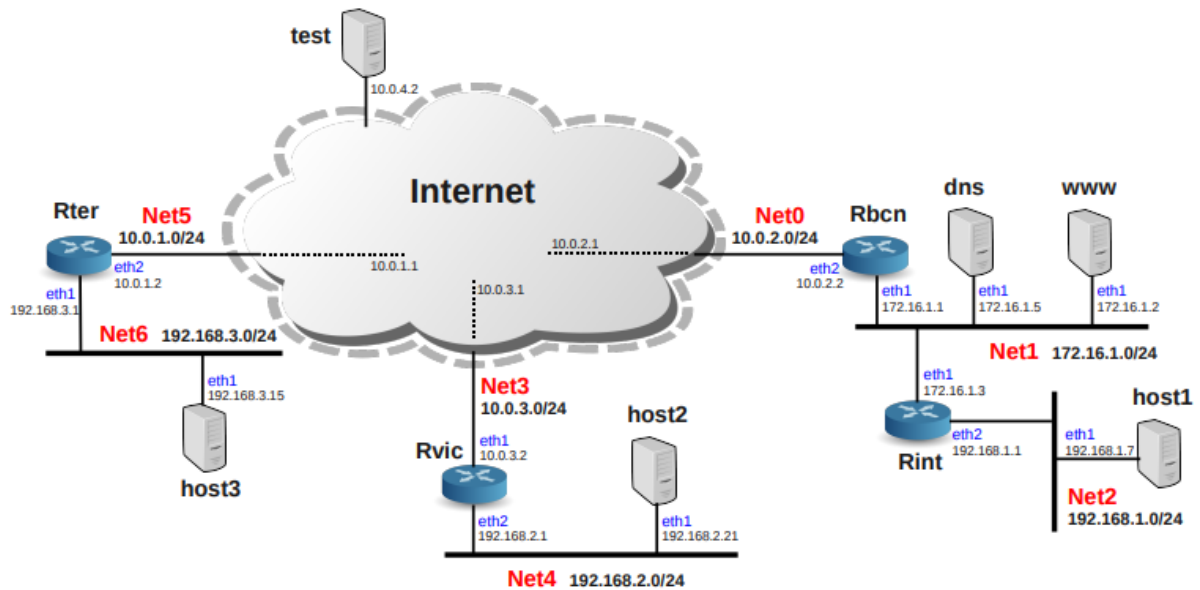


## P8: FWNAT



**Exercise1**– El objetivo de este ejercicio es que se familiarice con los conceptos básicos de filtrado de paquetes. Para la realización de este ejercicio se utilizará el esquema de red mostrado en la figura 1. En primer lugar pondremos en marcha la simulación utilizando el comando:

//from terminal

**simctl fwnat sh**

**start**

**vms**

//from terminal

**simctl fwnat exec ifcfg**

**simctl fwnat exec routecfg**

**get Rbcn**

**get www**

**get Rint**

**get host1**

1. Configure las tablas de filtrado de la máquina host1 de manera que no se permita ningún tipo de tráfico ICMP entrante a los procesos internos (locales) de dicha máquina. Con este filtrado responda a las siguientes preguntas:

```
//from hos1
```

```
iptables -A INPUT -p icmp -j DROP
```

```
iptables -L
```

#-A=append, (las operaciones a hacer), INPUT = (packet filtering), -p(packet matching condition), -j “action” (acciones a realizar)

```
host1:~# iptables -A INPUT -p icmp -j DROP
ip_tables: (C) 2000-2006 Netfilter Core Team
host1:~#
host1:~# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination
DROP       icmp -- anywhere             anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

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

(a) Si desde Rint se ejecuta un ping con destino host1 ¿se transmitirá el correspondiente mensaje ICMP echo-request por la red? ¿Es posible capturar el mensaje de respuesta ICMP echo-reply? Describa lo que ocurre en este caso.

```
//from Rint
```

```
ping -c 1 192.168.1.7
```

```
Rint:~# ping -c 1 192.168.1.7
PING 192.168.1.7 (192.168.1.7) 56(84) bytes of data.

--- 192.168.1.7 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.1.1	192.168.1.7	ICMP	98	Echo (ping) request id=0xa604, seq=1/25

Solo hay un en este echo request y no hay echo reply.

Ocurre porque el ping no llega a ser procesado por host1 ya que tiene bloqueadas las entradas que vengan por protocolo icmp. Entonces no lo procesa, por lo tanto no hay respuesta.

(b) Si en lugar de enviar el ping desde Rint hacia host1, lo hacemos en sentido contrario (ping desde host1 hacia Rint) ¿se transmitirá el correspondiente mensaje ICMP echo-request? ¿Y el echo-reply? Describa lo que ocurre en este caso.

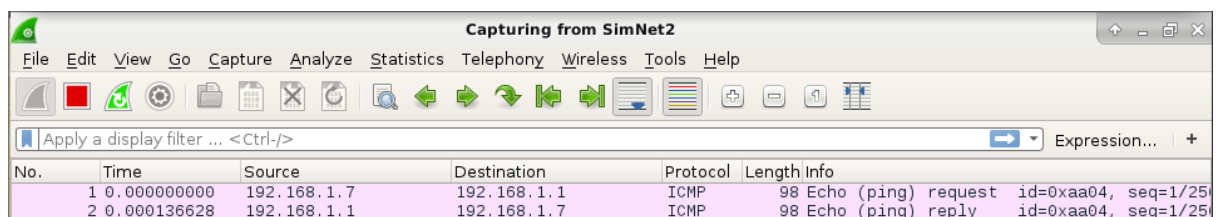
**Se transmitirá el echo-request desde host1 y Rint responderá con un echo-reply. Pasará por la red (SimNet2) pero no llegará a ser procesada por host1, porque el reply esta bloqueado, en el ejercicio anterior.**

```
//from Rint
ifconfig
(eth2 tiene 192.168.1.1)
```

```
//from host1
ping -c 1 192.168.1.1
```

```
host1:~# ping -c 1 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.

--- 192.168.1.1 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.1.7	192.168.1.1	ICMP	98	Echo (ping) request id=0xaa04, seq=1/255
2	0.000136628	192.168.1.1	192.168.1.7	ICMP	98	Echo (ping) reply id=0xaa04, seq=1/255

**Por eso en la SimNet2 se observa el echo-request y el echo-reply, pero desde host1 se ve que se ha perdido, no llega a pasar por el firewall.**

2. Borre la configuración de filtrado anterior de host1 y vuelva a configurar sus tablas de filtrado para obtener el siguiente comportamiento:

```
//from host1
iptables -D INPUT -p icmp -j DROP
iptables -L
```

```
host1:~# iptables -D INPUT -p icmp -j DROP
host1:~# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

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

(a) Desde host1 se debe poder realizar correctamente un ping a una máquina remota (Rint).

//from host1

**ping -c 1 192.168.1.1**

```
host1:~# ping -c 1 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=20.9 ms

--- 192.168.1.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 20.997/20.997/20.997/0.000 ms
```

(b) host1 no debe responder a ninguna petición de ping externa. En esta nueva situación, responde a las mismas preguntas del apartado anterior.

**Para que no responda le quitamos el icmp echo-request a la entrada.**

//from host1

**iptables -A INPUT -p icmp --icmp-type echo-request -j DROP**

```
host1:~# iptables -A INPUT -p icmp --icmp-type echo-request -j DROP
host1:~# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination           icmp echo-request
DROP       icmp -- anywhere             anywhere              icmp echo-request

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

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

//from host1

**ping -c 1 192.168.1.1**

```
host1:~# ping -c 1 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=20.4 ms

--- 192.168.1.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 20.431/20.431/20.431/0.000 ms
```

Capturing from SimNet2						
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:fd:00:00:07:01	Broadcast	ARP	42	Who has 192.168.1.1? Tell 192.168.1.7
2	0.000078565	fe:fd:00:00:06:02	fe:fd:00:00:07:01	ARP	42	192.168.1.1 is at fe:fd:00:00:06:02
3	0.000124526	192.168.1.7	192.168.1.1	ICMP	98	Echo (ping) request id=0xb404, seq=1/255
4	0.000169816	192.168.1.1	192.168.1.7	ICMP	98	Echo (ping) reply id=0xb404, seq=1/255
5	5.015022254	fe:fd:00:00:06:02	fe:fd:00:00:07:01	ARP	42	Who has 192.168.1.7? Tell 192.168.1.1
6	5.015072192	fe:fd:00:00:07:01	fe:fd:00:00:06:02	ARP	42	192.168.1.7 is at fe:fd:00:00:07:01

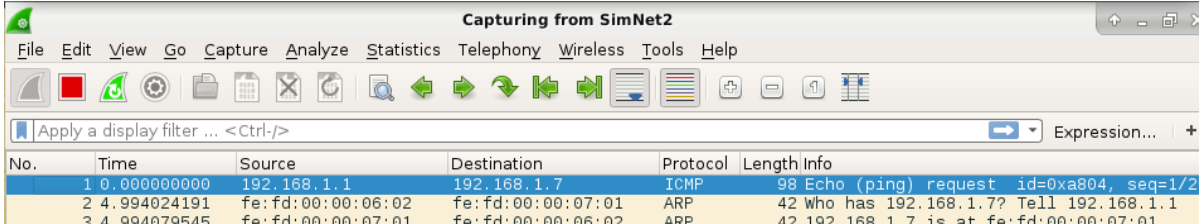
En este caso host1 sí que puede hacer pings y le son respondidos.

//from Rint

**ping -c 1 192.168.1.7**

```
Rint:~# ping -c 1 192.168.1.7
PING 192.168.1.7 (192.168.1.7) 56(84) bytes of data.

--- 192.168.1.7 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.1.1	192.168.1.7	ICMP	98	Echo (ping) request id=0xa804, seq=1/25
2	4.994024191	fe:fd:00:00:06:02	fe:fd:00:00:07:01	ARP	42	who has 192.168.1.7? Tell 192.168.1.1
3	4.994079545	fe:fd:00:00:07:01	fe:fd:00:00:06:02	ARP	42	192.168.1.7 is at fe:fd:00:00:07:01

Si hacemos un ping desde Rint a Host1 podemos ver que solo pasa el echo-request por la red, pero ningún echo-reply ya que Host1 no llega ni a procesar el echo-request.

3. El problema de los esquemas de filtrado anteriores es que hay que configurar las tablas de filtrado en cada una de las máquinas, haciendo que la administración de la red sea compleja. La solución más utilizada es “confiar” la seguridad al router de la red, ya que todas las comunicaciones con el exterior fluyen a través de él y se puede aplicar un control centralizado a las mismas facilitando la administración. Cuando un router realiza funciones de filtrado o firewall se le conoce con el nombre de bastión de la red. En este punto, usted tiene que configurar el router Rint como bastión para proteger a los hosts internos (host1). Para ello prepare el escenario realizando las siguientes tareas:

- Elimine las entradas de las tablas de filtrado de host1.

```
//from host1
```

```
iptables -F #flush = deleting all the route
```

```
iptables -L
```

```
host1:~# iptables -F
host1:~# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

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

- Verifique que las redes Net1 y Net2 están correctamente configuradas (direcciones IP y tablas de encaminamiento) de forma que exista conectividad entre ellas a nivel IP. En este momento, debe ser posible realizar con éxito un ping desde www o Rbcn a cualquiera de las máquinas de la Net2 y viceversa.

```
//from www to host1
```

```
ping -c 1 192.168.1.7
```

```

www:~# ping -c 1 192.168.1.7
PING 192.168.1.7 (192.168.1.7) 56(84) bytes of data.
From 172.16.1.1: icmp_seq=1 Redirect Host(New nexthop: 172.16.1.3)
64 bytes from 192.168.1.7: icmp_seq=1 ttl=63 time=56.1 ms

--- 192.168.1.7 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 56.125/56.125/56.125/0.000 ms

```

//from Rbcn to host1

ping -c 1 192.168.1.7

```

Rbcn:~# ping -c 1 192.168.1.7
PING 192.168.1.7 (192.168.1.7) 56(84) bytes of data.
64 bytes from 192.168.1.7: icmp_seq=1 ttl=63 time=0.376 ms

--- 192.168.1.7 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.376/0.376/0.376/0.000 ms

```

Ahora, añada las entradas necesarias en su bastión (Rint) para obtener el siguiente comportamiento:

(a) Un ping realizado desde una máquina externa a Net2 hacia una máquina perteneciente a Net2 no debe ser respondido, pero en el caso contrario, es decir un ping iniciado desde una máquina de Net2 hacia una máquina externa, sí que debe funcionar correctamente.

Verifique el funcionamiento de este filtro.

//from Rint

iptables -A FORWARD -p icmp --icmp-type echo-request -i eth1 -j DROP

```

Rint:~# iptables -A FORWARD -p icmp --icmp-type echo-request -i eth1 -j DROP
Rint:~# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
DROP       icmp -- anywhere             anywhere             icmp echo-request

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

```

//from www to host1

ping -c 1 192.168.1.7

```

www:~# ping -c 1 192.168.1.7
PING 192.168.1.7 (192.168.1.7) 56(84) bytes of data.
From 172.16.1.1: icmp_seq=1 Redirect Host(New nexthop: 172.16.1.3)

--- 192.168.1.7 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

```

Capturing from SimNet1						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	172.16.1.2	192.168.1.7	ICMP	98	Echo (ping) request id=0xa704, seq=1/
2	0.000133245	172.16.1.1	172.16.1.2	ICMP	126	Redirect (Redirect for hos
3	0.000143125	172.16.1.2	192.168.1.7	ICMP	98	Echo (ping) request id=0xa704, seq=1/
4	5.002050323	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42	Who has 172.16.1.2? Tell 172.16.1.1
5	5.002178122	fe:fd:00:00:04:01	fe:fd:00:00:03:01	ARP	42	172.16.1.2 is at fe:fd:00:00:04:01
6	5.002266016	fe:fd:00:00:03:01	fe:fd:00:00:06:01	ARP	42	Who has 172.16.1.3? Tell 172.16.1.1
7	5.002308762	fe:fd:00:00:06:01	fe:fd:00:00:03:01	ARP	42	172.16.1.3 is at fe:fd:00:00:06:01

Capturing from SimNet2						
No.	Time	Source	Destination	Protocol	Length	Info

//from host1 to www  
ping -c 1 172.16.1.2

```
host1:~# ping -c 1 172.16.1.2
PING 172.16.1.2 (172.16.1.2) 56(84) bytes of data.
64 bytes from 172.16.1.2: icmp_seq=1 ttl=62 time=57.9 ms

--- 172.16.1.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 57.914/57.914/57.914/0.000 ms
```

Capturing from SimNet2						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	fe:fd:00:00:07:01	Broadcast	ARP	42	Who has 192.168.1.1? Tell 192.168.1.7
2	0.000081017	fe:fd:00:00:06:02	fe:fd:00:00:07:01	ARP	42	192.168.1.1 is at fe:fd:00:00:06:02
3	0.000135540	192.168.1.7	172.16.1.2	ICMP	98	Echo (ping) request id=0xbc04, seq=1/25
4	0.036708880	172.16.1.2	192.168.1.7	ICMP	98	Echo (ping) reply id=0xbc04, seq=1/25
5	5.042362290	fe:fd:00:00:06:02	fe:fd:00:00:07:01	ARP	42	Who has 192.168.1.7? Tell 192.168.1.1
6	5.042449035	fe:fd:00:00:07:01	fe:fd:00:00:06:02	ARP	42	192.168.1.7 is at fe:fd:00:00:07:01

Capturing from SimNet1						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	fe:fd:00:00:06:01	Broadcast	ARP	42	Who has 172.16.1.2? Tell 172.16.1.3
2	0.000079585	fe:fd:00:00:04:01	fe:fd:00:00:06:01	ARP	42	172.16.1.2 is at fe:fd:00:00:04:01
3	0.000197508	192.168.1.7	172.16.1.2	ICMP	98	Echo (ping) request id=0xbc04, seq=1/
4	0.021110175	fe:fd:00:00:04:01	Broadcast	ARP	42	Who has 172.16.1.1? Tell 172.16.1.2
5	0.021198978	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42	172.16.1.1 is at fe:fd:00:00:03:01
6	0.021289238	172.16.1.2	192.168.1.7	ICMP	98	Echo (ping) reply id=0xbc04, seq=1/
7	0.021419498	172.16.1.1	172.16.1.2	ICMP	126	Redirect (Redirect for hos
8	0.021430466	172.16.1.2	192.168.1.7	ICMP	98	Echo (ping) reply id=0xbc04, seq=1/
9	5.026973015	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42	Who has 172.16.1.2? Tell 172.16.1.1
10	5.027007635	fe:fd:00:00:03:01	fe:fd:00:00:06:01	ARP	42	Who has 172.16.1.3? Tell 172.16.1.1
11	5.027373063	fe:fd:00:00:06:01	fe:fd:00:00:03:01	ARP	42	172.16.1.3 is at fe:fd:00:00:06:01
12	5.027438538	fe:fd:00:00:04:01	fe:fd:00:00:03:01	ARP	42	172.16.1.2 is at fe:fd:00:00:04:01

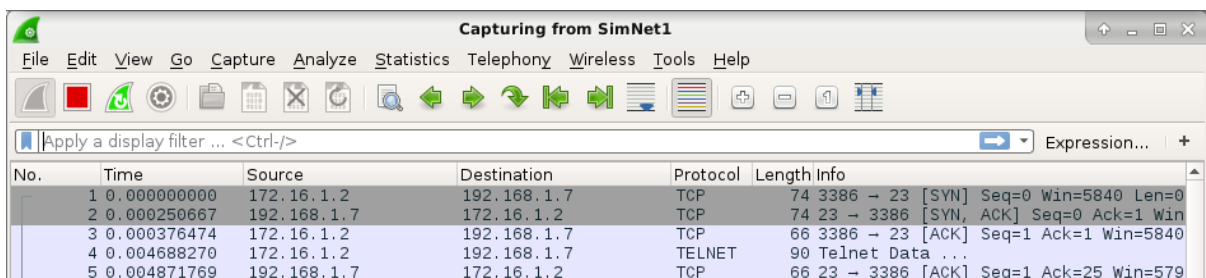
Se puede observar como se puede enviar un ping de host1 a www, pero no al revés.  
Nuestra regla funciona.

(b) Si una máquina de una red externa a Net2, realiza un intento de conexión a un servicio TCP de un servidor alojado en Net2, este intento de conexión debe ser rechazado, pero en el caso contrario sí que debe funcionar correctamente. Verifique el funcionamiento de este filtro utilizando las máquinas de Net1 como red externa de pruebas.

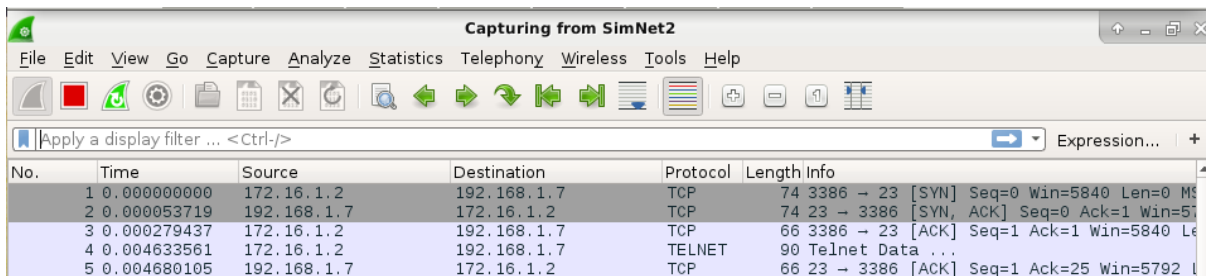
//from www to host1

telnet 192.168.1.7

```
www:~# telnet 192.168.1.7
Trying 192.168.1.7...
Connected to 192.168.1.7.
Escape character is '^]'.
Debian GNU/Linux 5.0
host1 login:
```



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	172.16.1.2	192.168.1.7	TCP	74	3386 → 23 [SYN] Seq=0 Win=5840 Len=0
2	0.000250667	192.168.1.7	172.16.1.2	TCP	74	23 → 3386 [SYN, ACK] Seq=0 Ack=1 Win=5840
3	0.000376474	172.16.1.2	192.168.1.7	TCP	66	3386 → 23 [ACK] Seq=1 Ack=1 Win=5840
4	0.004688270	172.16.1.2	192.168.1.7	TELNET	90	Telnet Data ...
5	0.004871769	192.168.1.7	172.16.1.2	TCP	66	23 → 3386 [ACK] Seq=1 Ack=25 Win=579



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	172.16.1.2	192.168.1.7	TCP	74	3386 → 23 [SYN] Seq=0 Win=5840 Len=0
2	0.000053719	192.168.1.7	172.16.1.2	TCP	74	23 → 3386 [SYN, ACK] Seq=0 Ack=1 Win=5840
3	0.000279437	172.16.1.2	192.168.1.7	TCP	66	3386 → 23 [ACK] Seq=1 Ack=1 Win=5840
4	0.004633561	172.16.1.2	192.168.1.7	TELNET	90	Telnet Data ...
5	0.004680105	192.168.1.7	172.16.1.2	TCP	66	23 → 3386 [ACK] Seq=1 Ack=25 Win=5792

Usamos telnet porque funciona sobre tcp.

//from Rint

(manera finolis y versátil)

**iptables -A FORWARD -i eth1 -p tcp --tcp-flags SYN, ACK SYN -j DROP**

//después de --tcp-flags la primera condición son los flags que miras y la segunda es la selección de los que están activos.

```
Rint:~# iptables -A FORWARD -i eth1 -p tcp --tcp-flags SYN,ACK SYN -j DROP
Rint:~# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination
DROP      icmp -- anywhere             anywhere             icmp echo-request
DROP      tcp  -- anywhere             anywhere             tcp flags:SYN,ACK,SYN

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

//from www to host1

telnet 192.168.1.7



```
www:~# telnet 192.168.1.7
Trying 192.168.1.7...
telnet: Unable to connect to remote host: Connection timed out
```

Capturing from SimNet1

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	fe80::6026:f5ff:fef...	ff02::2	ICMPv6	70	Router Solicitation from 62:26:f5:f2:b7:c1
2	90.477876070	fe:fd:00:00:04:01	Broadcast	ARP	42	Who has 172.16.1.1? Tell 172.16.1.2
3	90.478069733	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42	172.16.1.1 is at fe:fd:00:00:03:01
4	90.478202489	172.16.1.2	192.168.1.7	TCP	74	2901 → 23 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=366
5	90.478324596	172.16.1.1	172.16.1.2	ICMP	102	Redirect (Redirect for host)
6	90.495753729	fe:fd:00:00:03:01	Broadcast	ARP	42	Who has 172.16.1.3? Tell 172.16.1.1
7	90.495869200	fe:fd:00:00:06:01	fe:fd:00:00:03:01	ARP	42	172.16.1.3 is at fe:fd:00:00:06:01
8	90.495954441	172.16.1.2	192.168.1.7	TCP	74	[TCP Retransmission] 2901 → 23 [SYN] Seq=0 Win=5840 Len=0 MSS=1460
9	90.498655523	fe:fd:00:00:04:01	Broadcast	ARP	42	Who has 172.16.1.3? Tell 172.16.1.2
10	90.498812817	fe:fd:00:00:06:01	fe:fd:00:00:04:01	ARP	42	172.16.1.3 is at fe:fd:00:00:06:01
11	93.403841946	172.16.1.2	192.168.1.7	TCP	74	[TCP Retransmission] 2901 → 23 [SYN] Seq=0 Win=5840 Len=0 MSS=1460
12	93.403967285	172.16.1.1	172.16.1.2	ICMP	102	Redirect (Redirect for host)
13	93.404174986	172.16.1.2	192.168.1.7	TCP	74	[TCP Retransmission] 2901 → 23 [SYN] Seq=0 Win=5840 Len=0 MSS=1460
14	98.419451682	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42	Who has 172.16.1.2? Tell 172.16.1.1
15	98.419586094	fe:fd:00:00:04:01	fe:fd:00:00:03:01	ARP	42	172.16.1.2 is at fe:fd:00:00:04:01
16	99.404734417	172.16.1.2	192.168.1.7	TCP	74	[TCP Retransmission] 2901 → 23 [SYN] Seq=0 Win=5840 Len=0 MSS=1460
17	111.411601755	172.16.1.2	192.168.1.7	TCP	74	[TCP Retransmission] 2901 → 23 [SYN] Seq=0 Win=5840 Len=0 MSS=1460
18	111.411766695	172.16.1.1	172.16.1.2	ICMP	102	Redirect (Redirect for host)
19	111.411784296	172.16.1.2	192.168.1.7	TCP	74	[TCP Retransmission] 2901 → 23 [SYN] Seq=0 Win=5840 Len=0 MSS=1460

Capturing from SimNet2

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	fe80::749a:47ff:fe4...	ff02::2	ICMPv6	70	Router Solicitation from 76:9a:47:44:aa:f
2	278.528034439	fe80::749a:47ff:fe4...	ff02::2	ICMPv6	70	Router Solicitation from 76:9a:47:44:aa:f

Se lanza el primer SYN del handshake de TCP, pero no recibe respuesta ya que no llega a la SimNet2.

```
//from host1 to www
telnet 172.16.1.2
```

```
host1:~# telnet 172.16.1.2
Trying 172.16.1.2...
Connected to 172.16.1.2.
Escape character is '^]'.
Debian GNU/Linux 5.0
www login: █
```

Capturing from SimNet2

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.1.7	172.16.1.2	TCP	74	2579 → 23 [SYN] Seq=0 Win=5840 Len=0 MS
2	0.000295025	172.16.1.2	192.168.1.7	TCP	74	23 → 2579 [SYN, ACK] Seq=0 Ack=1 Win=57
3	0.000376645	192.168.1.7	172.16.1.2	TCP	66	2579 → 23 [ACK] Seq=1 Ack=1 Win=5840 Le
4	0.001648815	192.168.1.7	172.16.1.2	TELNET	90	Telnet Data ...
5	0.001901649	172.16.1.2	192.168.1.7	TCP	66	23 → 2579 [ACK] Seq=1 Ack=25 Win=5792 L
6	0.083027047	172.16.1.2	192.168.1.7	TELNET	78	Telnet Data ...
7	0.083133101	192.168.1.7	172.16.1.2	TCP	66	2579 → 23 [ACK] Seq=25 Ack=13 Win=5840
8	0.083364959	192.168.1.7	172.16.1.2	TELNET	69	Telnet Data ...
9	0.083784687	172.16.1.2	192.168.1.7	TELNET	99	Telnet Data ...
10	0.084366625	192.168.1.7	172.16.1.2	TELNET	109	Telnet Data ...
11	0.102469067	172.16.1.2	192.168.1.7	TELNET	69	Telnet Data ...
12	0.102642425	192.168.1.7	172.16.1.2	TELNET	69	Telnet Data ...
13	0.103675893	172.16.1.2	192.168.1.7	TELNET	69	Telnet Data ...

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.1.7	172.16.1.2	TCP	74	2579 → 23 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=68830 TSecr=71425
2	0.000158904	172.16.1.2	192.168.1.7	TCP	74	23 → 2579 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1 TSval=68830 TSecr=71425
3	0.000376745	192.168.1.7	172.16.1.2	TCP	66	2579 → 23 [ACK] Seq=1 Ack=1 Win=5840 Len=0 TSval=68830 TSecr=71425
4	0.001633535	192.168.1.7	172.16.1.2	TELNET	90	Telnet Data ...
5	0.001737177	172.16.1.2	192.168.1.7	TCP	66	23 → 2579 [ACK] Seq=1 Ack=25 Win=5792 Len=0 TSval=71425 TSecr=68830
6	0.077568686	172.16.1.2	172.16.1.5	DNS	84	Standard query 0xaa20 PTR 7.1.168.192.in-addr.arpa
7	0.078014296	172.16.1.5	172.16.1.2	DNS	151	Standard query response 0xaa20 PTR 7.1.168.192.in-addr.arpa PTR host1
8	0.080423242	172.16.1.2	172.16.1.5	DNS	77	Standard query 0x4309 AAAA www.practnet.tcgi
9	0.080868530	172.16.1.5	172.16.1.2	DNS	121	Standard query response 0x4309 AAAA www.practnet.tcgi SOA dns.practnet.tcgi
10	0.081609769	172.16.1.2	172.16.1.5	DNS	77	Standard query 0xaf6 A www.practnet.tcgi
11	0.082105661	172.16.1.5	172.16.1.2	DNS	127	Standard query response 0xaf6 A www.practnet.tcgi A 172.16.1.2 NS dns.practnet.tcgi
12	0.082855131	172.16.1.2	192.168.1.7	TELNET	78	Telnet Data ...
13	0.083120713	192.168.1.7	172.16.1.2	TCP	66	2579 → 23 [ACK] Seq=25 Ack=13 Win=5840 Len=0 TSval=68842 TSecr=71425
14	0.083353863	192.168.1.7	172.16.1.2	TELNET	69	Telnet Data ...
15	0.083628935	172.16.1.2	192.168.1.7	TELNET	99	Telnet Data ...
16	0.084354450	192.168.1.7	172.16.1.2	TELNET	109	Telnet Data ...
17	0.102290438	172.16.1.2	192.168.1.7	TELNET	69	Telnet Data ...
18	0.102621354	192.168.1.7	172.16.1.2	TELNET	69	Telnet Data ...
19	0.103506555	172.16.1.2	192.168.1.7	TELNET	69	Telnet Data ...

Se puede observar que con esta nueva regla solo se puede establecer una conexión TCP desde dentro de la red SimNet2 hacia afuera y no al revés, ya que lo que hace Rint es descartar los paquetes SYN individuales que llegan de fuera.

(c) Finalmente, filtre todo el tráfico UDP que entre o salga de Net2, excepto el tráfico UDP que vaya dirigido a un servidor DNS (que se supone externo a Net2)

//from Rint

**iptables -A FORWARD --protocol udp --source-port ! 53 --destination-port ! 53 -j DROP**

//usamos el netcat en modo udp para las pruebas

//from www

nc -l -u -p 77

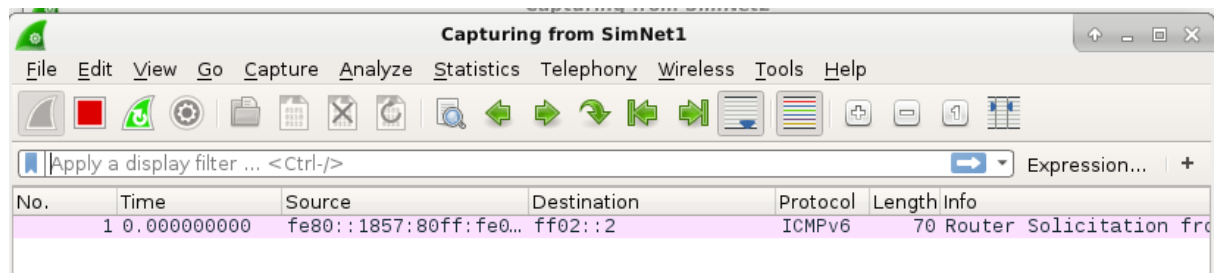
//from host1

nc -u 172.16.1.2 77

```
www:~# nc -l -u -p 77
```

```
host1:~# nc -u 172.16.1.2 77
Hola soy host1
responde porfa
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.1.7	172.16.1.2	UDP	57	2049 → 77 Len=15
2	5.015193499	fe:fd:00:00:07:01	fe:fd:00:00:06:02	ARP	42	who has 192.168.1.1? Tell 192.168.1.1
3	5.015357706	fe:fd:00:00:06:02	fe:fd:00:00:07:01	ARP	42	192.168.1.1 is at fe:fd:00:00:06:02
4	8.374166466	192.168.1.7	172.16.1.2	UDP	57	2049 → 77 Len=15



**Se puede observar que el tráfico enviado por udp se queda en SimNet2 y no llega a SimNet1 por la nueva regla de Rint. Lo mismo pasará si www y host1 cambian de papeles.**

**Problema: en el nc se puede usar el puerto 53, cosa que permite una conexión udp que no es dns**

**Exercise2**– El objetivo de este ejercicio es que usted se familiarice con las técnicas de NAT. Se utilizará el mismo escenario mostrado en la figura 1. Si usted se centra en la redes formadas por Net0, Net1 y Net2, puede observar que desde un punto de vista administrativo respecto al espacio de direcciones IP, la red de la figura anterior se puede ver de la siguiente manera:

En este caso se ha considerado que los rangos de direcciones 192.168.0.0/22 y 172.16.1.0/24 se corresponden con direcciones privadas. Por otro lado, se ha considerado que los rangos de direcciones 10.0.0.0/22 hacen referencia a un sistema de direccionamiento público (en la figura se ha considerado que Internet hace uso del rango 10.0.0.0/22) Arranque la simulación ejecutando desde el host de virtualización el comando:

```
//from terminal
simctl fwnat start
simctl fwnat exec ifcfg
simctl fwnat exec routecfg
simctl fwnat exec fwcfg
```

1. Desde el host www de Net1, realice un ping a 10.0.4.2 (test) ¿funciona? ¿Es un problema de filtrado o de direccionamiento?

```
//from host1 to test
ping -c 1 10.0.4.2
```

```
host1:~# ping -c 1 10.0.4.2
PING 10.0.4.2 (10.0.4.2) 56(84) bytes of data.

--- 10.0.4.2 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```

Capturing from SimNet2					
No.	Time	Source	Destination	Protocol	Length Info
1	0.000000000	192.168.1.7	10.0.4.2	ICMP	98 Echo (ping) request id=0xaa04, seq=1
2	5.001294319	fe:fd:00:00:07:01	fe:fd:00:00:06:02	ARP	42 Who has 192.168.1.1? Tell 192.168.1.1
3	5.001361216	fe:fd:00:00:06:02	fe:fd:00:00:07:01	ARP	42 192.168.1.1 is at fe:fd:00:00:06:02

Capturing from SimNet1					
No.	Time	Source	Destination	Protocol	Length Info
1	0.000000000	192.168.1.7	10.0.4.2	ICMP	98 Echo (ping) request id=0xaa04, seq=1
2	5.011382829	fe:fd:00:00:06:01	fe:fd:00:00:03:01	ARP	42 Who has 172.16.1.1? Tell 172.16.1.3
3	5.011493476	fe:fd:00:00:03:01	fe:fd:00:00:06:01	ARP	42 172.16.1.1 is at fe:fd:00:00:03:01

Capturing from SimNet0					
No.	Time	Source	Destination	Protocol	Length Info
1	0.000000000	192.168.1.7	10.0.4.2	ICMP	98 Echo (ping) request id=0xaa04, seq=1/
2	5.000787943	fe:fd:00:00:03:02	fe:fd:00:00:01:01	ARP	42 Who has 10.0.2.1? Tell 10.0.2.2
3	5.000877207	fe:fd:00:00:01:01	fe:fd:00:00:03:02	ARP	42 10.0.2.1 is at fe:fd:00:00:01:01

**El echo-request de host llega hasta la SimNet0, pero no a test. No hay ninguna regla en iptables de test por lo que implica que es un problema de direccionamiento. Aparte la ip es privada, habria que hacer la traduccion**

2. Para solucionar el problema anterior configure el router externo Rbcn para que realice SNAT para sus redes internas. Una vez configurado pruebe a realizar el ping a 10.0.4.2 ¿funciona ahora? Utilice las herramientas de análisis de tráfico que conoce para ver que está sucediendo en la red

**//from Rbcn**

**iptables -t nat -A POSTROUTING -o eth2 -j SNAT --to 10.0.2.2**

#-o = output, -j acction, --to= destination, POSTROUTING=source ip translation( una vez #encaminado),

```
Rbcn:~# iptables -t nat -A POSTROUTING -o eth2 -j SNAT --to 10.0.2.2
ip_tables: (C) 2000-2006 Netfilter Core Team
Netfilter messages via NETLINK v0.30.
ip_conntrack version 2.4 (474 buckets, 3792 max) - 224 bytes per conntrack
Rbcn:~#
```

**//from www**

**ping -c 1 10.0.4.2**

```
www:~# ping -c 1 10.0.4.2
PING 10.0.4.2 (10.0.4.2) 56(84) bytes of data.
64 bytes from 10.0.4.2: icmp_seq=1 ttl=62 time=61.7 ms

--- 10.0.4.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0
rtt min/avg/max/mdev = 61.795/61.795/61.795/0.000 ms
```

Capturing from SimNet1					
No.	Time	Source	Destination	Protocol	Length Info
1	0.000000000	fe:fd:00:00:04:01	Broadcast	ARP	42 Who has 172.16.1.1? Tell 172.16.1.2
2	0.000649857	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42 172.16.1.1 is at fe:fd:00:00:03:01
3	0.000899069	172.16.1.2	10.0.4.2	ICMP	98 Echo (ping) request id=0xa904, seq=1/256
4	0.040947798	10.0.4.2	172.16.1.2	ICMP	98 Echo (ping) reply id=0xa904, seq=1/256
5	5.053427001	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42 Who has 172.16.1.2? Tell 172.16.1.1
6	5.053717892	fe:fd:00:00:04:01	fe:fd:00:00:03:01	ARP	42 172.16.1.2 is at fe:fd:00:00:04:01

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	fe:fd:00:00:03:02	Broadcast	ARP	42	Who has 10.0.2.1? Tell 10.0.2.2
2	0.000274966	fe:fd:00:00:01:01	fe:fd:00:00:03:02	ARP	42	10.0.2.1 is at fe:fd:00:00:01:01
3	0.000390437	10.0.2.2	10.0.4.2	ICMP	98	Echo (ping) request id=0xa904, seq=1/256
4	0.021135542	10.0.4.2	10.0.2.2	ICMP	98	Echo (ping) reply id=0xa904, seq=1/256
5	5.033326917	fe:fd:00:00:01:01	fe:fd:00:00:03:02	ARP	42	Who has 10.0.2.2? Tell 10.0.2.1
6	5.034103207	fe:fd:00:00:03:02	fe:fd:00:00:01:01	ARP	42	10.0.2.2 is at fe:fd:00:00:03:02

Con el comando **-t nat....** estamos diciendo al Rbcn que todos los paquetes que vayan a salir por la eth2 se les cambia la direccion IP de origen por la de esa interfaz del propio router.

El router se encarga de traducir la direccion privada a 10.0.2.2 y al viceversa.

3. La figura muestra el típico esquema de firewall con doble bastión (bastion externo –Rbcn– y bastión interno –Rint–), zona desmilitarizada (Net1) o DMZ (DeMilitarized Zone) para los servidores con acceso externo, y red interna (Net2). En este esquema las máquinas de la red interna pueden establecer conexiones a los servidores de la DMZ y a servidores externos (Internet), tal y como se ha configurado en el ejercicio anterior. En este esquema de doble bastión, se debe poder acceder a los servidores de la DMZ desde el exterior pero no a los hosts de la red interna.

Configure el bastión externo (Rbcn) para dar acceso al servidor web de www desde Internet y haga uso de la máquina externa (test) para verificar la configuración. Utilice las herramientas de análisis de tráfico que conoce para ver que está sucediendo en la red.

//from test

**ping -c 1 172.16.1.2 # we do ping to www**

```
test:~# ping -c 1 172.16.1.2
PING 172.16.1.2 (172.16.1.2) 56(84) bytes of data.
From 10.0.4.1 icmp_seq=1 Destination Net Unreachable

--- 172.16.1.2 ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms
```

//from Rbcn

**iptables -t nat -A PREROUTING -i eth2 -d 10.0.2.2 -j DNAT --to 172.16.1.2**

i= input (destino del paquete), -j= traduce la ip del destino antes de routing

```
Rbcn.15855.0
File Edit View Search Terminal Help
Rbcn:~# iptables -t nat -A PREROUTING -i eth2 -d 10.0.2.2 -j DNAT --to 172.16.1.2
Rbcn:~# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
Rbcn:~#
```

//from host1

**ping -c 1 10.0.4.2**

```
host1:~# ping -c 1 10.0.4.2
PING 10.0.4.2 (10.0.4.2) 56(84) bytes of data.
64 bytes from 10.0.4.2: icmp_seq=1 ttl=61 time=45.9 ms

--- 10.0.4.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 45.949/45.949/45.949/0.000 ms
```

Capturing from SimNet1						
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:fd:00:00:06:01	Broadcast	ARP	42	Who has 172.16.1.1? Tell 172.16.1.3
2	0.000246488	fe:fd:00:00:03:01	fe:fd:00:00:06:01	ARP	42	172.16.1.1 is at fe:fd:00:00:03:01
3	0.000422130	192.168.1.7	10.0.4.2	ICMP	98	Echo (ping) request id=0x9904, seq=1/256
4	0.012133689	10.0.4.2	192.168.1.7	ICMP	98	Echo (ping) reply id=0x9904, seq=1/256
5	5.015985933	fe:fd:00:00:03:01	fe:fd:00:00:06:01	ARP	42	Who has 172.16.1.3? Tell 172.16.1.1
6	5.016234885	fe:fd:00:00:06:01	fe:fd:00:00:03:01	ARP	42	172.16.1.3 is at fe:fd:00:00:06:01

Capturing from SimNet2						
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	fe80::e8e3:ebff:fe9...	ff02::2	ICMPv6	70	Router Solicitation from ea:e3:eb:9a:03:...
2	733.165630811	fe:fd:00:00:07:01	Broadcast	ARP	42	Who has 192.168.1.1? Tell 192.168.1.7
3	733.165842682	fe:fd:00:00:06:02	fe:fd:00:00:07:01	ARP	42	192.168.1.1 is at fe:fd:00:00:06:02
4	733.165937322	192.168.1.7	10.0.4.2	ICMP	98	Echo (ping) request id=0x9904, seq=1/256
5	733.190603836	10.0.4.2	192.168.1.7	ICMP	98	Echo (ping) reply id=0x9904, seq=1/256
6	738.194450993	fe:fd:00:00:06:02	fe:fd:00:00:07:01	ARP	42	Who has 192.168.1.7? Tell 192.168.1.1
7	738.194500000	fe:fd:00:00:07:01	fe:fd:00:00:06:02	ARP	42	192.168.1.7 is at fe:fd:00:00:07:01

The image shows the Wireshark network protocol analyzer interface. The title bar reads "Capturing from SimNet0". The menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, and Help. Below the menu is a toolbar with various icons for packet capture and analysis. The main display area shows a list of captured packets. The first packet is an ICMP Echo (ping) request from 10.0.2.2 to 10.0.2.1. The fourth packet is the first echo request, and the fifth is the corresponding reply. The packet details pane on the right shows the structure of the selected packet, including Ethernet II, Internet Protocol Version 4, and ICMP Echo (ping) request.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	fe80::789e:7aff:fee...	ff02::2	ICMPv6	70	Router Solicitation from 7a:9e:7a:e1:b8:c
2	864.261680871	fe:fd:00:00:03:02	Broadcast	ARP	42	Who has 10.0.2.1? Tell 10.0.2.2
3	864.262064056	fe:fd:00:00:01:01	fe:fd:00:00:03:02	ARP	42	10.0.2.1 is at fe:fd:00:00:01:01
4	864.262153108	10.0.2.2	10.0.4.2	ICMP	98	Echo (ping) request id=0x9904, seq=1/25
5	864.262464423	10.0.4.2	10.0.2.2	ICMP	98	Echo (ping) reply id=0x9904, seq=1/25
6	869.266890784	fe:fd:00:00:01:01	fe:fd:00:00:03:02	ARP	42	Who has 10.0.2.2? Tell 10.0.2.1
7	869.267013201	fe:fd:00:00:03:02	fe:fd:00:00:01:01	ARP	42	10.0.2.2 is at fe:fd:00:00:03:02