IPTables Tutorial

IP Tables works by using the packet filtering hooks in the Linux Kernel's Networking Stack. These kernel hooks are known as the *netfilter* framework. The packet filtering mechanism is divided into tables, chains, and targets.

Chains: There are 5 *netfilter* hooks.

- PREROUTING: Triggered by the NF_IP_PRE_ROUTING hook. Rules apply to incoming traffic coming from outside the system passing to the kernel networking stack. This is before a routing decision has been made.
- INPUT: Triggered by the NF_IP_LOCAL_IN hook. Rules apply to traffic destined to a local process after routing.
- FORWARD: Triggered by the NF_IP_FORWARD hook. Rules apply to incoming traffic that has been routed and if the packet is to be forwarded to another host
- OUTPUT: Triggered by the NF_IP_LOCAL_OUT hook. Rules apply to traffic initiated by a local process once it enters the networking stack.
- POSTROUTING: Triggered by the NF_IP_POST_ROUTING hook. Rules apply to traffic after it has been routed and before it is sent on the wire.

Tables: They allow you to do particular things with packets. There are 4 types of tables.
1) Filter: whether a packet should be allowed or dropped. (Applicable to: Input, Output, Forward Chains)
2) Mangle table: If a packet header needs to be altered. (Applicable to Prerouting, Postrouting, Forward, Input and Output Chains)
3) NAT table: To route packets to different hosts using NAT by changing the source or destination IP. (Applicable to Prerouting, Postrouting, Input and Output Chains)
4) Raw Table: Allows you to work with packets before the kernel starts tracking its state. (Applicable to Prerouting and Output Chains)

Targets:
To decide the action to be taken.

- Terminating
  - Accept - Accept and process the packet
  - Drop - It would appear as if the system doesn't exist
  - Reject - A connection reset is sent

IP Tables is stateful

Demo:

1) To see your IPTables rules

iptables -L -v --line-numbers [-L -> Listing the rules -v -> verbose output]

Sample Output

```
nirav@CIA:~$ sudo iptables -L -v --line-numbers
Chain INPUT (policy ACCEPT 2355K packets, 6293M bytes)
num   pkts bytes target     prot opt in     out     source
destination
1       18  1169 ACCEPT     udp  --  virbr0 any     anywhere
anywhere             udp dpt:domain
2        0     0 ACCEPT     tcp  --  virbr0 any     anywhere
anywhere             tcp dpt:domain
3       52 17056 ACCEPT     udp  --  virbr0 any     anywhere
anywhere             udp dpt:bootps
4        0     0 ACCEPT     tcp  --  virbr0 any     anywhere
anywhere             tcp dpt:bootps

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
num   pkts bytes target     prot opt in     out     source
destination
1    51199   77M ACCEPT     all  --  any    virbr0  anywhere
192.168.122.0/24     ctstate RELATED,ESTABLISHED
2     8716  489K ACCEPT     all  --  virbr0 any     192.168.122.0/24
anywhere
3        0     0 ACCEPT     all  --  virbr0 virbr0  anywhere
anywhere
4        0     0 REJECT     all  --  any    virbr0  anywhere
anywhere             reject-with icmp-port-unreachable
5        0     0 REJECT     all  --  virbr0 any     anywhere
anywhere             reject-with icmp-port-unreachable

Chain OUTPUT (policy ACCEPT 1557K packets, 3910M bytes)
num   pkts bytes target     prot opt in     out     source
destination
1       52 17056 ACCEPT     udp  --  any    virbr0  anywhere
anywhere             udp dpt:bootpc
```

2) Before we start the demo let us save the rules in a file

nirav@CIA:~$ sudo iptables-save > ~/iptables-rules

3) Let us flush the rules

iptables -F

Sample Output:

```
nirav@CIA:~$ sudo iptables -L -v --line-numbers
Chain INPUT (policy ACCEPT 232 packets, 4017K bytes)
num   pkts bytes target     prot opt in     out     source
destination
```

```
Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
num   pkts bytes target     prot opt in     out     source
destination

Chain OUTPUT (policy ACCEPT 240 packets, 4018K bytes)
num   pkts bytes target     prot opt in     out     source
destination
```

4) To set default policy

iptables -P INPUT ACCEPT(DROP/REJECT)
iptables -P OUTPUT ACCEPT(DROP/REJECT)
iptables -P FORWARD ACCEPT(DROP/REJECT)

Sample output:

```
nirav@CIA:~$ sudo iptables -P INPUT DROP
nirav@CIA:~$ sudo iptables -P OUTPUT DROP
nirav@CIA:~$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

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

Let us try pinging

```
nirav@CIA:~$ sudo iptables -P INPUT DROP
nirav@CIA:~$ sudo iptables -P OUTPUT DROP
nirav@CIA:~$ ping www.google.com
ping: unknown host www.google.com
nirav@CIA:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2032ms
```

As expected, we dont get any output.

Now let us allow packets which are going out of our device.

```
nirav@CIA:~$ sudo iptables -P OUTPUT ACCEPT
```

```
nirav@CIA:~$ sudo ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
^C
--- 8.8.8.8 ping statistics ---
9 packets transmitted, 0 received, 100% packet loss, time 8177ms
```

Since the ICMP replies are blocked by the input chain, we arent getting any reply

Now let us add a rule which makes all traffic originated by our machine to be accepted by the input chain.

```
nirav@CIA:~$ sudo iptables -A INPUT -m state --state NEW,ESTABLISHED -j ACCEPT
nirav@CIA:~$ sudo ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=45 time=35.7 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=45 time=37.8 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=45 time=32.6 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 32.694/35.441/37.893/2.138 ms
nirav@CIA:~$ sudo iptables -L -v --line-numbers
Chain INPUT (policy DROP 0 packets, 0 bytes)
num   pkts bytes target     prot opt in     out     source
destination
1      821  113K ACCEPT     all  --  any    any     anywhere
anywhere             state NEW,ESTABLISHED

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
num   pkts bytes target     prot opt in     out     source
destination

Chain OUTPUT (policy ACCEPT 711 packets, 75683 bytes)
num   pkts bytes target     prot opt in     out     source
destination
```

5) Adding Rules

```
nirav@CIA:~$ sudo iptables -A OUTPUT  -p tcp -j DROP
nirav@CIA:~$
nirav@CIA:~$
nirav@CIA:~$ sudo iptables -L --line-numbers
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
num  target     prot opt source               destination
```

```
Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination
1    DROP       tcp  --  anywhere             anywhere
nirav@CIA:~$ sudo iptables -I OUTPUT 1 -p icmp -j DROP
nirav@CIA:~$ sudo iptables -L --line-numbers
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
num  target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination
1    DROP       icmp --  anywhere             anywhere
2    DROP       tcp  --  anywhere             anywhere
nirav@CIA:~$
```

6) To delete a rule

iptables -D INPUT 4

7)To see NAT Table

iptables -t nat -L

Sample Output:

```
nirav@CIA:~$ sudo iptables -t nat -L -v --line-numbers
Chain PREROUTING (policy ACCEPT 378 packets, 37709 bytes)
num   pkts bytes target     prot opt in     out     source
destination

Chain INPUT (policy ACCEPT 137 packets, 18231 bytes)
num   pkts bytes target     prot opt in     out     source
destination

Chain OUTPUT (policy ACCEPT 17982 packets, 1166K bytes)
num   pkts bytes target     prot opt in     out     source
destination

Chain POSTROUTING (policy ACCEPT 17321 packets, 1106K bytes)
num   pkts bytes target     prot opt in     out     source
destination
1        7   873 RETURN     all  --  any    any     192.168.122.0/24
base-address.mcast.net/24
```

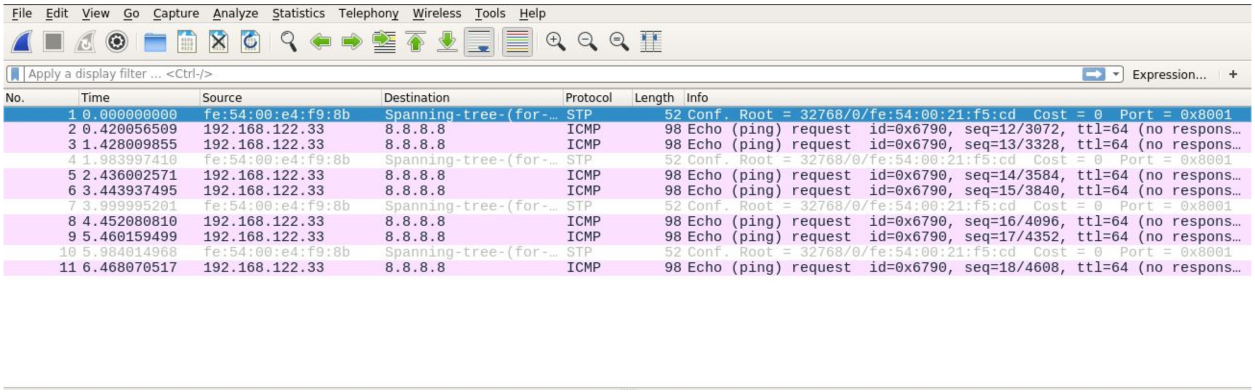
```
2        0     0 RETURN     all  --  any    any     192.168.122.0/24
255.255.255.255
3        0     0 MASQUERADE  tcp  --  any    any     192.168.122.0/24
!192.168.122.0/24     masq ports: 1024-65535
4      124 24800 MASQUERADE  udp  --  any    any     192.168.122.0/24
!192.168.122.0/24     masq ports: 1024-65535
5        2   168 MASQUERADE  all  --  any    any     192.168.122.0/24
!192.168.122.0/24
```

Let us add a NAT rule

Any packet coming from `192.168.122.0/24` is destination natted to 5.5.5.5 and source natted to 8.8.8.8

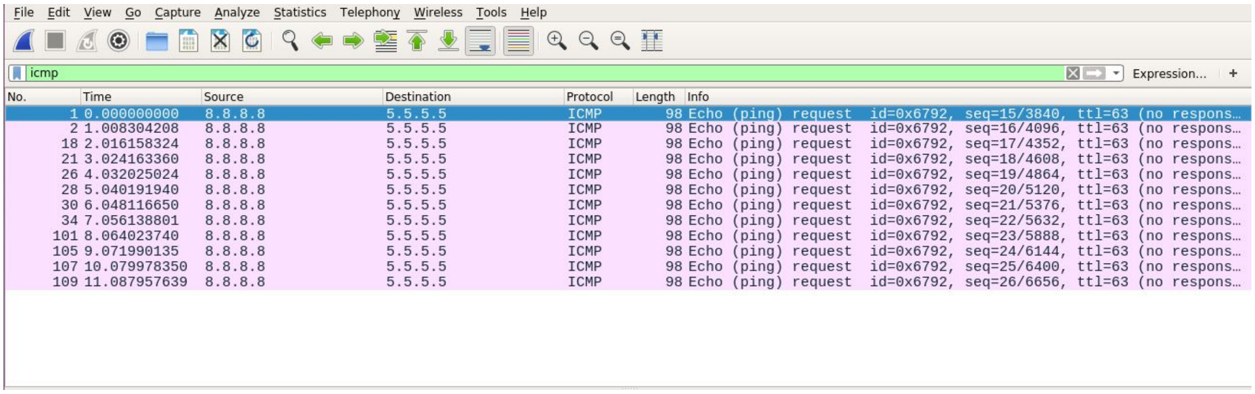
```
nirav@CIA:~$ sudo iptables -t nat -I PREROUTING 1 -s 192.168.122.0/24 -j DNAT
--to 5.5.5.5
nirav@CIA:~$ sudo iptables -t nat -I POSTROUTING 1 -s 192.168.122.0/24 -j SNAT
--to-source 8.8.8.8
nirav@CIA:~$ sudo iptables -t nat -L -v --line-numbers
Chain PREROUTING (policy ACCEPT 0 packets, 0 bytes)
num   pkts bytes target     prot opt in     out     source
destination
1       14  2580 DNAT       all  --  any    any     192.168.122.0/24
anywhere             to:5.5.5.5

Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
num   pkts bytes target     prot opt in     out     source
destination

Chain OUTPUT (policy ACCEPT 3 packets, 186 bytes)
num   pkts bytes target     prot opt in     out     source
destination

Chain POSTROUTING (policy ACCEPT 3 packets, 186 bytes)
num   pkts bytes target     prot opt in     out     source
destination
1       14  2580 SNAT       all  --  any    any     192.168.122.0/24
anywhere             to:8.8.8.8
2        8  1042 RETURN     all  --  any    any     192.168.122.0/24
base-address.mcast.net/24
3        0     0 RETURN     all  --  any    any     192.168.122.0/24
255.255.255.255
4        0     0 MASQUERADE  tcp  --  any    any     192.168.122.0/24
!192.168.122.0/24     masq ports: 1024-65535
5      134 26800 MASQUERADE  udp  --  any    any     192.168.122.0/24
!192.168.122.0/24     masq ports: 1024-65535
6        2   168 MASQUERADE  all  --  any    any     192.168.122.0/24
!192.168.122.0/24
```

At the output interface of the VM


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+

```
No.  Time        Source             Destination             Protocol Length Info
   1 0.000000000 fe:54:00:e4:f9:8b  Spanning-tree-(for-…  STP      52 Conf. Root = 32768/0/fe:54:00:21:f5:cd  Cost = 0  Port = 0x8001
   2 0.420056509 192.168.122.33     8.8.8.8                 ICMP     98 Echo (ping) request  id=0x6790, seq=12/3072, ttl=64 (no respons…
   3 1.428009855 192.168.122.33     8.8.8.8                 ICMP     98 Echo (ping) request  id=0x6790, seq=13/3328, ttl=64 (no respons…
   4 1.983997410 fe:54:00:e4:f9:8b  Spanning-tree-(for-…  STP      52 Conf. Root = 32768/0/fe:54:00:21:f5:cd  Cost = 0  Port = 0x8001
   5 2.436002571 192.168.122.33     8.8.8.8                 ICMP     98 Echo (ping) request  id=0x6790, seq=14/3584, ttl=64 (no respons…
   6 3.443937495 192.168.122.33     8.8.8.8                 ICMP     98 Echo (ping) request  id=0x6790, seq=15/3840, ttl=64 (no respons…
   7 3.999995201 fe:54:00:e4:f9:8b  Spanning-tree-(for-…  STP      52 Conf. Root = 32768/0/fe:54:00:21:f5:cd  Cost = 0  Port = 0x8001
   8 4.452080810 192.168.122.33     8.8.8.8                 ICMP     98 Echo (ping) request  id=0x6790, seq=16/4096, ttl=64 (no respons…
   9 5.460159499 192.168.122.33     8.8.8.8                 ICMP     98 Echo (ping) request  id=0x6790, seq=17/4352, ttl=64 (no respons…
  10 5.984014968 fe:54:00:e4:f9:8b  Spanning-tree-(for-…  STP      52 Conf. Root = 32768/0/fe:54:00:21:f5:cd  Cost = 0  Port = 0x8001
  11 6.468070517 192.168.122.33     8.8.8.8                 ICMP     98 Echo (ping) request  id=0x6790, seq=18/4608, ttl=64 (no respons…
```

At the WLAN interface


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...
+

icmp

```
No.  Time         Source   Destination  Protocol Length Info
   1 0.000000000  8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=15/3840, ttl=63 (no respons…
   2 1.008304208  8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=16/4096, ttl=63 (no respons…
  18 2.016158324  8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=17/4352, ttl=63 (no respons…
  21 3.024163360  8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=18/4608, ttl=63 (no respons…
  26 4.032025024  8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=19/4864, ttl=63 (no respons…
  28 5.040191940  8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=20/5120, ttl=63 (no respons…
  30 6.048116650  8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=21/5376, ttl=63 (no respons…
  34 7.056138801  8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=22/5632, ttl=63 (no respons…
 101 8.064023740  8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=23/5888, ttl=63 (no respons…
 105 9.071990135  8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=24/6144, ttl=63 (no respons…
 107 10.079978350 8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=25/6400, ttl=63 (no respons…
 109 11.087957639 8.8.8.8  5.5.5.5      ICMP     98 Echo (ping) request  id=0x6792, seq=26/6656, ttl=63 (no respons…
```

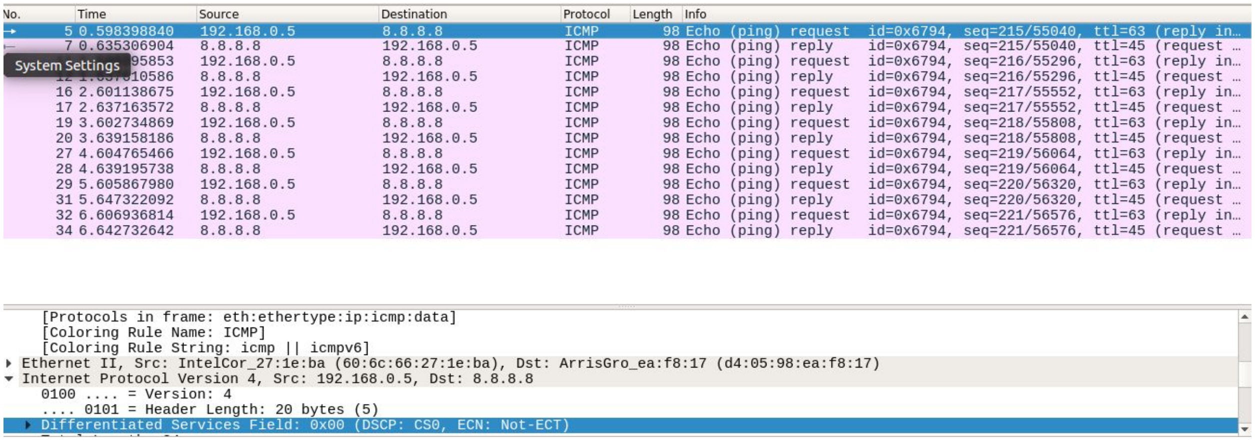
8)Mangle Table

You can change TOS and TTL.

```
nirav@CIA:~$ sudo iptables -t mangle -L -v --line-numbers
Chain PREROUTING (policy ACCEPT 488K packets, 7742M bytes)
num   pkts bytes target     prot opt in     out     source               destination

Chain INPUT (policy ACCEPT 484K packets, 7742M bytes)
num   pkts bytes target     prot opt in     out     source               destination

Chain FORWARD (policy ACCEPT 4004 packets, 357K bytes)
num   pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 396K packets, 7462M bytes)
num   pkts bytes target     prot opt in     out     source               destination

Chain POSTROUTING (policy ACCEPT 400K packets, 7462M bytes)
num   pkts bytes target     prot opt in     out     source               destination
1       22  7216 CHECKSUM   udp  --  any    virbr0  anywhere             anywhere             udp dpt:bootpc
CHECKSUM fill
```

Let's add a rule.

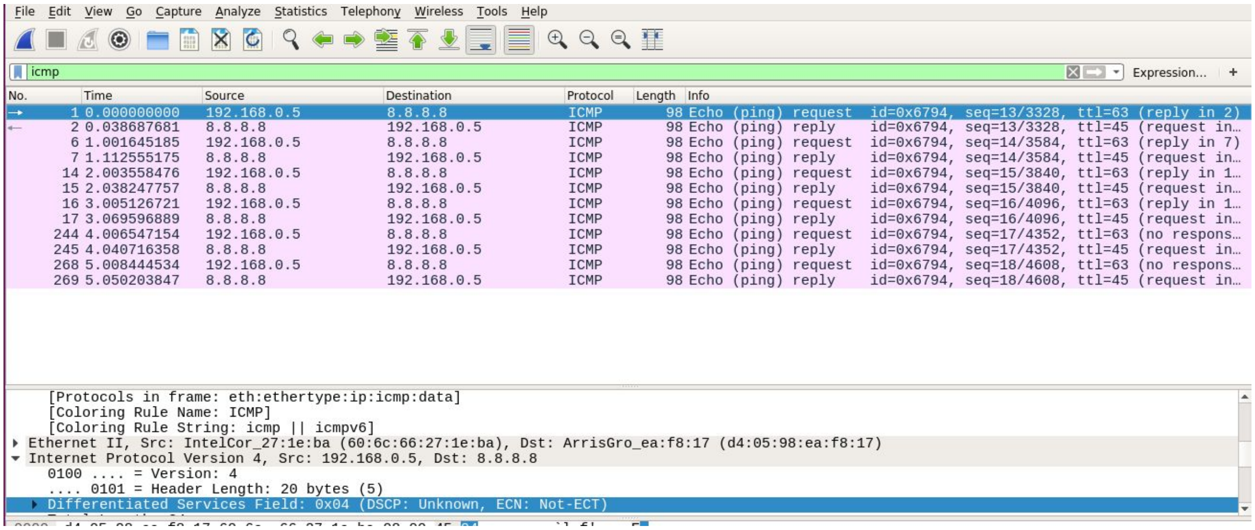
```
nirav@CIA:~$ sudo iptables -t mangle -A PREROUTING -s 192.168.122.0/24 -j TOS --set-tos 0x04
nirav@CIA:~$ sudo iptables -t mangle -L -v --line-numbers
Chain PREROUTING (policy ACCEPT 10713 packets, 247M bytes)
num   pkts bytes target     prot opt in     out     source               destination
1       63  6756 TOS        all  --  any    any     192.168.122.0/24     anywhere             TOS set 0x04/0xff

Chain INPUT (policy ACCEPT 10594 packets, 247M bytes)
num   pkts bytes target     prot opt in     out     source               destination

Chain FORWARD (policy ACCEPT 119 packets, 11460 bytes)
num   pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 8561 packets, 242M bytes)
num   pkts bytes target     prot opt in     out     source               destination

Chain POSTROUTING (policy ACCEPT 8680 packets, 242M bytes)
num   pkts bytes target     prot opt in     out     source               destination
1       22  7216 CHECKSUM   udp  --  any    virbr0  anywhere             anywhere             udp dpt:bootpc
CHECKSUM fill
```



```
No.  Time          Source        Destination   Protocol Length Info
   5 0.598398840   192.168.0.5   8.8.8.8       ICMP       98 Echo (ping) request  id=0x6794, seq=215/55040, ttl=63 (reply in…
   7 0.635306904   8.8.8.8       192.168.0.5   ICMP       98 Echo (ping) reply    id=0x6794, seq=215/55040, ttl=45 (request …
         95853     192.168.0.5   8.8.8.8       ICMP       98 Echo (ping) request  id=0x6794, seq=216/55296, ttl=63 (reply in…
         10586     8.8.8.8       192.168.0.5   ICMP       98 Echo (ping) reply    id=0x6794, seq=216/55296, ttl=45 (request …
  16 2.601138675   192.168.0.5   8.8.8.8       ICMP       98 Echo (ping) request  id=0x6794, seq=217/55552, ttl=63 (reply in…
  17 2.637163572   8.8.8.8       192.168.0.5   ICMP       98 Echo (ping) reply    id=0x6794, seq=217/55552, ttl=45 (request …
  19 3.602734869   192.168.0.5   8.8.8.8       ICMP       98 Echo (ping) request  id=0x6794, seq=218/55808, ttl=63 (reply in…
  20 3.639158186   8.8.8.8       192.168.0.5   ICMP       98 Echo (ping) reply    id=0x6794, seq=218/55808, ttl=45 (request …
  27 4.604765466   192.168.0.5   8.8.8.8       ICMP       98 Echo (ping) request  id=0x6794, seq=219/56064, ttl=63 (reply in…
  28 4.639195738   8.8.8.8       192.168.0.5   ICMP       98 Echo (ping) reply    id=0x6794, seq=219/56064, ttl=45 (request …
  29 5.605867980   192.168.0.5   8.8.8.8       ICMP       98 Echo (ping) request  id=0x6794, seq=220/56320, ttl=63 (reply in…
  31 5.647322092   8.8.8.8       192.168.0.5   ICMP       98 Echo (ping) reply    id=0x6794, seq=220/56320, ttl=45 (request …
  32 6.606936814   192.168.0.5   8.8.8.8       ICMP       98 Echo (ping) request  id=0x6794, seq=221/56576, ttl=63 (reply in…
  34 6.642732642   8.8.8.8       192.168.0.5   ICMP       98 Echo (ping) reply    id=0x6794, seq=221/56576, ttl=45 (request …
```

```
    [Protocols in frame: eth:ethertype:ip:icmp:data]
    [Coloring Rule Name: ICMP]
    [Coloring Rule String: icmp || icmpv6]
▸ Ethernet II, Src: IntelCor_27:1e:ba (60:6c:66:27:1e:ba), Dst: ArrisGro_ea:f8:17 (d4:05:98:ea:f8:17)
▾ Internet Protocol Version 4, Src: 192.168.0.5, Dst: 8.8.8.8
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▸ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
```

After the mangle operation is done



File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmp
Expression…
+

```
No.     Time           Source          Destination     Protocol  Length Info
    1 0.000000000    192.168.0.5     8.8.8.8          ICMP       98 Echo (ping) request  id=0x6794, seq=13/3328, ttl=63 (reply in 2)
    2 0.038687681    8.8.8.8         192.168.0.5      ICMP       98 Echo (ping) reply    id=0x6794, seq=13/3328, ttl=45 (request in…
    6 1.001645185    192.168.0.5     8.8.8.8          ICMP       98 Echo (ping) request  id=0x6794, seq=14/3584, ttl=63 (reply in 7)
    7 1.112555175    8.8.8.8         192.168.0.5      ICMP       98 Echo (ping) reply    id=0x6794, seq=14/3584, ttl=45 (request in…
   14 2.003558476    192.168.0.5     8.8.8.8          ICMP       98 Echo (ping) request  id=0x6794, seq=15/3840, ttl=63 (reply in 1…
   15 2.038247757    8.8.8.8         192.168.0.5      ICMP       98 Echo (ping) reply    id=0x6794, seq=15/3840, ttl=45 (request in…
   16 3.005126721    192.168.0.5     8.8.8.8          ICMP       98 Echo (ping) request  id=0x6794, seq=16/4096, ttl=63 (reply in 1…
   17 3.069596889    8.8.8.8         192.168.0.5      ICMP       98 Echo (ping) reply    id=0x6794, seq=16/4096, ttl=45 (request in…
  244 4.006547154    192.168.0.5     8.8.8.8          ICMP       98 Echo (ping) request  id=0x6794, seq=17/4352, ttl=63 (no respons…
  245 4.040716358    8.8.8.8         192.168.0.5      ICMP       98 Echo (ping) reply    id=0x6794, seq=17/4352, ttl=45 (request in…
  268 5.008444534    192.168.0.5     8.8.8.8          ICMP       98 Echo (ping) request  id=0x6794, seq=18/4608, ttl=63 (no respons…
  269 5.050203847    8.8.8.8         192.168.0.5      ICMP       98 Echo (ping) reply    id=0x6794, seq=18/4608, ttl=45 (request in…
```

```
    [Protocols in frame: eth:ethertype:ip:icmp:data]
    [Coloring Rule Name: ICMP]
    [Coloring Rule String: icmp || icmpv6]
▸ Ethernet II, Src: IntelCor_27:1e:ba (60:6c:66:27:1e:ba), Dst: ArrisGro_ea:f8:17 (d4:05:98:ea:f8:17)
▾ Internet Protocol Version 4, Src: 192.168.0.5, Dst: 8.8.8.8
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▸ Differentiated Services Field: 0x04 (DSCP: Unknown, ECN: Not-ECT)
```

Now let us restore the initial rules

nirav@CIA:~$ sudo iptables-restore < ~/iptables-rules

References:

https://www.digitalocean.com/community/tutorials/a-deep-dive-into-iptables-and-netfilter-architecture
http://www.iptables.info/en/structure-of-iptables.html
https://www.booleanworld.com/depth-guide-iptables-linux-firewall/