

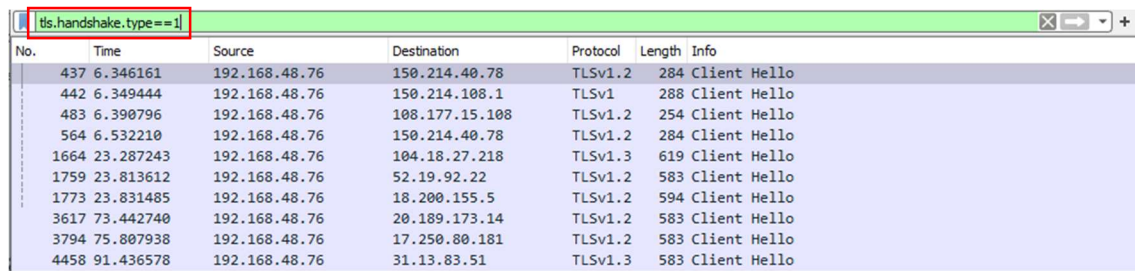
Práctica: Análisis de Protocolos y Cortafuegos

Ejercicio 1

1. ¿Cuántas conexiones TLS se establecen y con cuantos servidores diferentes? ¿Qué filtro de Wireshark te permite responder fácilmente a esta pregunta?

Realiza 10 conexiones TLS y con 9 servidores diferentes. He utilizado el filtro `tls.handshake.type==1`, pues muestra las tramas de protocolo TLS que contienen un mensaje del tipo “Client Hello”, que se envían por el cliente al servidor al iniciar la conexión TLS. Los servidores con los que realiza conexión son:

- 150.214.40.78
- 150.214.108.1
- 108.177.15.108
- 104.18.27.218
- 52.19.92.22
- 18.200.155.5
- 20.189.173.14
- 17.250.80.181
- 31.13.83.51

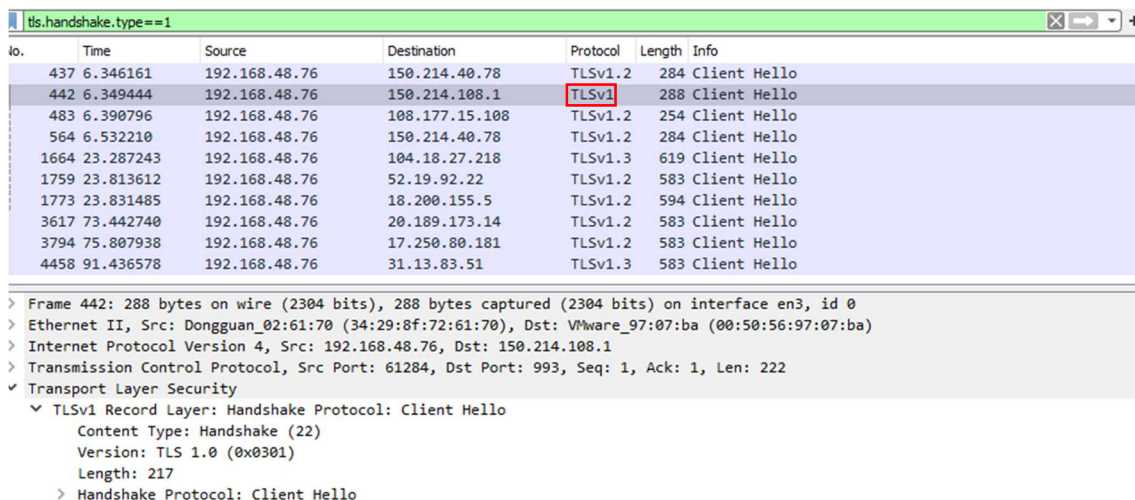


The screenshot shows a Wireshark packet capture with a filter `tls.handshake.type==1` applied. The packet list contains 10 entries, all of type 'Client Hello' (type 1). The destinations are: 150.214.40.78, 150.214.108.1, 108.177.15.108, 150.214.40.78, 104.18.27.218, 52.19.92.22, 18.200.155.5, 20.189.173.14, 17.250.80.181, and 31.13.83.51.

No.	Time	Source	Destination	Protocol	Length	Info
437	6.346161	192.168.48.76	150.214.40.78	TLSv1.2	284	Client Hello
442	6.349444	192.168.48.76	150.214.108.1	TLSv1	288	Client Hello
483	6.390796	192.168.48.76	108.177.15.108	TLSv1.2	254	Client Hello
564	6.532210	192.168.48.76	150.214.40.78	TLSv1.2	284	Client Hello
1664	23.287243	192.168.48.76	104.18.27.218	TLSv1.3	619	Client Hello
1759	23.813612	192.168.48.76	52.19.92.22	TLSv1.2	583	Client Hello
1773	23.831485	192.168.48.76	18.200.155.5	TLSv1.2	594	Client Hello
3617	73.442740	192.168.48.76	20.189.173.14	TLSv1.2	583	Client Hello
3794	75.807938	192.168.48.76	17.250.80.181	TLSv1.2	583	Client Hello
4458	91.436578	192.168.48.76	31.13.83.51	TLSv1.3	583	Client Hello

2. ¿Qué versión de TLS se utiliza en la conexión con el host 150.214.108.1? ¿Y con el host 150.214.40.78?

Con el host 150.214.108.1 utiliza la versión TLS 1



The screenshot shows the details pane for packet 442, which is a 'Client Hello' message to 150.214.108.1. The 'Protocol' field is highlighted as 'TLSv1'. The 'Content Type' is 'Handshake (22)', the 'Version' is 'TLS 1.0 (0x0301)', and the 'Length' is 217. The 'Handshake Protocol' is 'Client Hello'.

No.	Time	Source	Destination	Protocol	Length	Info
437	6.346161	192.168.48.76	150.214.40.78	TLSv1.2	284	Client Hello
442	6.349444	192.168.48.76	150.214.108.1	TLSv1	288	Client Hello
483	6.390796	192.168.48.76	108.177.15.108	TLSv1.2	254	Client Hello
564	6.532210	192.168.48.76	150.214.40.78	TLSv1.2	284	Client Hello
1664	23.287243	192.168.48.76	104.18.27.218	TLSv1.3	619	Client Hello
1759	23.813612	192.168.48.76	52.19.92.22	TLSv1.2	583	Client Hello
1773	23.831485	192.168.48.76	18.200.155.5	TLSv1.2	594	Client Hello
3617	73.442740	192.168.48.76	20.189.173.14	TLSv1.2	583	Client Hello
3794	75.807938	192.168.48.76	17.250.80.181	TLSv1.2	583	Client Hello
4458	91.436578	192.168.48.76	31.13.83.51	TLSv1.3	583	Client Hello

> Frame 442: 288 bytes on wire (2304 bits), 288 bytes captured (2304 bits) on interface en3, id 0
> Ethernet II, Src: Dongguan_02:61:70 (34:29:8f:72:61:70), Dst: VMware_97:07:ba (00:50:56:97:07:ba)
> Internet Protocol Version 4, Src: 192.168.48.76, Dst: 150.214.108.1
> Transmission Control Protocol, Src Port: 61284, Dst Port: 993, Seq: 1, Ack: 1, Len: 222
✓ Transport Layer Security
 ✓ TLSv1 Record Layer: Handshake Protocol: Client Hello
 Content Type: Handshake (22)
 Version: TLS 1.0 (0x0301)
 Length: 217
 > Handshake Protocol: Client Hello

Con el host 150.214.40.78 se utiliza la versión TLS 1.2

tls.handshake.type==1						
No.	Time	Source	Destination	Protocol	Length	Info
437	6.346161	192.168.48.76	150.214.40.78	TLSv1.2	284	Client Hello
442	6.349444	192.168.48.76	150.214.108.1	TLSv1	288	Client Hello
483	6.390796	192.168.48.76	108.177.15.108	TLSv1.2	254	Client Hello
564	6.532210	192.168.48.76	150.214.40.78	TLSv1.2	284	Client Hello
1664	23.287243	192.168.48.76	104.18.27.218	TLSv1.3	619	Client Hello
1759	23.813612	192.168.48.76	52.19.92.22	TLSv1.2	583	Client Hello
1773	23.831485	192.168.48.76	18.200.155.5	TLSv1.2	594	Client Hello
3617	73.442740	192.168.48.76	20.189.173.14	TLSv1.2	583	Client Hello
3794	75.807938	192.168.48.76	17.250.80.181	TLSv1.2	583	Client Hello
4458	91.436578	192.168.48.76	31.13.83.51	TLSv1.3	583	Client Hello

> Frame 437: 284 bytes on wire (2272 bits), 284 bytes captured (2272 bits) on interface en3, id 0

> Ethernet II, Src: Dongguan_02:61:70 (34:29:8f:72:61:70), Dst: VMware_97:07:ba (00:50:56:97:07:ba)

> Internet Protocol Version 4, Src: 192.168.48.76, Dst: 150.214.40.78

> Transmission Control Protocol, Src Port: 61282, Dst Port: 993, Seq: 1, Ack: 1, Len: 218

▼ Transport Layer Security

- ▼ TLSv1.2 Record Layer: Handshake Protocol: Client Hello
 - Content Type: Handshake (22)
 - Version: TLS 1.0 (0x0301)
 - Length: 213
 - > Handshake Protocol: Client Hello

3. ¿Existe alguna conexión TLS 1.3? En caso afirmativo indica con qué host.

La conexión con el host 104.17.27.218 utiliza la versión TLS 1.3

tls.handshake.type==1						
No.	Time	Source	Destination	Protocol	Length	Info
437	6.346161	192.168.48.76	150.214.40.78	TLSv1.2	284	Client Hello
442	6.349444	192.168.48.76	150.214.108.1	TLSv1	288	Client Hello
483	6.390796	192.168.48.76	108.177.15.108	TLSv1.2	254	Client Hello
564	6.532210	192.168.48.76	150.214.40.78	TLSv1.2	284	Client Hello
1664	23.287243	192.168.48.76	104.18.27.218	TLSv1.3	619	Client Hello
1759	23.813612	192.168.48.76	52.19.92.22	TLSv1.2	583	Client Hello
1773	23.831485	192.168.48.76	18.200.155.5	TLSv1.2	594	Client Hello
3617	73.442740	192.168.48.76	20.189.173.14	TLSv1.2	583	Client Hello
3794	75.807938	192.168.48.76	17.250.80.181	TLSv1.2	583	Client Hello
4458	91.436578	192.168.48.76	31.13.83.51	TLSv1.3	583	Client Hello

> Frame 1664: 619 bytes on wire (4952 bits), 619 bytes captured (4952 bits) on interface en3, id 0

> Ethernet II, Src: Dongguan_02:61:70 (34:29:8f:72:61:70), Dst: VMware_97:07:ba (00:50:56:97:07:ba)

> Internet Protocol Version 4, Src: 192.168.48.76, Dst: 104.18.27.218

> Transmission Control Protocol, Src Port: 61286, Dst Port: 443, Seq: 1, Ack: 1, Len: 553

▼ Transport Layer Security

- ▼ TLSv1.3 Record Layer: Handshake Protocol: Client Hello
 - Content Type: Handshake (22)
 - Version: TLS 1.0 (0x0301)
 - Length: 548
 - > Handshake Protocol: Client Hello

La conexión con el host 31.13.83.51 usa versión TLS 1.3

tls.handshake.type==1						
No.	Time	Source	Destination	Protocol	Length	Info
437	6.346161	192.168.48.76	150.214.40.78	TLSv1.2	284	Client Hello
442	6.349444	192.168.48.76	150.214.108.1	TLSv1	288	Client Hello
483	6.390796	192.168.48.76	108.177.15.108	TLSv1.2	254	Client Hello
564	6.532210	192.168.48.76	150.214.40.78	TLSv1.2	284	Client Hello
1664	23.287243	192.168.48.76	104.18.27.218	TLSv1.3	619	Client Hello
1759	23.813612	192.168.48.76	52.19.92.22	TLSv1.2	583	Client Hello
1773	23.831485	192.168.48.76	18.200.155.5	TLSv1.2	594	Client Hello
3617	73.442740	192.168.48.76	20.189.173.14	TLSv1.2	583	Client Hello
3794	75.807938	192.168.48.76	17.250.80.181	TLSv1.2	583	Client Hello
4458	91.436578	192.168.48.76	31.13.83.51	TLSv1.3	583	Client Hello

> Frame 4458: 583 bytes on wire (4664 bits), 583 bytes captured (4664 bits) on interface en3, id 0

> Ethernet II, Src: Dongguan_02:61:70 (34:29:8f:72:61:70), Dst: VMware_97:07:ba (00:50:56:97:07:ba)

> Internet Protocol Version 4, Src: 192.168.48.76, Dst: 31.13.83.51

> Transmission Control Protocol, Src Port: 61291, Dst Port: 443, Seq: 1, Ack: 1, Len: 517

▼ Transport Layer Security

- ▼ TLSv1.3 Record Layer: Handshake Protocol: Client Hello
 - Content Type: Handshake (22)
 - Version: TLS 1.0 (0x0301)
 - Length: 512
 - > Handshake Protocol: Client Hello

4. En la conexión con el servidor 20.189.173.14, ¿qué suites de cifrado se ofertan al servidor? ¿cuál es la elegida para establecer la conexión?

Utilizo el filtro `tls.handshake.type == 2` para ver conexiones del tipo `tls` que contienen un mensaje del tipo “Server Hello”.

Se ofertan 17 distintos suites de cifrado. El que utiliza finalmente el servidor es el 11º en la lista, es decir, `TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384`.

tls.handshake.type==1					
No.	Time	Source	Destination	Protocol	Length Info
437	6.346161	192.168.48.76	150.214.40.78	TLSv1.2	284 Client Hello
442	6.349444	192.168.48.76	150.214.108.1	TLSv1	288 Client Hello
483	6.390796	192.168.48.76	108.177.15.108	TLSv1.2	254 Client Hello
564	6.532210	192.168.48.76	150.214.40.78	TLSv1.2	284 Client Hello
1664	23.287243	192.168.48.76	104.18.27.218	TLSv1.3	619 Client Hello
1759	23.813612	192.168.48.76	52.19.92.22	TLSv1.2	583 Client Hello
1773	23.831485	192.168.48.76	18.200.155.5	TLSv1.2	594 Client Hello
3617	73.442740	192.168.48.76	20.189.173.14	TLSv1.2	583 Client Hello
3794	75.807938	192.168.48.76	17.250.80.181	TLSv1.2	583 Client Hello
4458	91.436578	192.168.48.76	31.13.83.51	TLSv1.3	583 Client Hello

✓ Cipher Suites (17 suites)

Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
Cipher Suite: TLS_CHACHA20_POLY1305_SHA256 (0x1303)
Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (0xc024)
Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (0xc023)
Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
Cipher Suite: TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca9)
Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028)
Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xc027)
Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
Cipher Suite: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca8)

Compression Methods Length: 1

tls.handshake.type==2 && ip.src == 20.189.173.14					
No.	Time	Source	Destination	Protocol	Length Info
3644	73.811353	20.189.173.14	192.168.48.76	TLSv1.2	566 Server Hello, Certificate, Certificate Status, Serve...

> [5 Reassembled TCP Segments (6292 bytes): #3640(1448), #3641(1448), #3642(1448), #3643(1448), #3644(500)]

✓ Transport Layer Security

✓ TLSv1.2 Record Layer: Handshake Protocol: Multiple Handshake Messages

Content Type: Handshake (22)

Version: TLS 1.2 (0x0303)

Length: 6287

✓ Handshake Protocol: Server Hello

Handshake Type: Server Hello (2)

Length: 85

Version: TLS 1.2 (0x0303)

> Random: 639c6028cb3e24805f7b392e71aea515940238423d51ba6bc0d45a4ea7546f57

Session ID Length: 32

Session ID: 910a0000bbbe85e1ffc35c44feb57fb1c66b03e1352b18af2933ccbd116b4de

Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)

Compression Method: null (0)

Extensions Length: 13

> Extension: status_request (len=0)

> Extension: extended_master_secret (len=0)

> Extension: renegotiation_info (len=1)

5. ¿Cuál es la clave pública del servidor 20.189.173.14? ¿De qué tipo de clave se trata (i.e., para qué algoritmo se utiliza)?

Se utiliza Diffie Helman efímero (ECDH) con algoritmo de cifrado RSA y SHA (Secure Hash Algorithm) 256 (visto en el apartado anterior)

No.	Time	Source	Destination	Protocol	Length	Info
3644	73.811353	20.189.173.14	192.168.48.76	TLSv1.2	566	Server Hello, Certificate, Certificate Status, Serve...

Handshake Type: Certificate Status (22)
Length: 1781
Certificate Status Type: OCSPP (1)
OCSPP Response Length: 1777
> OCSPP Response
Handshake Protocol: Server Key Exchange
Handshake Type: Server Key Exchange (12)
Length: 361
EC Diffie-Hellman Server Params
Curve Type: named_curve (0x03)
Named Curve: secp384r1 (0x0018)
Pubkey Length: 97
Pubkey: 04b603efcfbf9f30b4f4e958a2b4049b676058ef6b62f3379d8d655d02274fc784e1a8a4...
> Signature Algorithm: rsa_pkcs1_sha256 (0x0401)
Signature Length: 256
Signature: 3bfa803ab8abb43cb62824d797b46ad90c3200618fb1082dbfcadb81fb07f5effd7b16bd...
Handshake Protocol: Server Hello Done
Handshake Type: Server Hello Done (14)
Length: 0

Ejercicio 2

```
#!/bin/sh
```

```
# -----
```

```
# IPTABLES script
```

```
# -----
```

#1) Eliminan todas las reglas y contadores de las tablas de IPTABLES y las cadenas de la tabla de NAT. Esto permite resetear las configuraciones de IPTABLES y de NAT.

```
iptables -F
```

```
iptables -X
```

```
iptables -Z
```

```
iptables -t nat -F
```

#2) Establecen las políticas predeterminadas de las tablas IPTRABLES a "DROP" (denegar) para todos los paquetes entrantes, salientes, de enrutamiento y todas las reglas NAT de preruteo y postruteo. Serán denegados por defecto a menos que cumplan una regla específica que permita el tráfico.

```
iptables -P INPUT DROP
```

```
iptables -P OUTPUT DROP
```

```
iptables -P FORWARD DROP
```

```
iptables -t nat -P PREROUTING DROP
```

```
iptables -t nat -P POSTROUTING DROP
```

#3) Establecen reglas de NAT de preruteo que redirigen el tráfico (de paquetes TCP) entrante a través de la interfaz eth0 (que conecta a internet) en los puertos 80 y 443 al host con dirección IP 192.168.3.2

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j DNAT --to 192.168.3.2:80
```

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 443 -j DNAT --to 192.168.3.2:443
```

#4) Establece (agrega) una regla al final de la tabla INPUT que permite el tráfico entrante desde la red 192.168.10.0 a través de la interfaz eth1. (Permite que los hosts de la red 192.168.10.0 accedan a la máquina a través de la interfaz eth1)

```
iptables -A INPUT -s 192.168.10.0/24 -i eth1 -j ACCEPT
```


#5) Establecen reglas de NAT de postrueto que permiten que los hosts de las redes 192.168.10.0 y 192.168.3.0 accedan a Internet utilizando la dirección IP pública compartida de la interfaz eth0.

```
iptables -t nat -A POSTROUTING -s 192.168.10.0/24 -o eth0 -j MASQUERADE
```

```
iptables -t nat -A POSTROUTING -s 192.168.3.0/24 -o eth0 -j MASQUERADE
```

#6) Activa el enrutamiento de paquetes IP en el kernel de linux. Escribe el valor "1" en el archivo ip_forward para activar el enrutamiento de paquetes IP. Esto permite que los hosts de la red interna accedan a Internet a través de la interfaz eth0.

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

#7) Establecen reglas de filtrado de paquetes en la tabla FORWARD que permiten el tráfico de paquetes entre las redes conectadas a las interfaces eth1 y eth2. Son reglas para permitir el tráfico de paquetes en estados NEW, ESTABLISHED o RELATED.

```
iptables -t filter -A FORWARD -i eth1 -o eth2 -m state --state  
NEW,ESTABLISHED,RELATED -j ACCEPT
```

```
iptables -t filter -A FORWARD -i eth2 -o eth1 -m state --state ESTABLISHED,RELATED -j  
ACCEPT
```

1. ¿Qué servicios pueden ser accedidos desde Internet?

Pueden ser accedidos los servicios HTTP y HTTPS (puertos 80 y 443, respectivamente) Se establece en los comandos:

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j DNAT --to 192.168.3.2:80
```

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 443 -j DNAT --to 192.168.3.2:443
```

2. ¿Existe una zona desmilitarizada (DMZ) en la red?

No. No se puede determinar si existe una DMZ en la red basándonos únicamente en las reglas de filtrado y NAT establecidas.

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j DNAT --to 192.168.3.2:80
```

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 443 -j DNAT --to 192.168.3.2:443
```

Las reglas agregadas con estos comandos utilizan NAT de destino para redirigir el tráfico entrante a los servicios de la IP 192.168.3.2.

Sin embargo no proporcionan información para determinar si existe una DMZ.

Para ver una posible DMZ deberíamos fijarnos si se establecen reglas NAT de preruteo o postruteo o reglas de filtrado que permiten el acceso a servicios específicos desde Internet mientras que el tráfico entrante y saliente a Internet es filtrado o bloqueado.

3. ¿Los hosts de la red 192.168.10.0/24 pueden acceder a Internet?

Sí, los hosts de la red 192.168.10.0/24 pueden acceder a Internet; siempre y cuando cumplan con las reglas de filtrado y NAT que se establecen en el script. Una de las condiciones es que el tráfico saliente hacia Internet esté permitido. Esto se establece con el siguiente comando.

```
iptables -t nat -A POSTROUTING -s 192.168.10.0/24 -o eth0 -j MASQUERADE
```

Permite que los hosts de la red accedan a Internet.

4. ¿Pueden los hosts de la red 192.168.10.0/24 recibir conexiones desde Internet?

Sí, siempre y cuando cumplan con las reglas de filtrado y NAT establecidas.

Una de las condiciones es que el tráfico entrante desde Internet hacia los servicios disponibles en la red esté permitido. Esto se establece con el siguiente comando.

```
iptables -A INPUT -s 192.168.10.0/24 -i eth1 -j ACCEPT
```

Permite que los hosts de la red reciban conexiones desde Internet

5. ¿Pueden los hosts de la red 192.168.10.0/24 recibir conexiones desde la red 192.168.3.0/24?

Sí, siempre y cuando cumplan con reglas de filtrado y NAT establecidas.

La condición es que el tráfico entre ambas redes esté permitido.

```
iptables -t filter -A FORWARD -i eth1 -o eth2 -m state --state  
NEW,ESTABLISHED,RELATED -j ACCEPT
```

```
iptables -t filter -A FORWARD -i eth2 -o eth1 -m state --state ESTABLISHED,RELATED -j  
ACCEPT
```