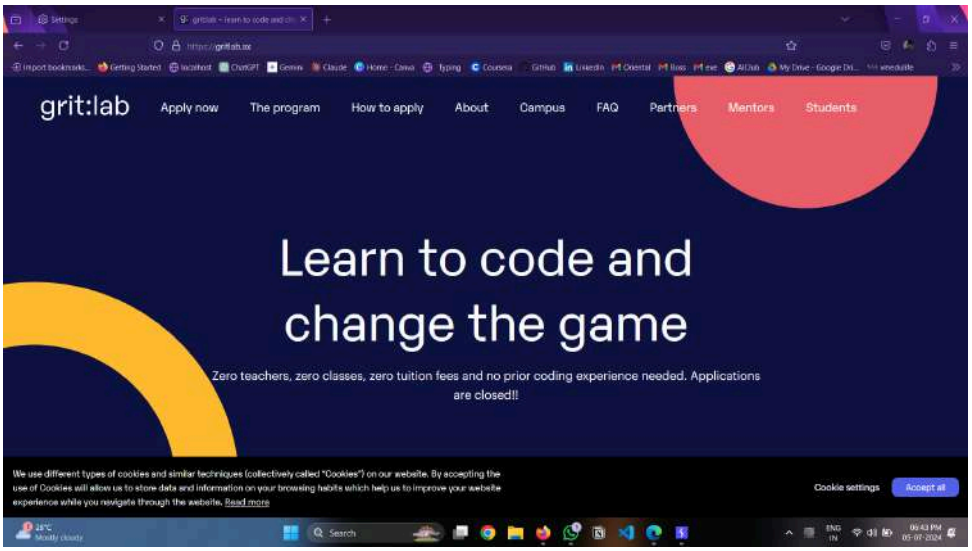
1. OWASP Host Header Attack
2. PortSwigger Research on HTTP Host header attacks

- **Procedure:**
  - **Target Website -** https://gritlab.ax/
  - **Payload -** Host: www.facebook.com
  - **Steps -**
  - The following steps show how I found this vulnerability on this site.

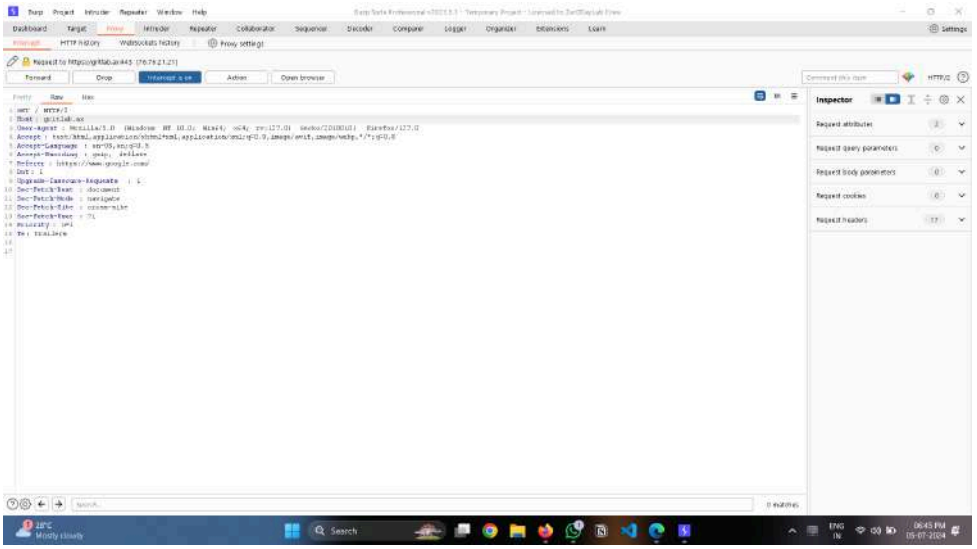
1. **Open BurpSuite and Configure the Proxy settings on Firefox.**



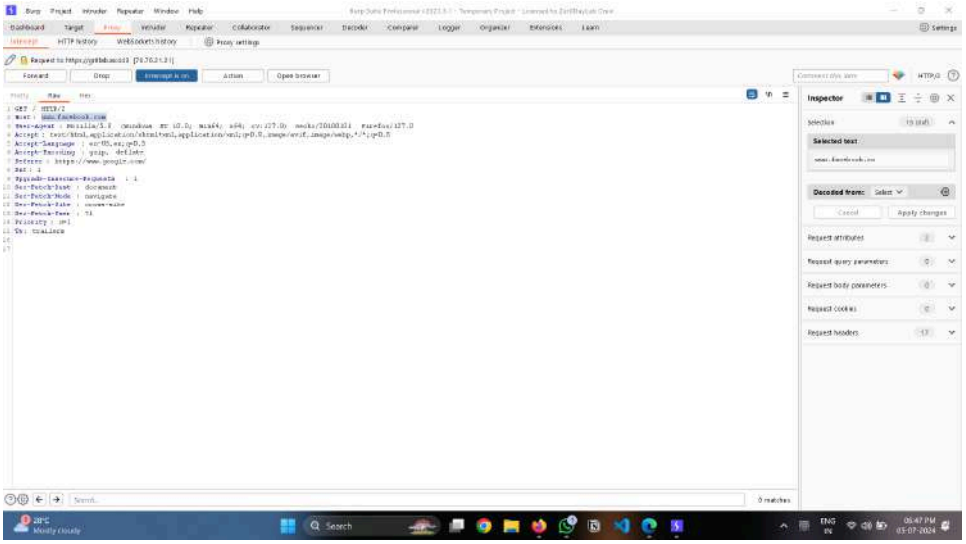
1. **On the Browser, go to the target website: *i.e* https://gritlab.ax/**



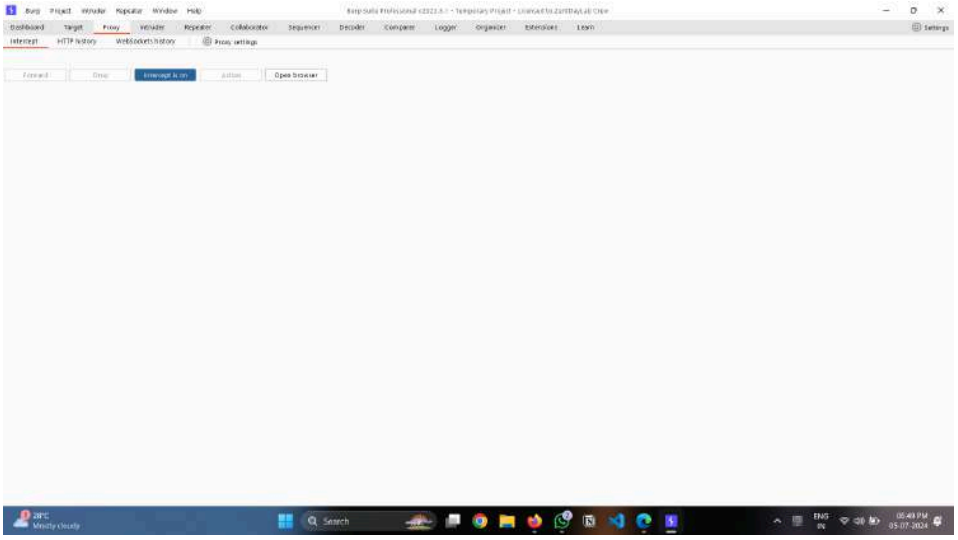
2. **Now open BurpSuite and turn on the Intercept and Refresh the Page.**



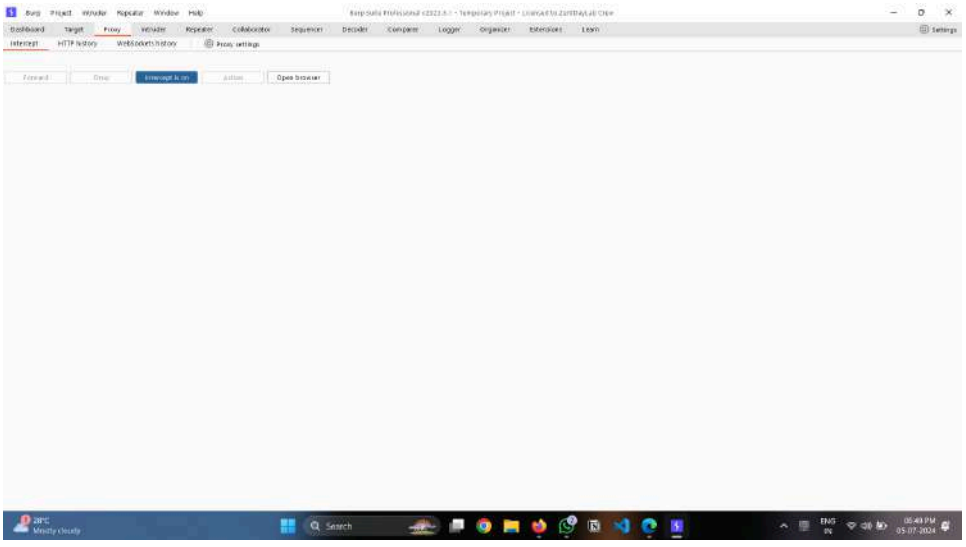
3. **Check for the value at Host header and change it to www.facebook.com.**



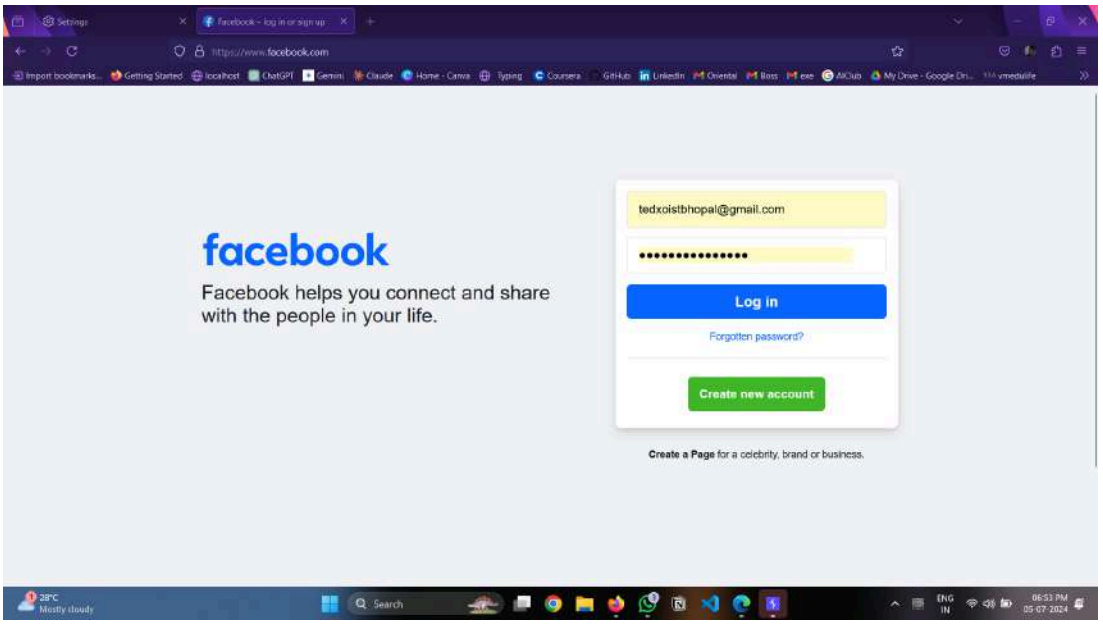
4. **Now Forward the packets.**



5. **Turn off the intercept and jump back to the browser and check if the website redirects to facebook page.**



6. **This website turns out to be Vulnerable to Host Header Injection.**



7. **Conclusion** – In the above steps we saw how we can check for Host Header Injection Vulnerability by modifying the Host header of the request. Here this site - gritlab.ax turns out to be vulnerable with this vulnerability.

B. Find 2 websites that are vulnerable to Open Redirect / URL Redirection Vulnerability.

Open Redirect / URL Redirection Vulnerability

- **CVSS Score** - 6.1 (Medium)
- **Related OWASP Top 10** - A10:2021 - Server-Side Request Forgery (SSRF)
- **Explanation**: An Open Redirect vulnerability occurs when an application takes a user-supplied URL as input and redirects to that URL without proper validation. This allows attackers to craft malicious URLs that appear to be from a trusted domain but actually redirect users to phishing sites or other malicious destinations.
- **Impact**: Phishing attacks, Credential theft, Malware distribution, Bypass of security controls, Reputation damage.
- **Recommendations**:

- ➢ Implement strict input validation: Use allowlists to validate and sanitize user-supplied URLs or parameters used in redirects.
- ➢ Avoid using user-supplied input for redirects
- ➢ Use indirect references: Instead of passing the full URL, use an ID or token that maps to the intended URL on the server-side.
- ➢ Use absolute URLs with hardcoded domains for redirects whenever possible.
- ➢ Implement warning pages

❖ **References**:

OWASP Unvalidated Redirects and Forwards Cheat Sheet:

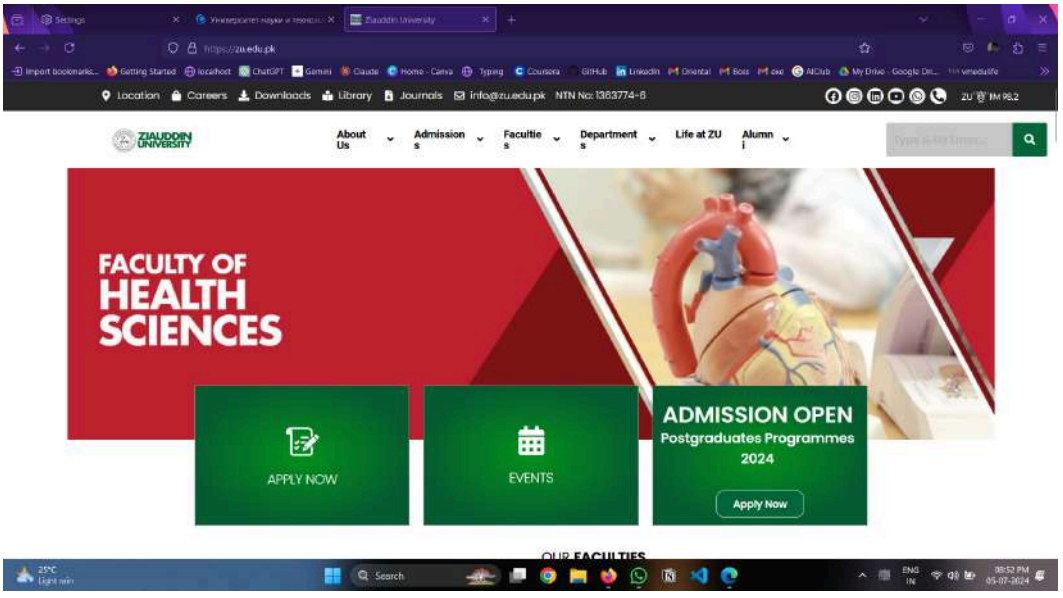
CWE-601: URL Redirection to Untrusted Site ('Open Redirect')

PortSwigger Web Security Academy: Open Redirection:

❖ **Procedure:**

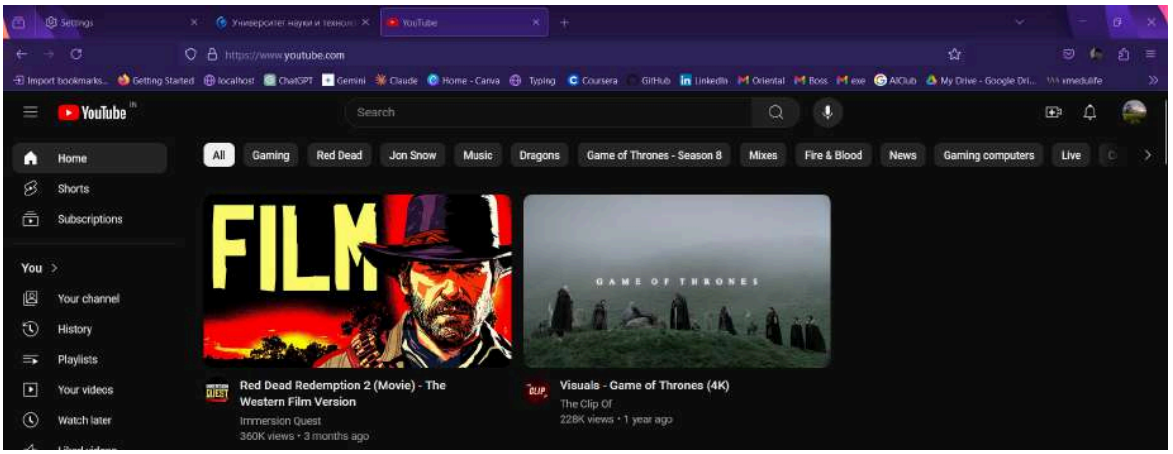
- ➢ **Target Website - Ziauddin University (zu.edu.pk) , https://misis.ru/**
- ➢ **Payload - "//youtube.com"**
- ➢ **Steps -**

1. **Open the Target Website- zu.edu.pk.**

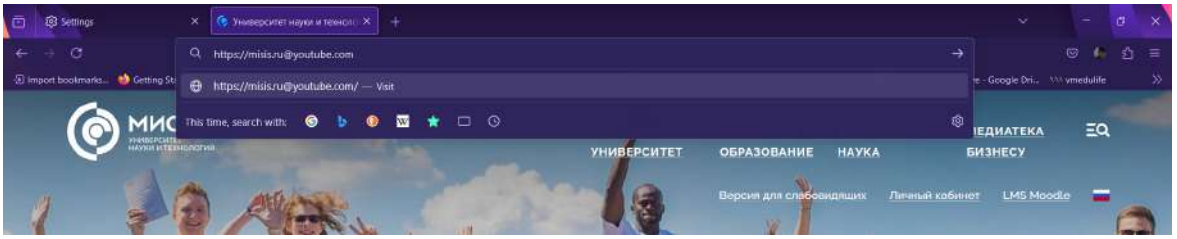


2. **Go to the URL Section and add "//youtube.com" at the end of the url.**

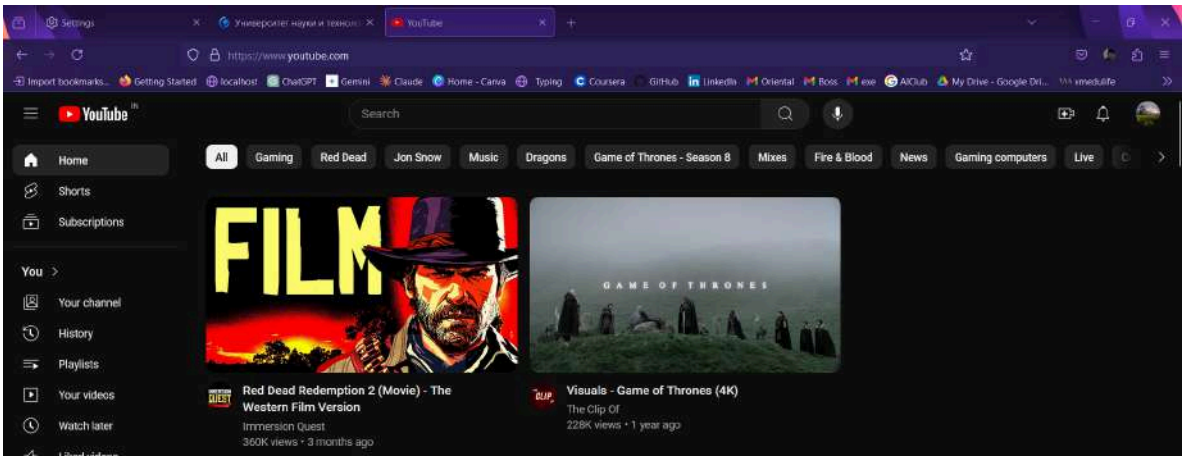
3. **Upon clicking Enter, the website starts redirecting to the YouTube site.**



4. **Next, go to the next url and add “@youtube.com” at the end.**



5. **This website also shows Open URL Vulnerability as it also redirects to the YouTube page upon clicking Enter.**



C. Find 2 websites that are vulnerable to iFrame Injection Vulnerability.

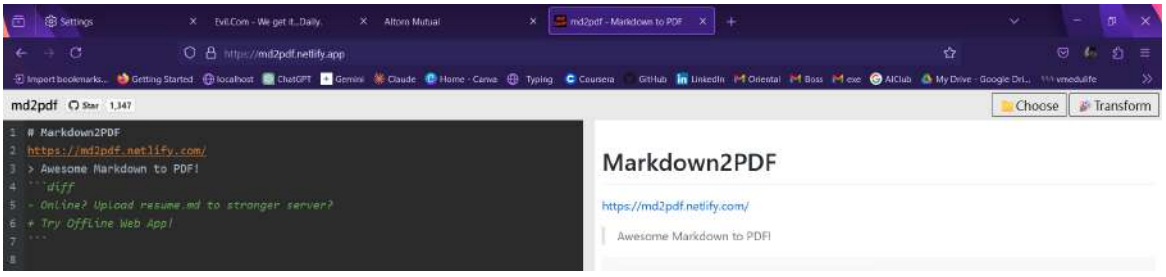
iFrame Injection Vulnerability

- **CVSS Score** - 6.5 (Medium)
- **Related OWASP Top 10** - A03:2021 - Injection
- **Explanation**: iFrame injection is a type of web vulnerability where an attacker can insert malicious iFrame HTML tags into a vulnerable web page. An iFrame (inline frame) is an HTML element that allows embedding one web page within another.
- **Impact**: Cross-Site Scripting (XSS), Phishing, Malware Distribution, Clickjacking, Data Theft, Loss of Reputation etc.
- **Recommendations**:
  - Input Validation and Sanitization
  - Content Security Policy
  - X-Frame-Options Header
  - Implement Frame Breaking Scripts
- **References**: OWASP iFrame Injection, CWE-80: Improper Neutralization of Script-Related HTML Tags in a Web Page (Basic XSS)
- **Procedure:**
  - **Target Websites -** https://md2pdf.netlify.app/, altoro.testfire.net
  - **Payload -**

```
<iframe width="560" height="315" src="https://www.evil.com/"
frameborder="0" allowfullscreen></iframe>
```

  - **Steps -**

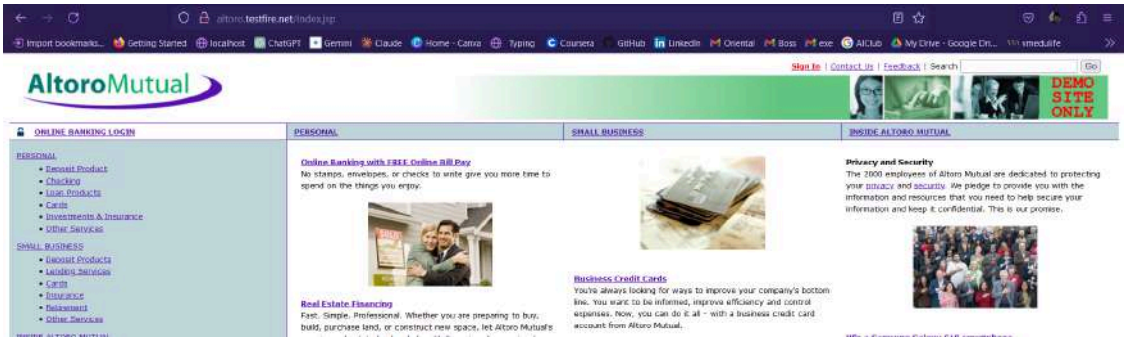
1. **Open the first target website.**



2. **Now on the text input field input the iFrame payload.**

3. **This site renders the iFrame of evil.com as seen here.**

4. **Now go to the next site altoro.testfire.net.**



5. **In the search field, enter the payload and hit Go.**

6. **A rendered iFrame Box is seen inside this website.**

❖ **Conclusion-**

iFrame is a critical vulnerability and should be avoided and mitigated for a secured web application. It can be tested by entering iframe HTML code in the input fields like search bars, comment boxes, Login Forms etc. of the websites.

In this case we tested md2pdf and altoro testfire sites for iFrame Vulnerability.

Supraja Technologies

23E05-ST#IS#6653-TASK12

A. Find cross-site scripting (XSS) Vulnerability Using the Reflected XSS test case in the website: Smtmax.com

Reflected XSS Vulnerability

- **CVSS Score** - 6.1 (Medium)
- **Related OWASP Top 10** - A03:2021 - Injection
- **Explanation**: Reflected XSS is a type of Cross-Site Scripting vulnerability where malicious scripts are injected into web applications and then immediately reflected back to the user and executed in their browser. This occurs when user input is not properly sanitized or encoded before being included in the output. The attack is typically delivered via a crafted URL, which, when clicked by the victim, causes the malicious script to execute in their browser context.
- **Impact**:

1. Session Hijacking
2. Data theft
3. Phishing
4. Malware distribution
5. Defacement: The appearance and content of the web page can be altered.
6. Keylogging: User keystrokes can be captured and sent to the attacker.
7. Reputation damage

- **Recommendations**:

1. Input validation
2. Output encoding
3. Content Security Policy (CSP): Implement a robust CSP
4. Use security headers: Headers like X-XSS-Protection and X-Content-Type-Options.
5. Sanitize input: Remove or neutralize potentially dangerous characters and scripts.
6. Use framework protection: Utilize built-in XSS protections provided by modern web frameworks.
7. Use HttpOnly flag: Set the HttpOnly flag on session cookies to prevent access via client-side scripts.

❖ **References**:

1. [OWASP Cross Site Scripting (XSS)]
2. [PortSwigger Web Security Academy: Cross-site scripting]:

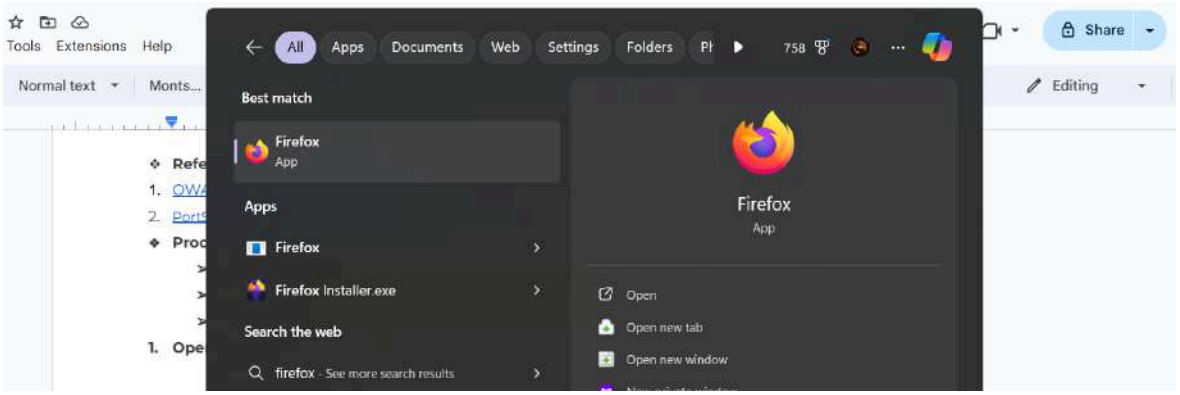
❖ **Procedure:**

➢ **Target Website -** smtmax.com

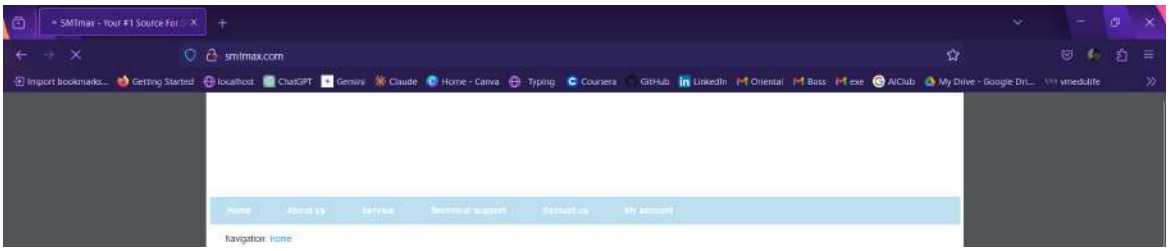
➢ **Payload -** `<script>alert("XSS Attack")</script>`

➢ **Steps -**

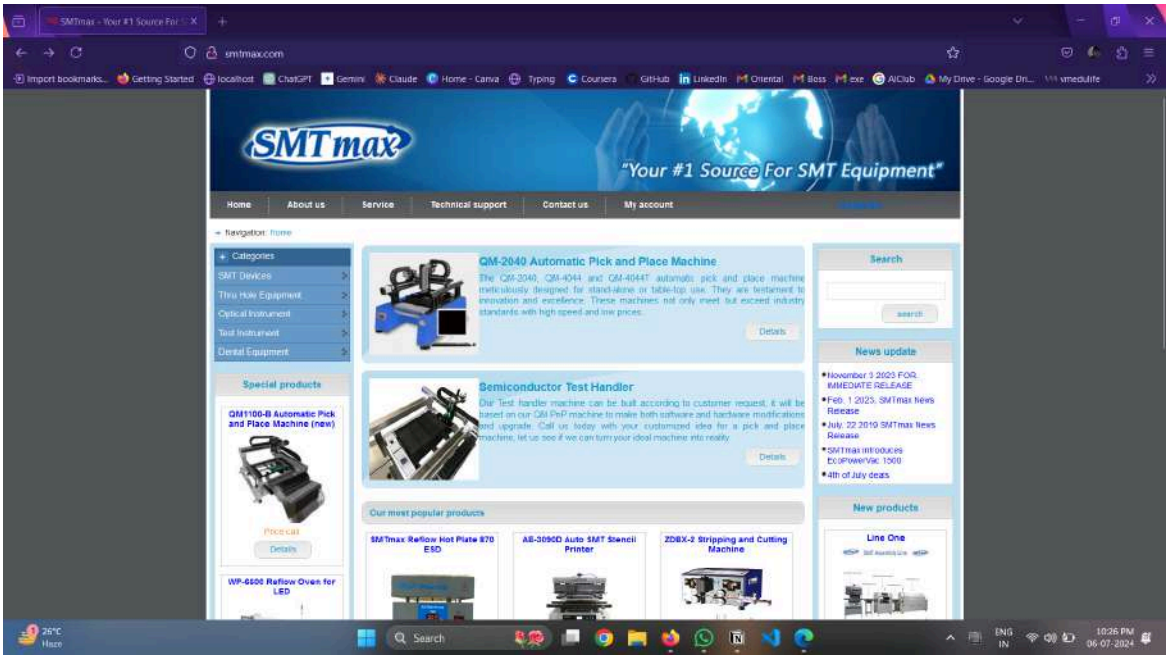
1. **Open any Browser.**



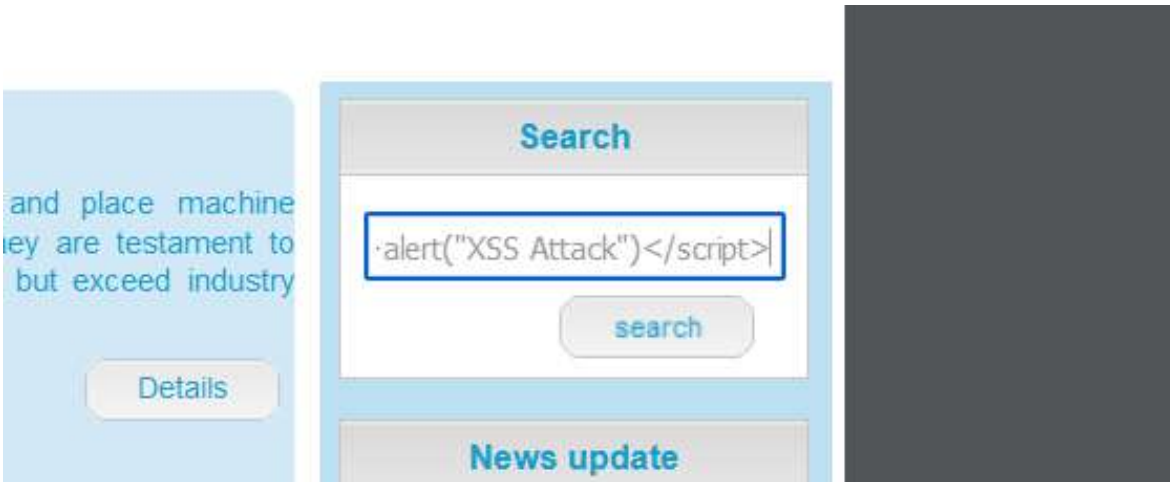
2. **Open the target website - smtmax.com**



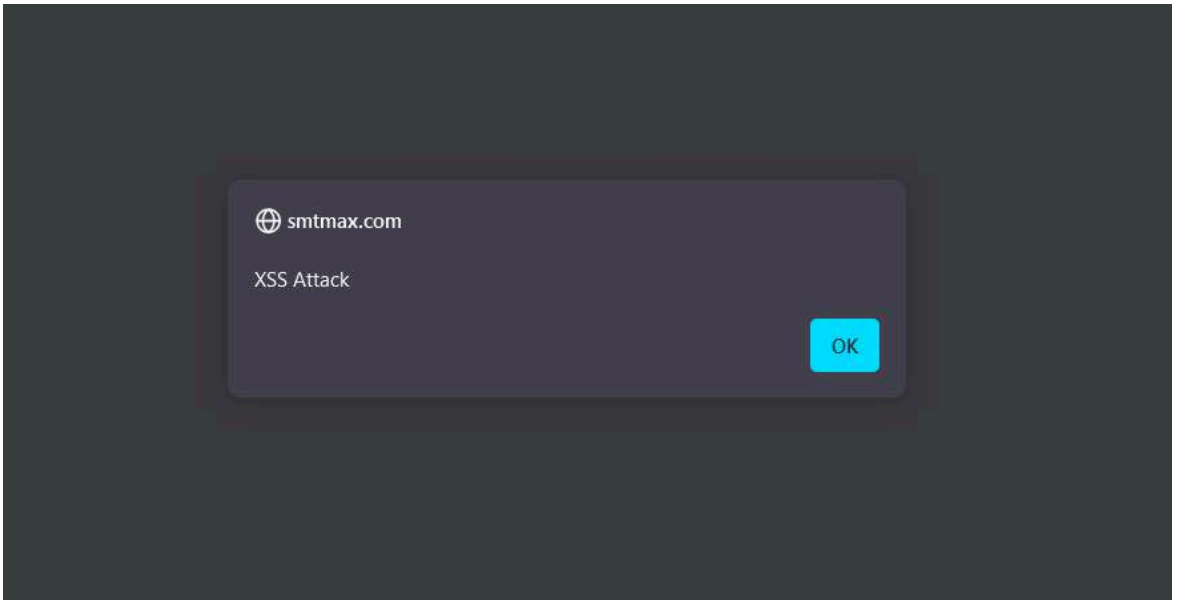
3. **Look out for any input fields on the website.**



4. **Now insert the payload of Reflected XSS on the search bar input field.**



5. **Upon clicking search, a pop up message appears. This website turns out to be vulnerable to a Reflected XSS attack.**



- **Conclusion** – In the above steps we saw how we can check for Reflected XSS Vulnerability by injecting html payload on the input field. Here this site - smtmax.com turns out to be vulnerable with this vulnerability.

B. Find a website that is vulnerable to IDOR (Insecure Direct Object References) Vulnerability.

IDOR (Insecure Direct Object References) Vulnerability

- **CVSS Score** - 6.5 (Medium)
- **Related OWASP Top 10** - A01:2021 - Broken Access Control
- **Explanation:** Insecure Direct Object References (IDOR) is a type of access control vulnerability that occurs when an application uses user-supplied input to access objects directly. This happens when the application fails to verify that the user has the appropriate permissions to access the requested object. As a result, attackers can manipulate parameters to access or modify data they shouldn't have permission to interact with, potentially leading to unauthorized access to sensitive information or functionality.
- **Impact**:

1. Unauthorized data access
2. Data manipulation
3. Privilege escalation
4. Privacy violations
5. Business logic bypass
6. Reputation damage

- **Recommendations**:
  - Implement proper access controls: Use RBAC or ABAC systems.
  - Use indirect reference maps: Replace direct object references with temporary, random indirect references.
  - Server-side validation:
  - Principle of least privilege: Ensure users have the minimum level of access required for their roles.
  - Use session-based tokens:
  - Input validation
  - Avoid exposing internal object references
  - Implement logging and monitoring
- **References**:

  OWASP Insecure Direct Object Reference Prevention

  CWE-639: Authorization Bypass Through User-Controlled Key

❖ **Procedure:**

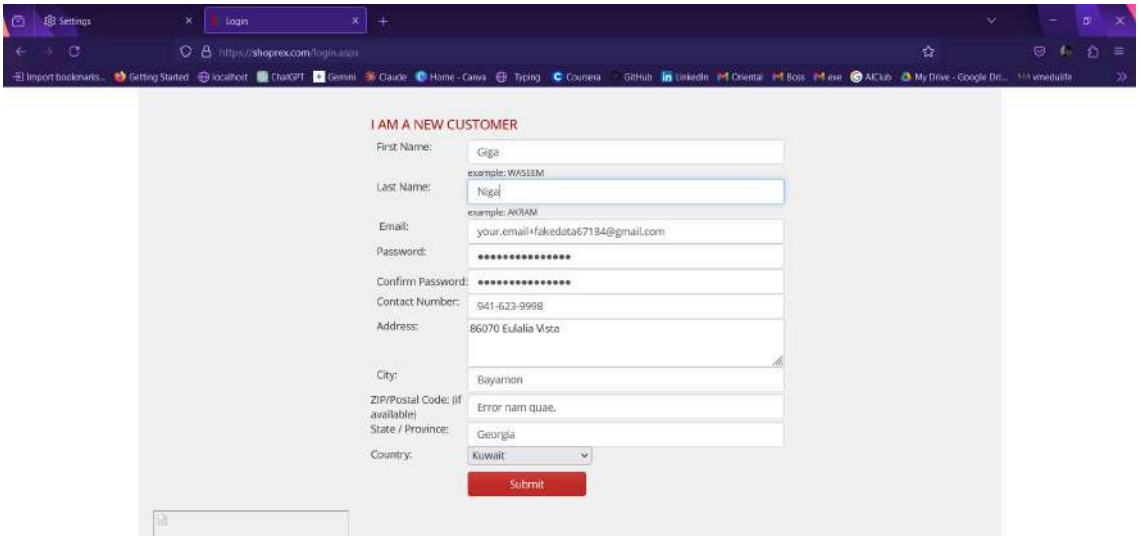
➢ Target Website - **https://www.shoprex.com**

➢ **Payload -** Modifying the User_ID

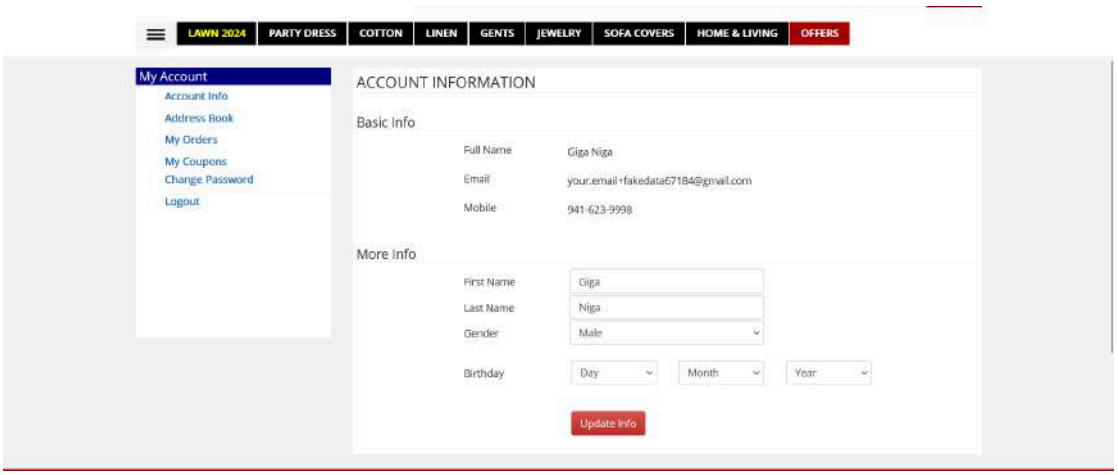
➢ **Steps -**

1. **Open the Target Website.**

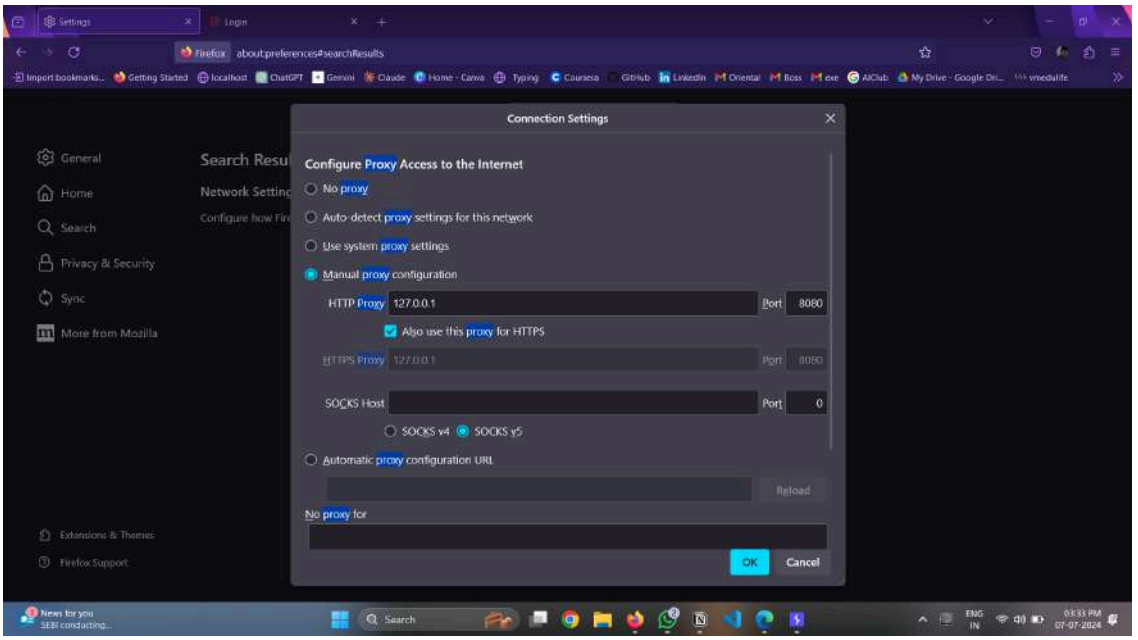
2. **Log on to the site by registering with fake email and password credentials.**



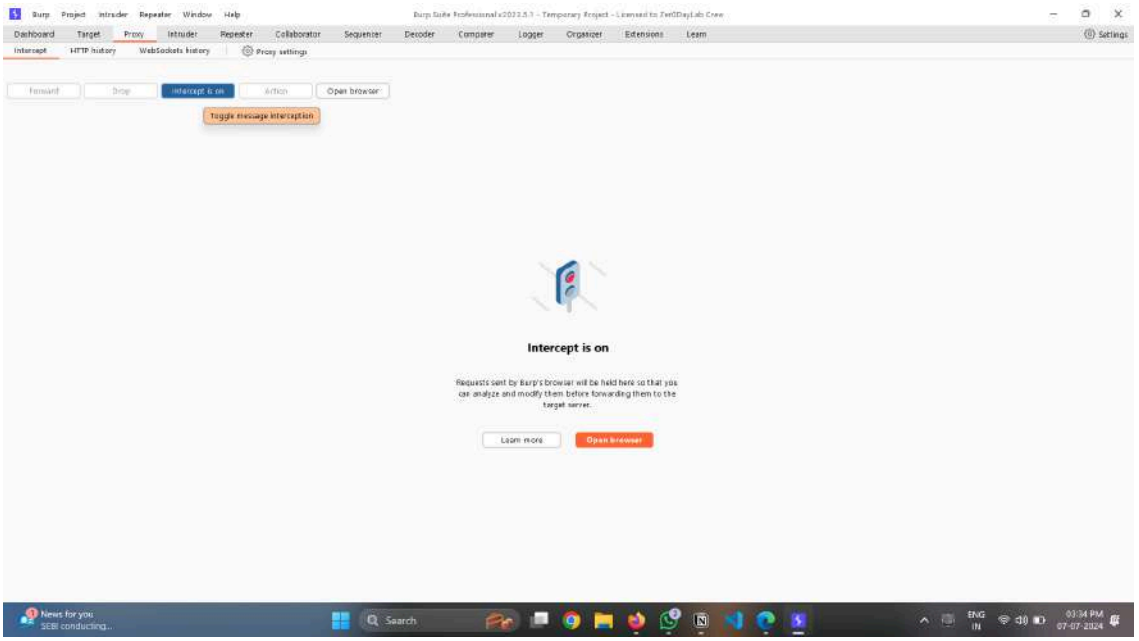
3. **The Profile section looks like this currently.**



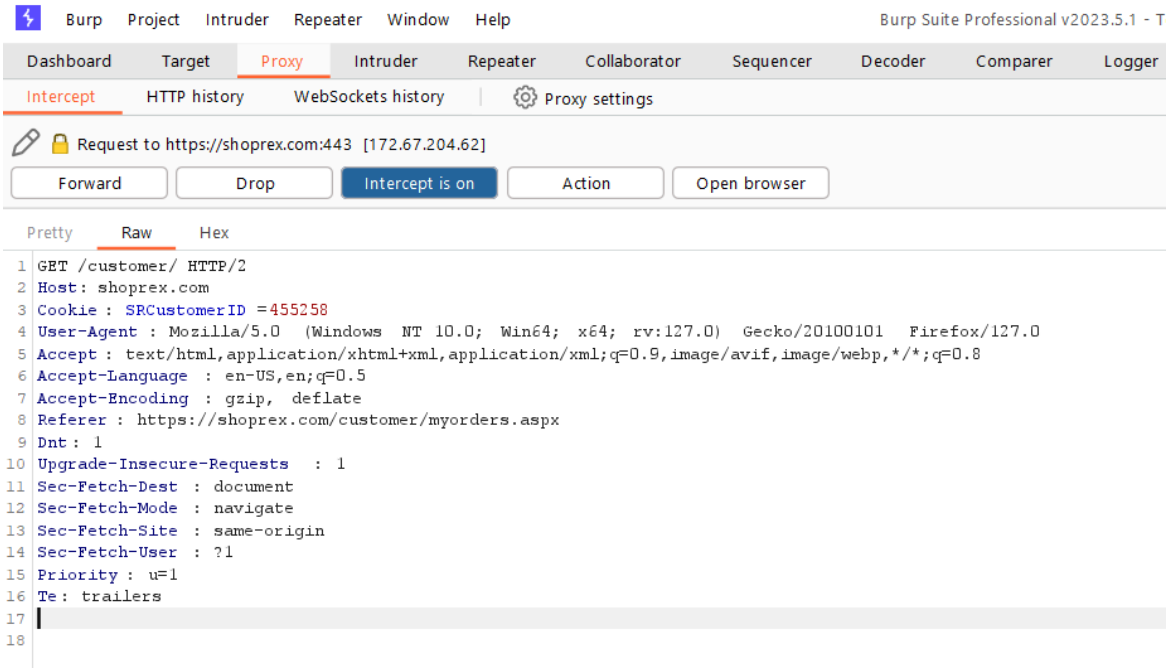
4. **Now open BurpSuite, and make sure the Proxy is configured.**



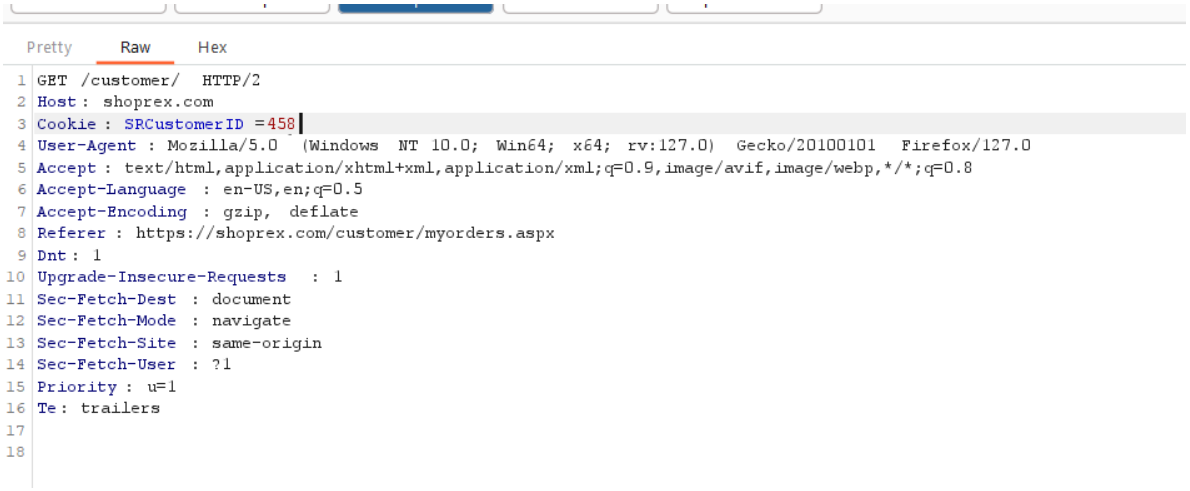
5. **Turn on the intercept and refresh the page.**



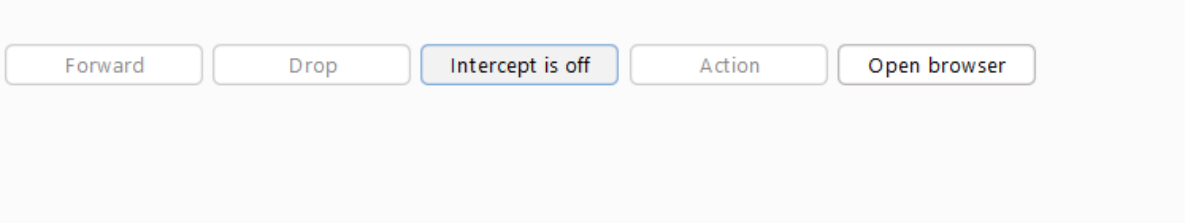
6. **The Proxy tab captures the requests involved on the website.**



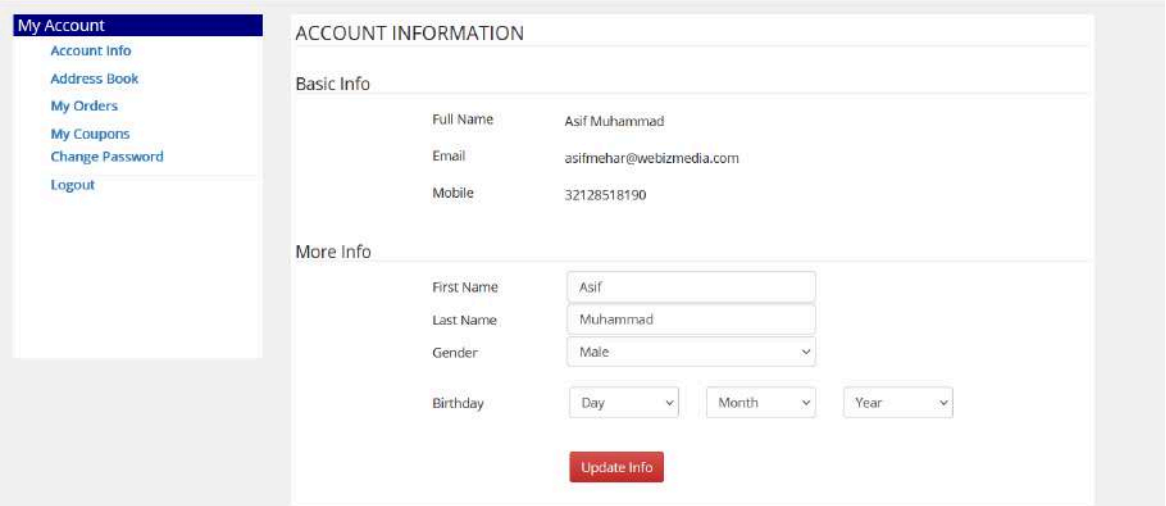
7. **Look out for *SRCustomerID*, here the information of the currently logged in user is shown. Modify it (Number smaller than the current number).**



8. **Now turn off the intercept, and jump back to your browser.**



9. **The information is changed and this site is vulnerable with IDOR.**



C. Find a website that is vulnerable to Broken Access Control Vulnerability.

Broken Access Control Vulnerability

- **CVSS Score** - 8.1 (High)
- **Related OWASP Top 10** - A01:2021 - Broken Access Control
- **Explanation**: Broken Access Control is a security vulnerability that occurs when an application fails to properly enforce restrictions on what authenticated users are allowed to do. This vulnerability allows attackers to access unauthorized functionality or data, potentially leading to information disclosure, modification, or destruction of all data, or performing unauthorized business functions.
- **Impact:**
    1. Unauthorized data access
    2. Data manipulation
    3. Business logic bypass:
    4. Vertical privilege escalation: Regular users accessing admin-only functionality.
    5. Horizontal privilege escalation: Users accessing other users' data or functions.
- **Recommendations**:
    - Implement proper access control mechanisms
    - Principle of least privilege

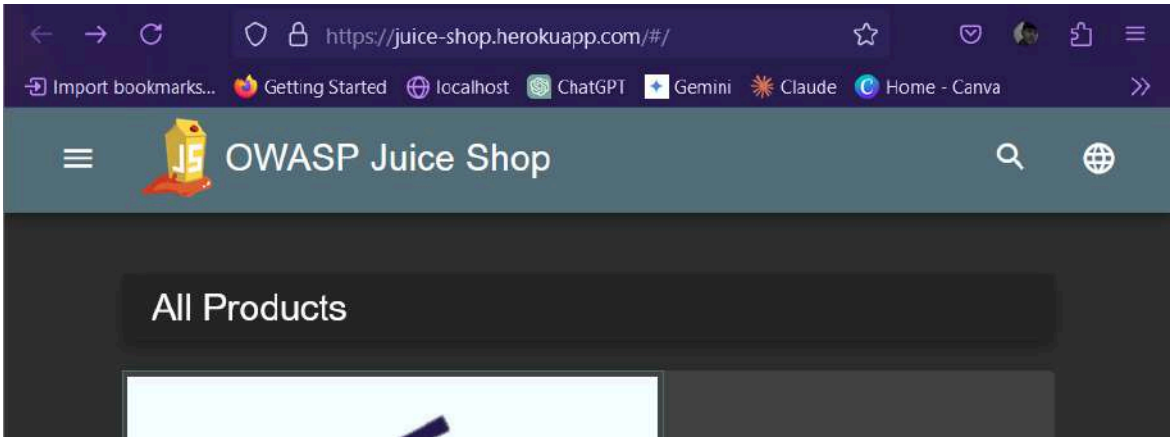
- ➢ Server-side validation
- ➢ Use session-based authentication
- ➢ Secure API endpoints

❖ **References**: OWASP Top 10 2021 - A01:2021 Broken Access Control

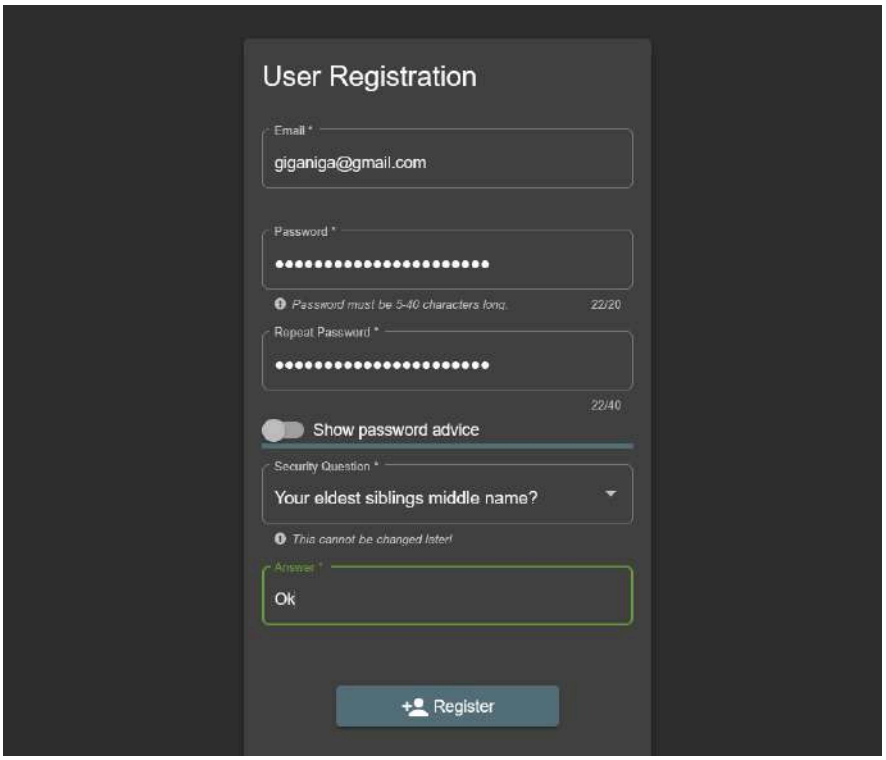
❖ **Procedure:**

- ➢ **Target Websites -** https://juice-shop.herokuapp.com
- ➢ **Steps -**

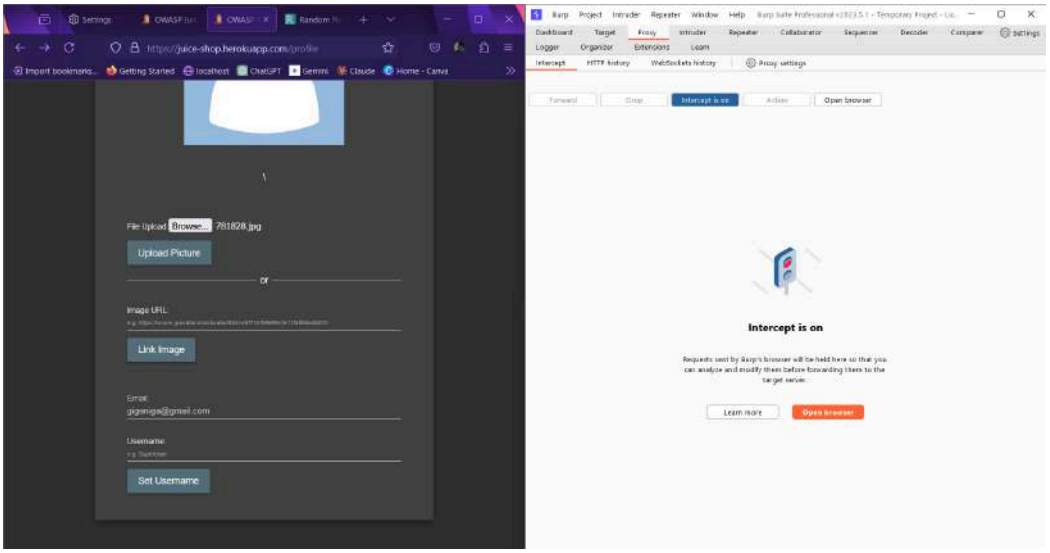
1. **Open the target website.**



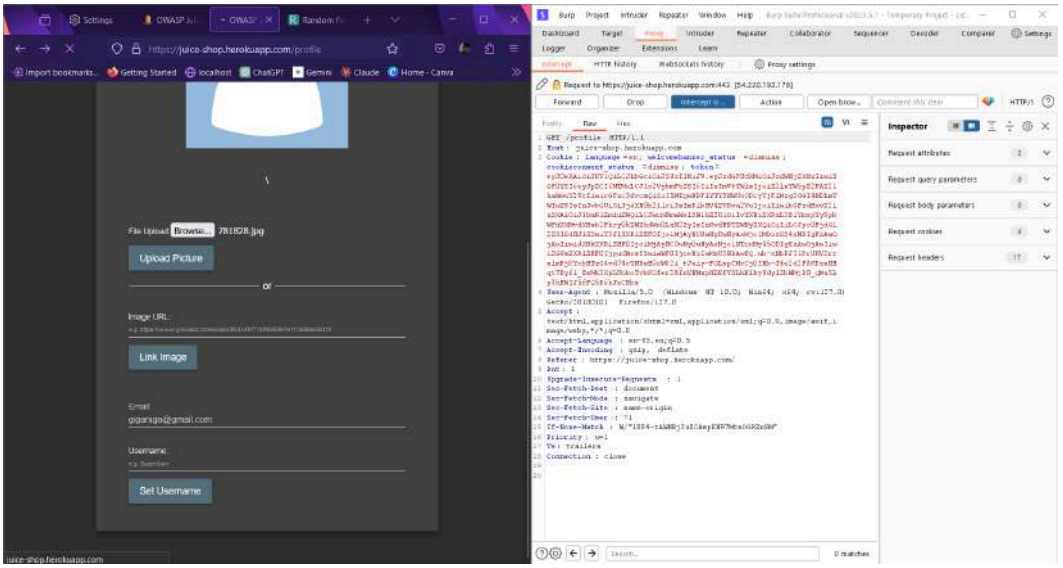
2. **Create two User Accounts on the website.**



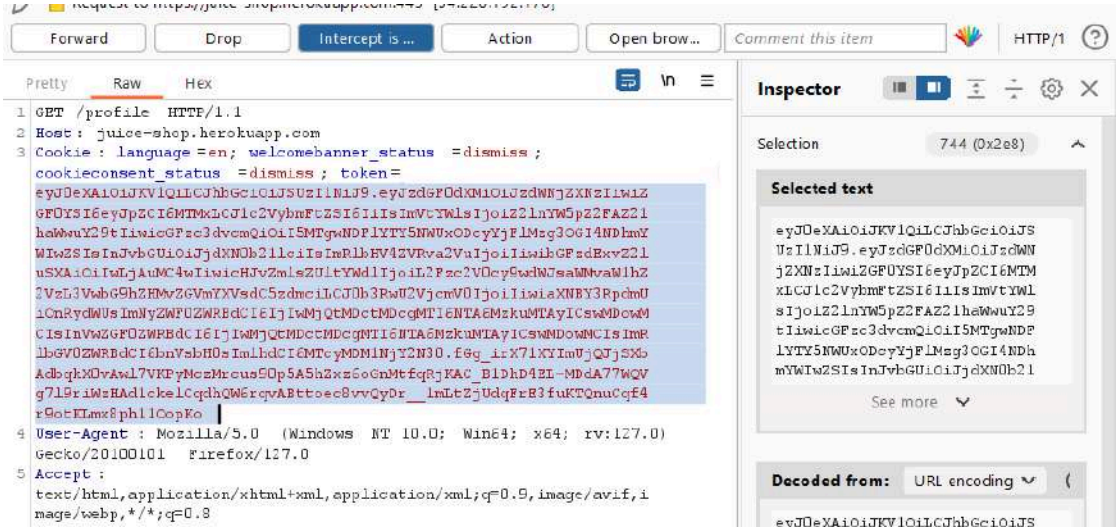
3. **Go to Profile Page of User Account 1 and turn on BurpSuite Intercept.**



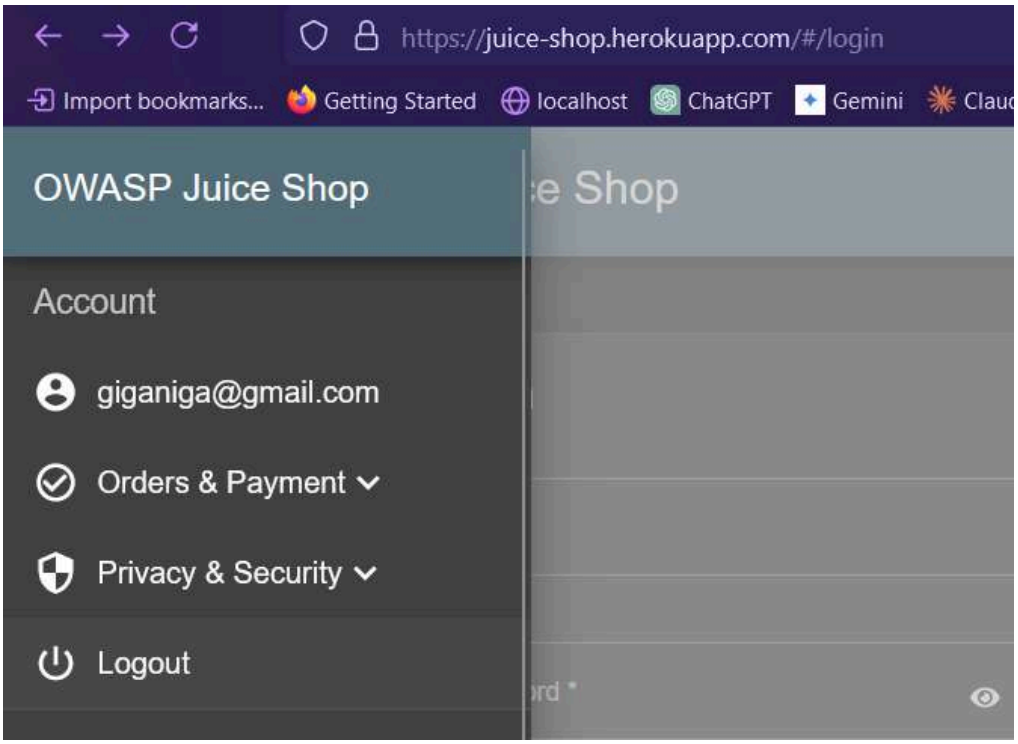
4. **Refresh the page and Capture the request.**



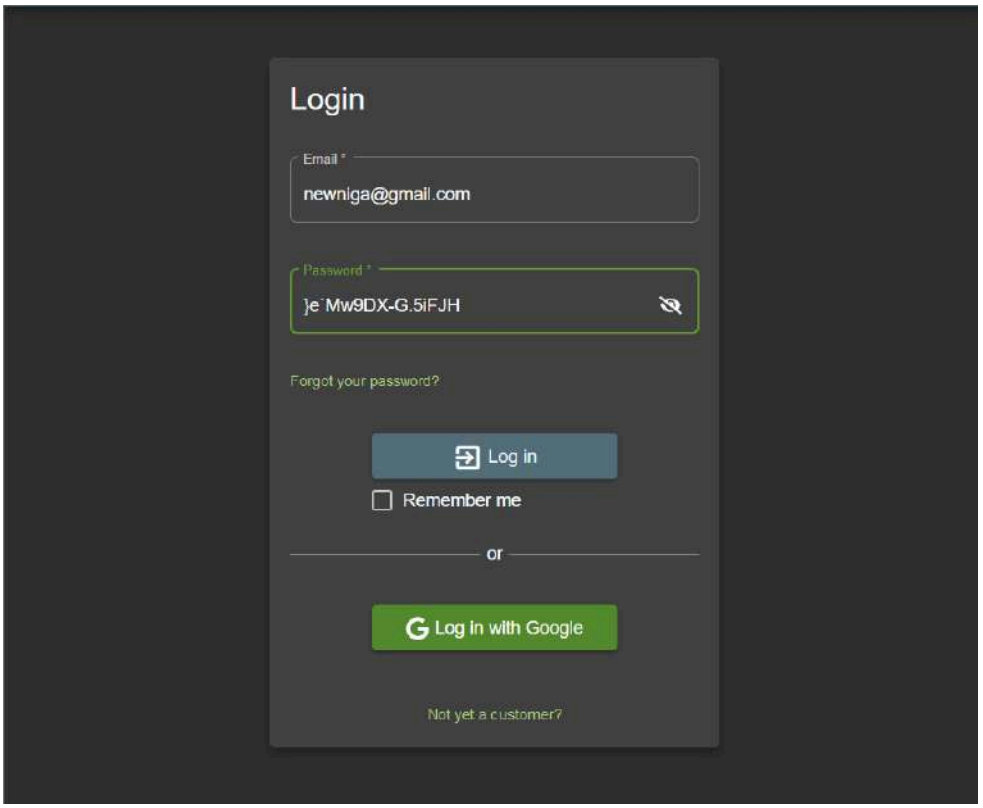
5. **Check for Tokens in the request and Copy the user 1 token value.**



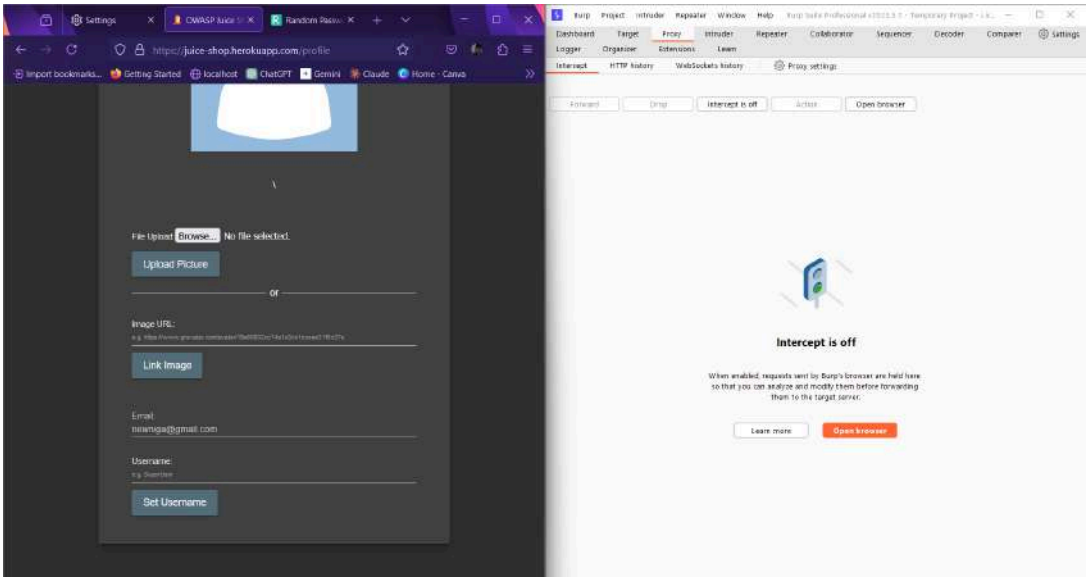
6. **Logout from the user 1 account.**



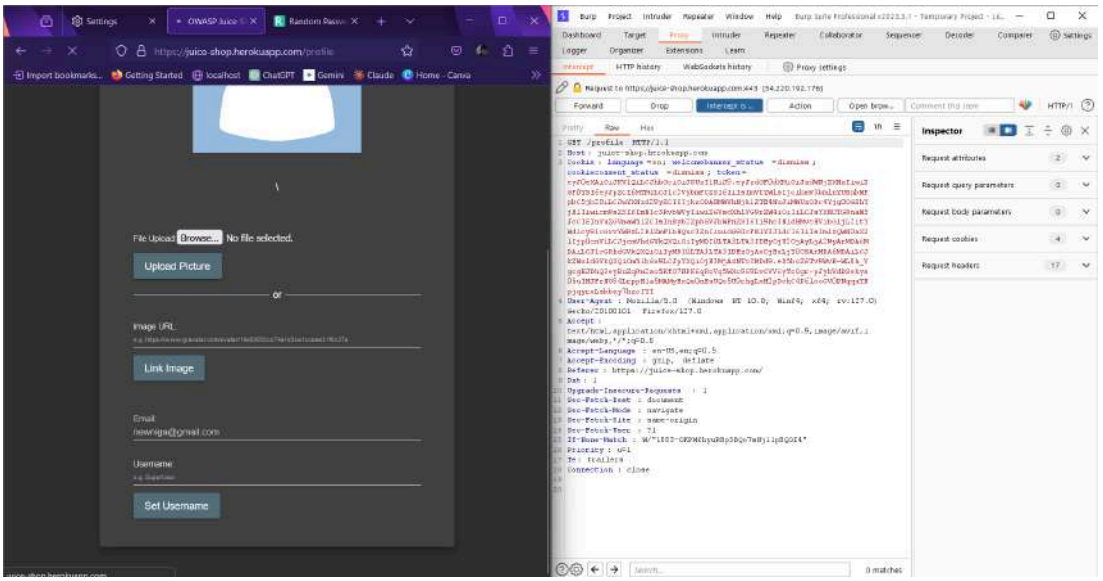
7. **Login into user 2.**



8. **Go to the profile page.**

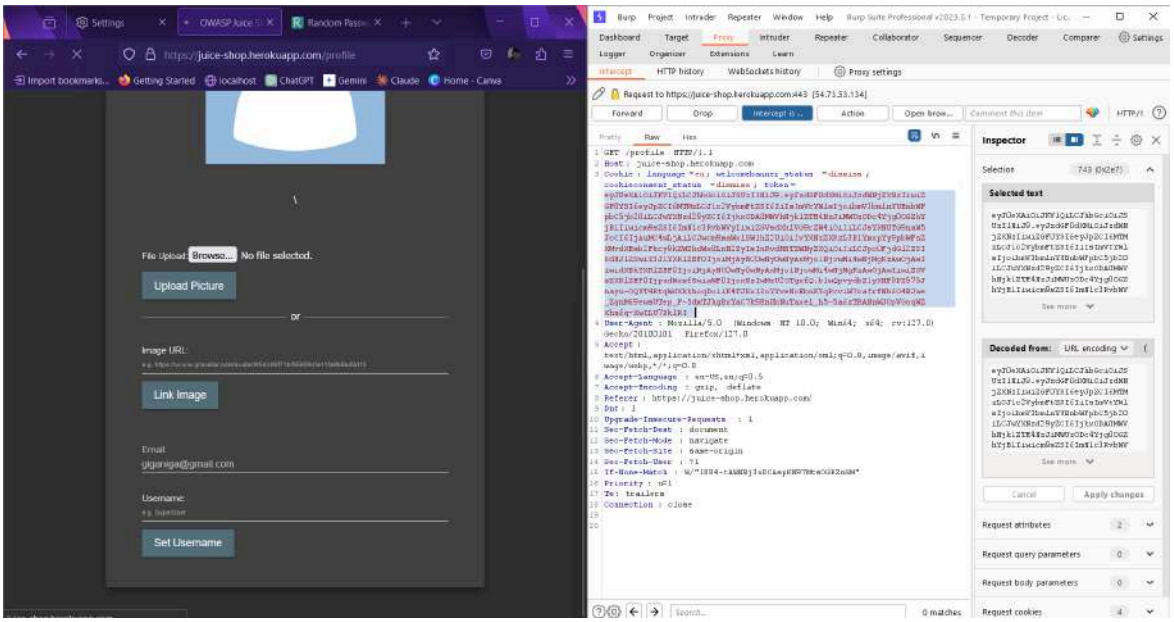


9. **Now Turn on Intercept in the burp suite tool. Refresh the user page & Capture the request.**



10. **Check for Tokens in the request then replace the user 2 token with the user 1 token value.**

11. **Forward the Request and check the User Details in the webpage.**



❖ **Conclusion-**

We got the information of user 1 by changing the token values in the header. Broken Access Vulnerability is found on this website. This vulnerability allows attackers to access unauthorized functionality or data, potentially leading to information disclosure, modification, or destruction of all data, or performing unauthorized business functions.