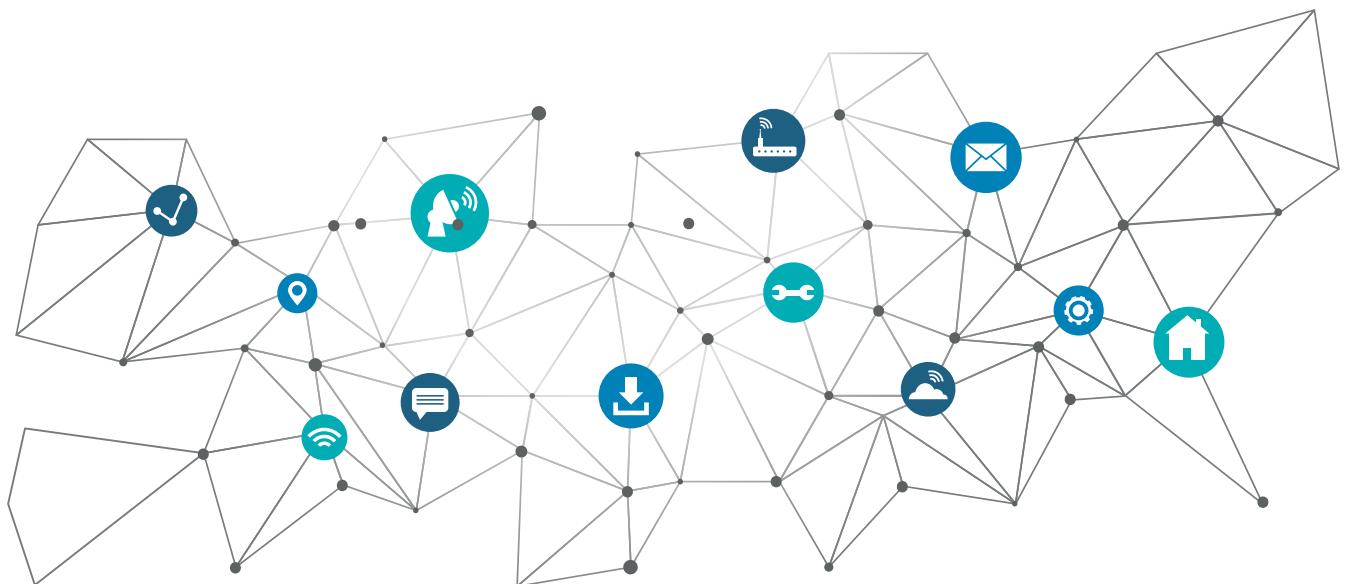


# Trabajo Práctico Especial

Configuración y Desarrollo de Aplicaciones en Redes



Grupo 26

Alumnos: Cesario Provenzano, Guillermo Tomás

Murillo, Francisco

Mails: [gcesarioprovenzano@alumnos.exa.unicen.edu.ar](mailto:gcesarioprovenzano@alumnos.exa.unicen.edu.ar)

[fmurillo@alumnos.exa.unicen.edu.ar](mailto:fmurillo@alumnos.exa.unicen.edu.ar)

Ayudante: Germán Imeroni

# Indice

Introducción.....	3
Ejercicio 1.....	4
Ejercicio 2.....	5
Ejercicio 3.....	6
Ejercicio 4.....	7
Ejercicio 5.....	8
Ejercicio 6.....	9
Ejercicio 7.....	10
Ejercicio 8.....	11
Ejercicio 9.....	12
Ejercicio 10.....	13
Ejercicio 11.....	14
Ejercicio 12.....	16
Ejercicio 13.....	19
Ejercicio 14.....	21
Conclusión.....	23

# Introducción

El objetivo del trabajo fue la configuración de redes y subredes de un Parque Industrial. En estos graficos se puede ver como se pensó el diagrama y se planificó la cantidad de IPs necesarias.

## DATACENTER

120 SERVIDORES + 3 ROUTER + BASE + BROADCAST = 125

$$2^7 = 128$$

## TRONCAL

300 IP + 4 ROUTER + BASE + BROADCAST = 306

$$2^9 = 512$$

## FABRICA A

S1(SW3) 1 IP + 45 DISPOSITIVOS + BASE + BROADCAST = 48

S2(R WIFI) 1 IP + 80 DISPOSITIVOS + BASE + BROADCAST = 83

S3(SW4) 10 SERVIDORES + BASE + BROADCAST = 12

S4(R6-R7) 2 + BASE + BROADCAST = 4

**TOTAL = 64(48) + 128(83) + 16(12) + 4(4) = 212**

$$2^8 = 256$$

## FABRICA B

S1(HUB) 1 IP + 8 INTERFACES + BASE + BROADCAST = 11

S2(SW5) 1 IP + 65 SERVIDORES + BASE + BROADCAST = 68

S3(SW6) 1 IP + 24 EQUIPOS + BASE + BROADCAST = 27

**TOTAL = 16(11) + 128(68) + 32(27) = 176**

$$2^8 = 256$$

## TOTAL

DATACENTER + TRONCAL + FABRICA A + FABRICA B

$$2^{11} = 2048$$

**TOTAL = 128 + 512 + 256 + 256 = 1152**

# Ejercicio 1

Para la cantidad de conexiones proyectadas para cada una de las redes, realice una asignación de direcciones IP, creando un VLSM general para el parque industrial, y uno particular para cada una de las fábricas. Consideré que las direcciones privadas se encuentren en la red 172.X.0.0/20, donde X es el número de grupo que se les asignó.

VLSM General

172.26.0.0		172.26.4.0	
Troncal		Datacenter	
172.26.1.255		172.26.4.127	
172.26.2.0	172.26.3.0		
Fabrica A	Fabrica B		
172.26.2.255	172.26.3.255		172.26.7.255

VLSM Fabrica A

172.26.2.0	172.26.2.128	
	SW3	
	172.26.2.191	
	172.26.2.192	
	SW4	
	172.26.2.207	
	172.26.2.208	
	R6-R7	
	172.26.2.211	
		172.26.2.255
172.26.2.127		

VLSM Fabrica B

172.26.3.0	172.26.3.128	
	SW6	
	172.26.3.159	
	172.26.3.160	
	HUB	
	172.26.3.175	
		172.26.3.255
172.26.3.127		

# Ejercicio 2

Realice una tabla en donde se indiquen cada una de las subredes resultantes, indicando el nombre de cada red, su dirección base, la máscara, y el rango de ip asignables que incluye cada bloque.

## Tabla General

Numero	Nombre	Dirección de Red	Prefijo de Mascara	Mascara de Subred Decimal	Primera IP Disponible	Ultima IP Disponible	IP Broadcast
1	Troncal	172.26.0.0	/23	255.255.255.254.0	172.26.0.1	172.26.1.254	172.26.1.255
2	Fabrica A	172.26.2.0	/24	255.255.255.128	172.26.2.1	172.26.2.254	172.26.2.255
3	Fabrica B	172.26.3.0	/24	255.255.255.0	172.26.3.1	172.26.3.254	172.26.3.255
4	Datacenter	172.26.4.0	/25	255.255.255.128	172.26.4.1	172.26.4.254	172.26.4.127

## Tabla Fabrica A

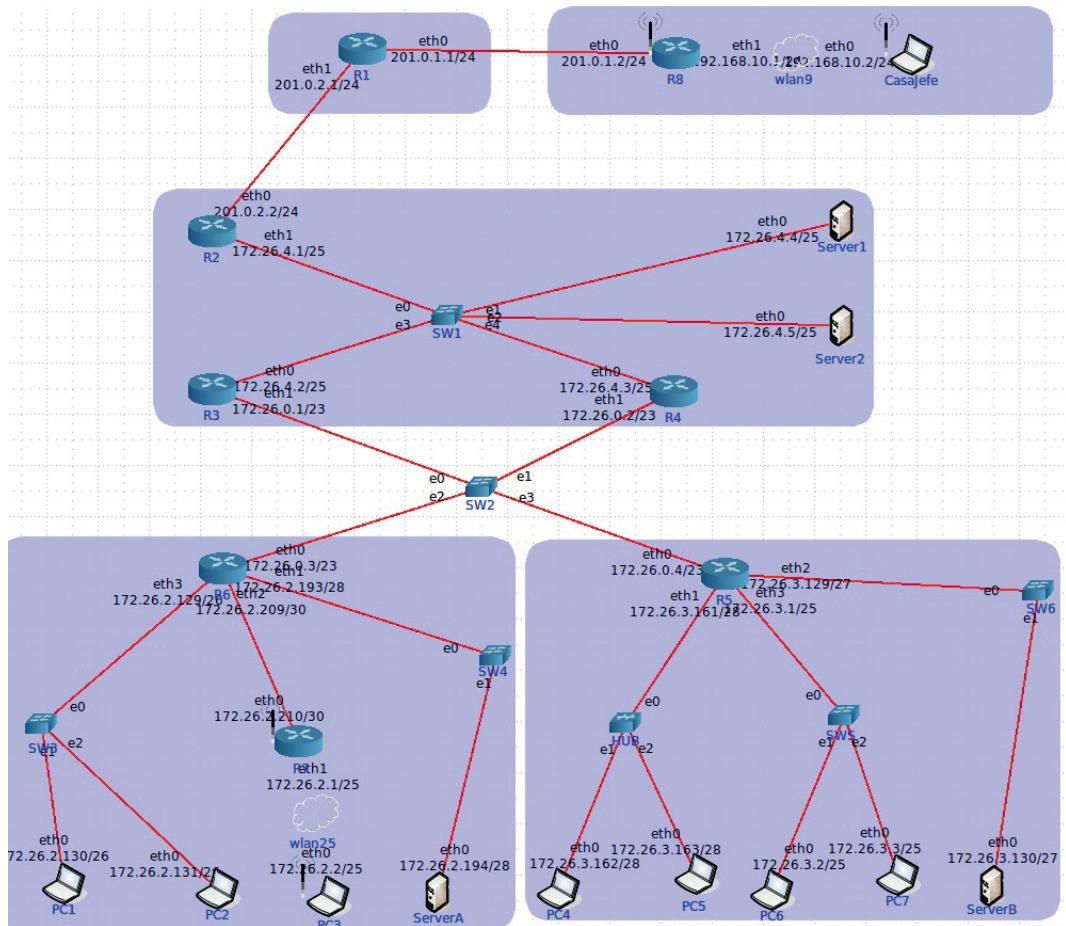
Numero	Nombre	Dirección de Red	Prefijo de Mascara	Mascara de Subred Decimal	Primera IP Disponible	Ultima IP Disponible	IP Broadcast
1	RWIFI	172.26.2.0	/25	255.255.255.0	172.26.2.1	172.26.2.126	172.26.2.127
2	SW3	172.26.2.128	/26	255.255.255.192	172.26.2.129	172.26.2.190	172.26.2.191
3	SW4	172.26.2.192	/28	255.255.255.240	172.26.2.193	172.26.2.206	172.26.2.207
4	R6-R7	172.26.2.208	/30	255.255.255.252	172.26.2.209	172.26.2.210	172.26.2.211

## Tabla Fabrica B

Numero	Nombre	Dirección de Red	Prefijo de Mascara	Mascara de Subred Decimal	Primera IP Disponible	Ultima IP Disponible	IP Broadcast
1	SW5	172.26.3.0	/25	255.255.255.128	172.26.3.1	172.26.3.126	172.26.3.127
2	SW6	172.26.3.128	/27	255.255.255.224	172.26.3.129	172.26.3.158	172.26.3.159
3	HUB	172.26.3.160	/28	255.255.255.240	172.26.3.161	172.26.3.174	172.26.3.175

# Ejercicio 3

Implemente la red propuesta en el emulador CORE con la disposición de equipos que actualmente se tienen conectados. Considere la asignación IP realizada en el ejercicio 1, y la colocación de direcciones públicas en donde corresponda.



UserDefined on node R6 (n19)

UserDefined service

Meta-data  Customize this service to do anything upon startup.

[Files](#) [Directories](#) [Startup/shutdown](#)

Startup index:

Start time:  (seconds after runtime; leave empty for default)

Startup Commands

```
ifconfig eth0 172.26.0.3/23
ifconfig eth1 172.26.2.193/28
ifconfig eth2 172.26.2.209/30
ifconfig eth3 172.26.2.129/26
```

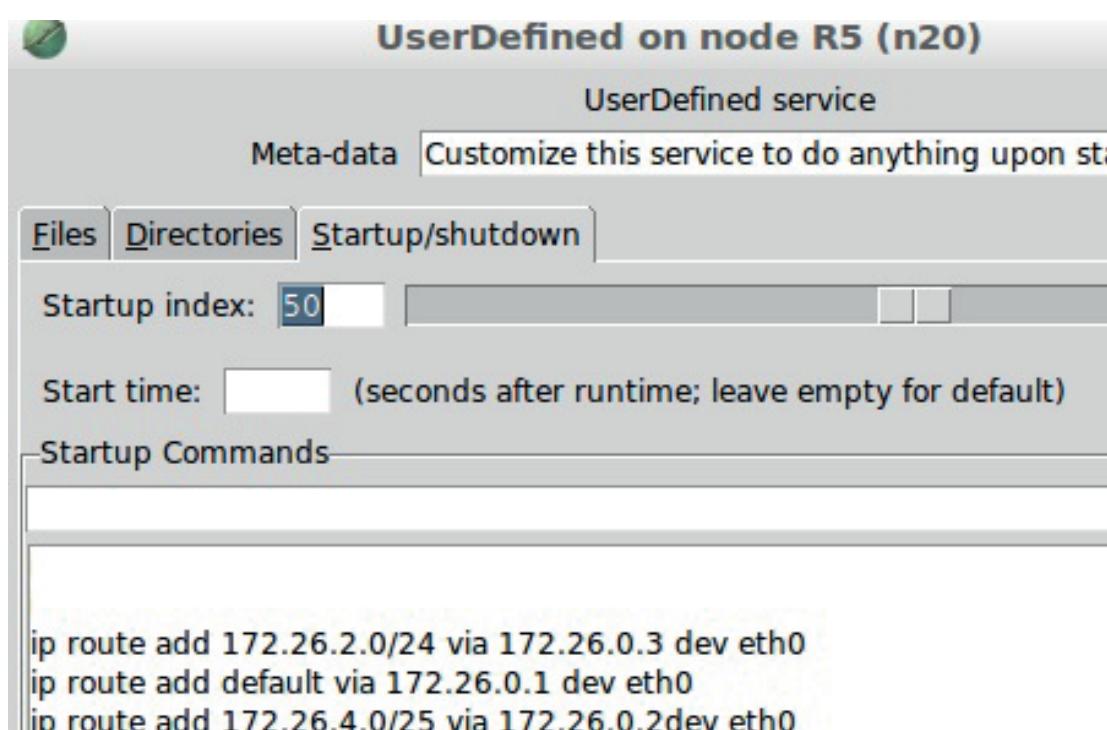
Ejemplo de uso del comando `ifconfig <interfaz> <ip>` en el router 6

# Ejercicio 4

Configurar todas las interfaces y rutas de los routers y hosts, minimizando la cantidad de entradas en las tablas de ruteo (considere el uso de rutas por defecto) para que todas las redes del parque industrial estén interconectadas.

Cabe destacar que los comandos correspondientes deben estar cargados en la opción “User Defined” -> “Startup Commands” de cada dispositivo, y que dentro de los servicios sólo deben quedar habilitados el “IP FORWARD” y “User Defined”.

Tenga en cuenta que el router del ISP no lleva ruta por defecto.



Ejemplo de uso del comando en el router 5

```
ip route add <red destino> via <puerta de enlace> dev <interfaz de salida>
```

# Ejercicio 5

Configurar el R2 para que solo tengan acceso a internet los equipos conectados a la red Datacenter, Fábrica A y Fabrica B.

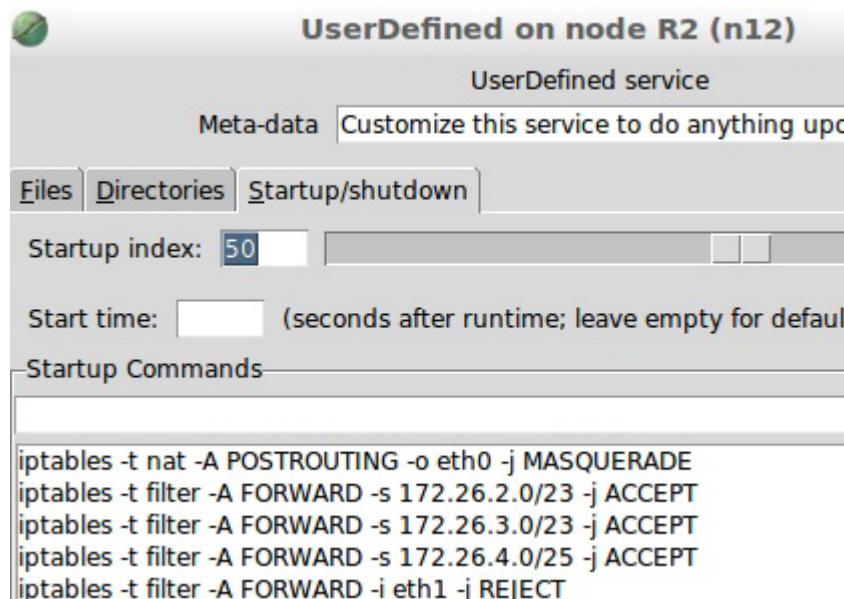


Imagen de los comandos usados en el router 2

***iptables -t nat -A POSTROUTING -o <interfaz> -j MASQUERADE***

Comando para enmascarar las ip privadas con la ip pública del router frontera

***iptables -t filter -A FORWARD -s <red> ACCEPT***

Comando para que el router acepte los paquetes para las redes indicadas

***iptables -t filter -A FORWARD -i <interfaz> -j REJECT***

Comando para rechazar los paquetes

# Ejercicio 6

Configurar el R5 para que todos los dispositivos del área de servicio conectados a través del HUB solo se puedan conectar al resto de los equipos de la Fábrica B y también a los equipos del centro de cómputo conectados en el Datacenter.

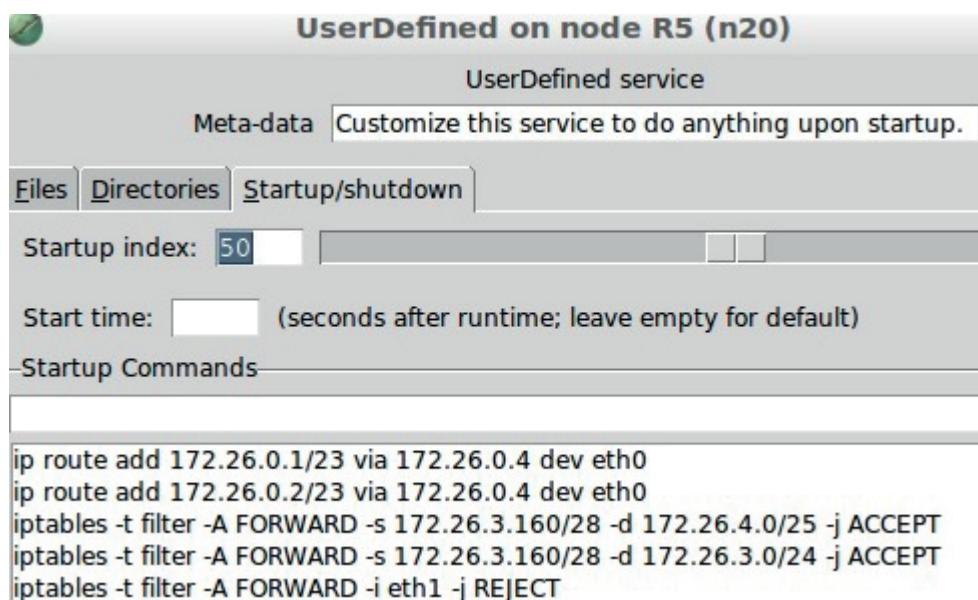


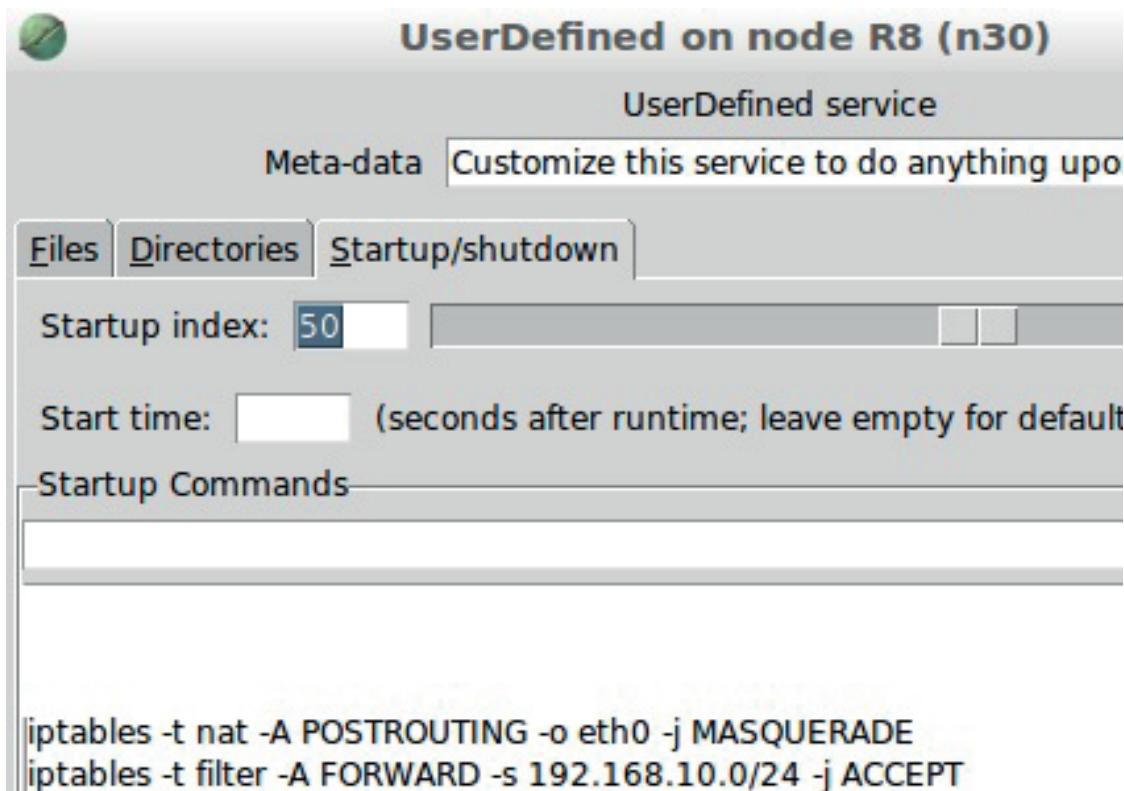
Imagen de los comandos usados en el router 5

***iptables -t filter -A FORWARD -s <origen> -d <destino> -j ACCEPT***

Comando para enmascarar las ip privadas con la ip pública del router frontera

# Ejercicio 7

Configurar el R8 para que la PC-Casa pueda acceder a internet.



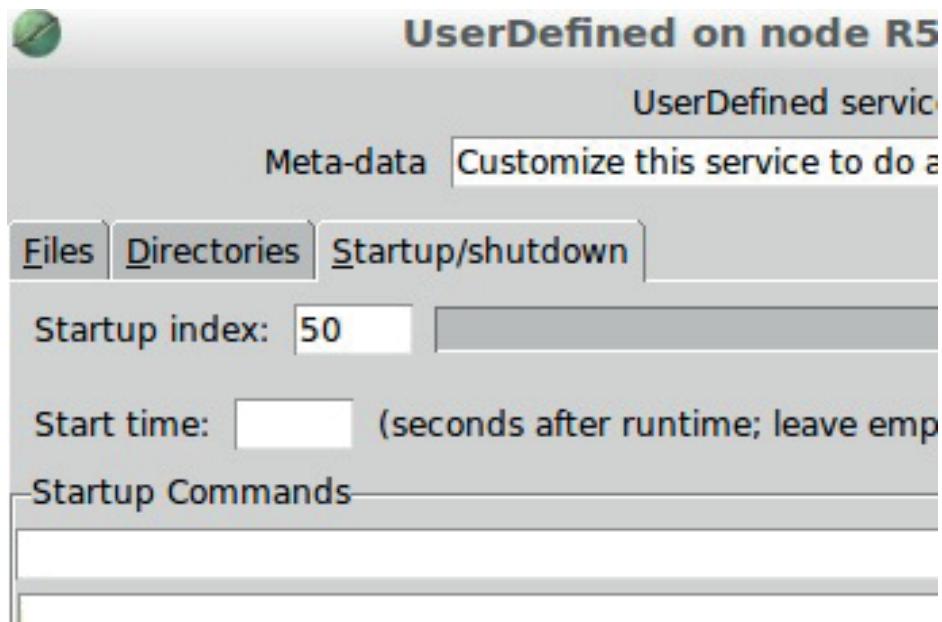
Para la resolución de este ejercicio se tuvo que recurrir a los mismos comandos que en el ejercicio 5, se configuró el router 8 posibilitando el acceso a internet de la PC-Casa del Jefe.

```
iptables -t nat -A POSTROUTING -o <interfaz> -j MASQUERADE
```

```
iptables -t filter -A FORWARD -s <red> ACCEPT
```

# Ejercicio 8

Por cuestiones de seguridad, todo el tráfico proveniente de las fábricas y dirigido al datacenter debe pasar por el router R4, y el tráfico dirigido a Internet, debe pasar por el router R3.



UserDefined on node R5

UserDefined service

Meta-data Customize this service to do anything up to

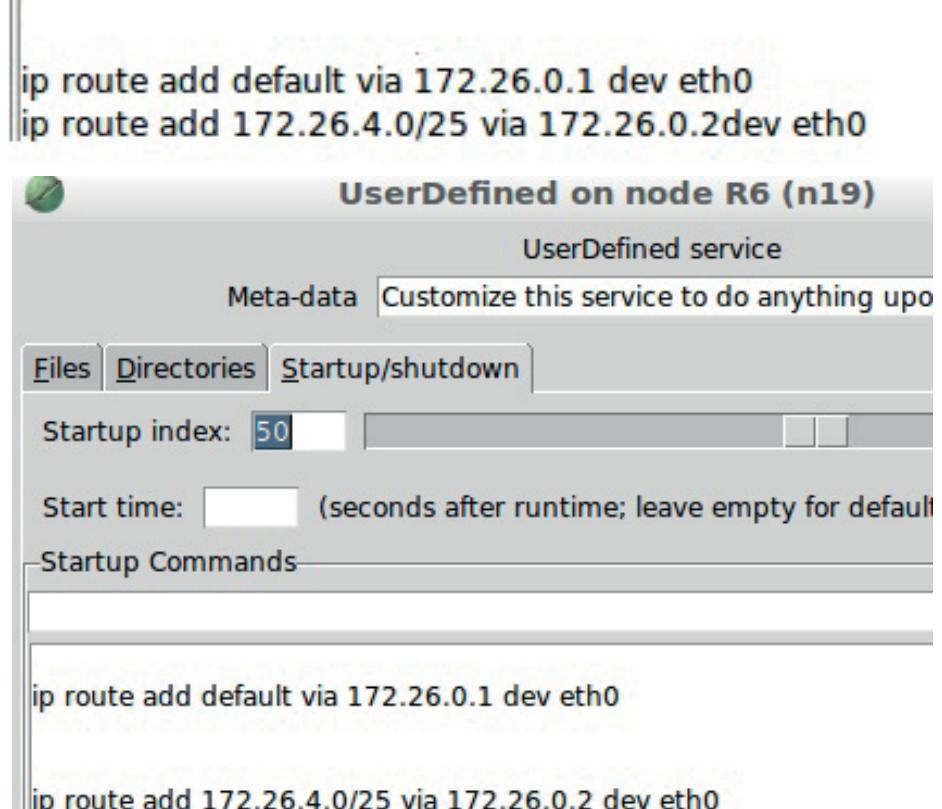
Files Directories Startup/shutdown

Startup index: 50

Start time: (seconds after runtime; leave empty for default)

Startup Commands

```
ip route add default via 172.26.0.1 dev eth0
ip route add 172.26.4.0/25 via 172.26.0.2 dev eth0
```

UserDefined on node R6 (n19)

UserDefined service

Meta-data Customize this service to do anything up to

Files Directories Startup/shutdown

Startup index: 50

Start time: (seconds after runtime; leave empty for default)

Startup Commands

```
ip route add default via 172.26.0.1 dev eth0

ip route add 172.26.4.0/25 via 172.26.0.2 dev eth0
```

Para ejercer control sobre el tráfico se recurrió a utilizar los “default”. Las imágenes muestran los routers (5 y 6) provenientes de las fábricas (A y B).

# Ejercicio 9

Configure la red de manera de poder enviar el mensaje “Hola Data Center” desde PCCasa hasta Server 1, utilizando Netcat. Tenga en cuenta configurar el reenvío de paquetes en Router 2. Considere que el puerto que está abierto en el Router 2 es el 80, mientras que el servicio en el Server 1 está corriendo en el puerto 8080.

Luego, replique la conexión desde la PC1. Indique para ambos casos, ¿qué dirección IP y qué puertos se deben utilizar?

## Resolución:

Para que puedan reenviarse los paquetes entrantes con el puerto 80 (Servidor 1) y 90 (PC1) usando el puerto 8080 se configuraron los siguientes comandos:

```
iptables -t nat -A PREROUTING -p tcp --dport 80 -j DNAT --to 172.26.4.4:8080
```

En la tabla de ruteo del router 2

```
iptables -t nat -A PREROUTING -p tcp --dport 90 -j DNAT --to 172.26.4.4:8080
```

En la tabla de ruteo del router 6

### En consola:

Primero se uso el comando **nc -4 -l 8080** para dejar escuchando el puerto 8080.

Luego el comando **nc -4 -t 201.0.2.2 80** desde la casa del jefe para conectarse al router frontera.

Se replicaron los mismos comandos desde PC1 (puerto 80 por 90)

The image shows two terminal windows side-by-side. Both are titled "LXTerminal". The left window shows the command "root@Server1:/tmp/pycore.45222/Server1.conf# nc -4 -l 8080" followed by the text "Hola Datacenter". The right window shows the command "root@CasaJefe:/tmp/pycore.45222/CasaJefe.conf# nc -4 -t 201.0.2.2 80" followed by the text "Hola Datacenter". This represents the initial connection from the server to the router.

Imagen del mensaje entre Server 1 y Casa Jefe

The image shows two terminal windows side-by-side. Both are titled "LXTerminal". The left window shows the command "root@Server1:/tmp/pycore.38013/Server1.conf# nc -4 -l 8080" followed by the text "Hola Datacenter". The right window shows the command "root@PC1:/tmp/pycore.38013/PC1.conf# nc -4 -t 172.26.0.3 90" followed by the text "Hola Datacenter". This represents the replicated connection from the PC to the server.

Imagen del mensaje entre PC1 y Server 1

# Ejercicio 10

Realice un Ping desde la PC4 a la PC6. Analice las diferencias en el funcionamiento entre el dispositivo de Hub y el Switch. Justificar con la captura de las pantallas de Wireshark correspondientes.

The image shows two Wireshark captures side-by-side. Both are titled "Capturing from n6.eth0.89 [Wireshark 1.6.7]" and "Capturing from n3.eth0.89 [Wireshark 1.6.7]". The left capture (n6.eth0.89) shows 32 ICMP echo requests (req=1 to req=15) sent from 172.26.3.2 to 172.26.3.162. The right capture (n3.eth0.89) shows 34 ICMP echo requests (req=1 to req=17) sent from 172.26.3.2 to 172.26.3.162. Both captures show the same sequence of ICMP requests and responses.

No.	Time	Source	Destination	Protocol	Length
14	5.0004342	172.26.3.2	172.26.3.162	ICMP	64
15	6.0007578	172.26.3.162	172.26.3.2	ICMP	64
16	6.0007615	172.26.3.2	172.26.3.162	ICMP	64
17	7.0006577	172.26.3.162	172.26.3.2	ICMP	64
18	7.0006614	172.26.3.2	172.26.3.162	ICMP	64
19	8.0006614	172.26.3.162	172.26.3.2	ICMP	64
20	8.0006651	172.26.3.2	172.26.3.162	ICMP	64
21	9.0006595	172.26.3.162	172.26.3.2	ICMP	64
22	9.0006638	172.26.3.2	172.26.3.162	ICMP	64
23	10.0005596	172.26.3.162	172.26.3.2	ICMP	64
24	10.0005633	172.26.3.2	172.26.3.162	ICMP	64
25	11.0005393	172.26.3.162	172.26.3.2	ICMP	64
26	11.0005433	172.26.3.2	172.26.3.162	ICMP	64
27	12.0005579	172.26.3.162	172.26.3.2	ICMP	64
28	12.0005621	172.26.3.2	172.26.3.162	ICMP	64
29	13.0007180	172.26.3.162	172.26.3.2	ICMP	64
30	13.0007222	172.26.3.2	172.26.3.162	ICMP	64
31	14.0006178	172.26.3.162	172.26.3.2	ICMP	64
32	14.0006215	172.26.3.2	172.26.3.162	ICMP	64
16	5.0004283	172.26.3.2	172.26.3.162	ICMP	64
17	6.0007548	172.26.3.162	172.26.3.2	ICMP	64
18	6.0007556	172.26.3.2	172.26.3.162	ICMP	64
19	7.0006547	172.26.3.162	172.26.3.2	ICMP	64
20	7.0006556	172.26.3.2	172.26.3.162	ICMP	64
21	8.0006584	172.26.3.162	172.26.3.2	ICMP	64
22	8.0006592	172.26.3.2	172.26.3.162	ICMP	64
23	9.0006566	172.26.3.162	172.26.3.2	ICMP	64
24	9.0006580	172.26.3.2	172.26.3.162	ICMP	64
25	10.0005566	172.26.3.162	172.26.3.2	ICMP	64
26	10.0005575	172.26.3.2	172.26.3.162	ICMP	64
27	11.0005363	172.26.3.162	172.26.3.2	ICMP	64
28	11.0005373	172.26.3.2	172.26.3.162	ICMP	64
29	12.0005554	172.26.3.162	172.26.3.2	ICMP	64
30	12.0005562	172.26.3.2	172.26.3.162	ICMP	64
31	13.0007154	172.26.3.162	172.26.3.2	ICMP	64
32	13.0007163	172.26.3.2	172.26.3.162	ICMP	64
33	14.0006148	172.26.3.162	172.26.3.2	ICMP	64
34	14.0006157	172.26.3.2	172.26.3.162	ICMP	64

The image shows an LXTerminal window with the title "LXTerminal". It displays a terminal session where the user is running a ping command from PC4 to PC6. The output shows 15 ICMP echo requests (req=1 to req=15) sent to 172.26.3.2, with their respective times, TTL values, and round-trip times (RTT). The session is still active, indicated by the "still listening" message at the end.

```
root@PC4:/tmp/pycore.44018/PC4.conf# ping 172.26.3.2
PING 172.26.3.2 (172.26.3.2) 56(84) bytes of data.
64 bytes from 172.26.3.2: icmp_req=1 ttl=63 time=0.094 ms
64 bytes from 172.26.3.2: icmp_req=2 ttl=63 time=0.081 ms
64 bytes from 172.26.3.2: icmp_req=3 ttl=63 time=0.106 ms
64 bytes from 172.26.3.2: icmp_req=4 ttl=63 time=0.085 ms
64 bytes from 172.26.3.2: icmp_req=5 ttl=63 time=0.060 ms
64 bytes from 172.26.3.2: icmp_req=6 ttl=63 time=0.058 ms
64 bytes from 172.26.3.2: icmp_req=7 ttl=63 time=0.055 ms
64 bytes from 172.26.3.2: icmp_req=8 ttl=63 time=0.053 ms
64 bytes from 172.26.3.2: icmp_req=9 ttl=63 time=0.055 ms
64 bytes from 172.26.3.2: icmp_req=10 ttl=63 time=0.062 ms
64 bytes from 172.26.3.2: icmp_req=11 ttl=63 time=0.053 ms
64 bytes from 172.26.3.2: icmp_req=12 ttl=63 time=0.058 ms
64 bytes from 172.26.3.2: icmp_req=13 ttl=63 time=0.061 ms
64 bytes from 172.26.3.2: icmp_req=14 ttl=63 time=0.060 ms
64 bytes from 172.26.3.2: icmp_req=15 ttl=63 time=0.053 ms

```

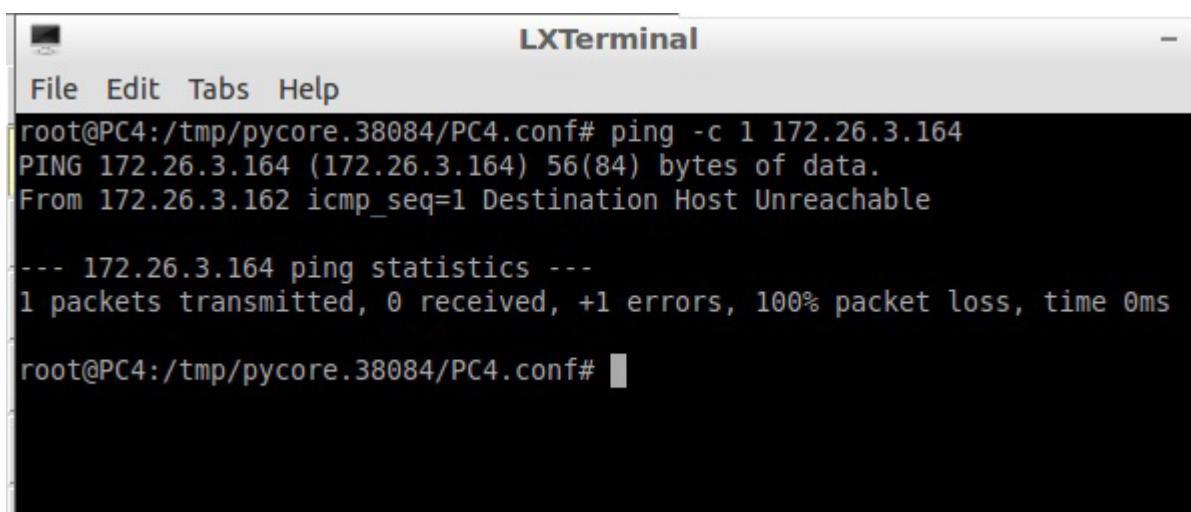
**Hub:** Al trabajar en la capa física regenera la señal de la red y reenvía por todos los puertos por lo que los paquetes llegan aunque no les estén haciendo un “ping”

**Switch:** Al trabajar en la capa de enlace de datos envía el paquete en base al encabezado de este.

# Ejercicio 11

Realice un ping desde la PC4 a una dirección que pertenezca a la misma red pero que no esté conectada. Luego realice un ping desde la PC4 a una dirección que pertenezca a la red Datacenter pero que no esté conectada. Analice en ambos casos los paquetes que se generan (mensajes ICMP y ARP). Utilice la opción -c 1 en ambos ping.

## A) Ping desde PC4 a dirección de la misma red



```
LXTerminal
File Edit Tabs Help
root@PC4:/tmp/pycore.38084/PC4.conf# ping -c 1 172.26.3.164
PING 172.26.3.164 (172.26.3.164) 56(84) bytes of data.
From 172.26.3.162 icmp_seq=1 Destination Host Unreachable
--- 172.26.3.164 ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms
root@PC4:/tmp/pycore.38084/PC4.conf#
```

Imagen del ping en consola

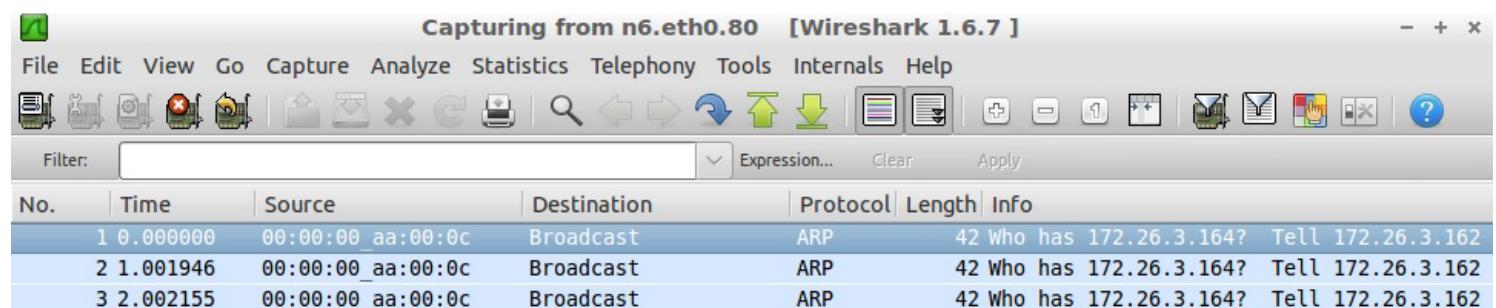


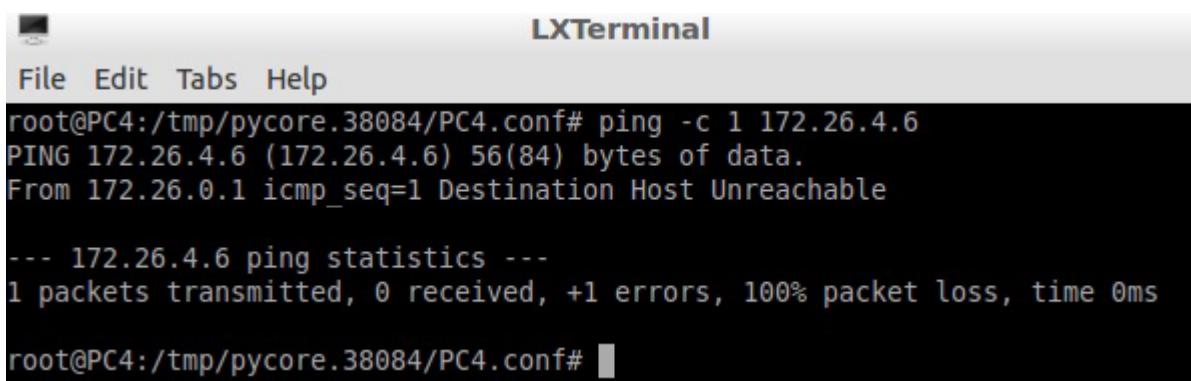
Imagen del Wireshark

## Resultado:

Se intenta hacer el ping. Esto genera mensajes ARP para intentar obtener la dirección MAC del host y no se genera ningún mensaje ICMP ya que no hay coincidencia.

# Ejercicio 11

## B) Ping desde PC4 a dirección del Datacenter

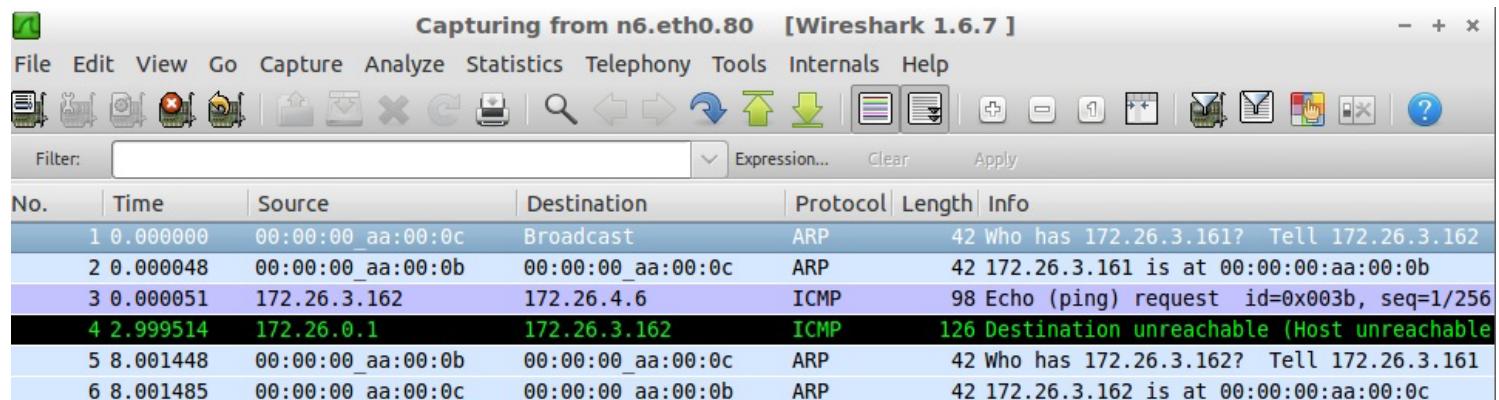


```
LXTerminal
File Edit Tabs Help
root@PC4:/tmp/pycore.38084/PC4.conf# ping -c 1 172.26.4.6
PING 172.26.4.6 (172.26.4.6) 56(84) bytes of data.
From 172.26.0.1 icmp_seq=1 Destination Host Unreachable

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

root@PC4:/tmp/pycore.38084/PC4.conf#
```

Imagen del ping en consola



Capturing from n6.eth0.80 [Wireshark 1.6.7]						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	00:00:00_aa:00:0c	Broadcast	ARP	42	Who has 172.26.3.161? Tell 172.26.3.162
2	0.000048	00:00:00_aa:00:0b	00:00:00_aa:00:0c	ARP	42	172.26.3.161 is at 00:00:00_aa:00:0b
3	0.000051	172.26.3.162	172.26.4.6	ICMP	98	Echo (ping) request id=0x003b, seq=1/256
4	2.999514	172.26.0.1	172.26.3.162	ICMP	126	Destination unreachable (Host unreachable)
5	8.001448	00:00:00_aa:00:0b	00:00:00_aa:00:0c	ARP	42	Who has 172.26.3.162? Tell 172.26.3.161
6	8.001485	00:00:00_aa:00:0c	00:00:00_aa:00:0b	ARP	42	172.26.3.162 is at 00:00:00_aa:00:0c

Imagen del Wireshark

## Resultado:

Se intenta hacer el ping. Esto genera mensajes ARP para obtener la dirección MAC del host pero en este caso a diferencia del anterior se generan dos mensajes ICMP. Uno con el ping y otro diciendo que el host es inalcanzable.

# Ejercicio 12

Realizar pruebas utilizando el comando ping entre los siguientes puntos y mediante la utilización de Wireshark, analice el camino seguido por los paquetes generados, adjuntando al informe las capturas de pantalla correspondientes:

- Desde un equipo conectado a la Fábrica A hasta la dirección privada del R2
- Desde un equipo conectado a la Fábrica A hasta la dirección pública del R2
- Desde Pc4 a la interfaz pública del R8

## A) Ping desde equipo de Fábrica A hasta dirección privada del R2

```
File Edit Tabs Help
root@PC1:/tmp/pycore.44018/PC1.conf# ping 172.26.4.1
PING 172.26.4.1 (172.26.4.1) 56(84) bytes of data.
64 bytes from 172.26.4.1: icmp_req=1 ttl=62 time=0.283 ms
64 bytes from 172.26.4.1: icmp_req=2 ttl=62 time=0.095 ms
64 bytes from 172.26.4.1: icmp_req=3 ttl=62 time=0.105 ms
64 bytes from 172.26.4.1: icmp_req=4 ttl=62 time=0.103 ms
64 bytes from 172.26.4.1: icmp_req=5 ttl=62 time=0.102 ms
64 bytes from 172.26.4.1: icmp_req=6 ttl=62 time=0.166 ms
64 bytes from 172.26.4.1: icmp_req=7 ttl=62 time=0.104 ms
64 bytes from 172.26.4.1: icmp_req=8 ttl=62 time=0.114 ms
64 bytes from 172.26.4.1: icmp_req=9 ttl=62 time=0.207 ms
64 bytes from 172.26.4.1: icmp_req=10 ttl=62 time=0.115 ms
64 bytes from 172.26.4.1: icmp_req=11 ttl=62 time=0.104 ms
64 bytes from 172.26.4.1: icmp_req=12 ttl=62 time=0.103 ms
64 bytes from 172.26.4.1: icmp_req=13 ttl=62 time=0.109 ms
64 bytes from 172.26.4.1: icmp_req=14 ttl=62 time=0.105 ms
64 bytes from 172.26.4.1: icmp_req=15 ttl=62 time=0.500 ms
64 bytes from 172.26.4.1: icmp_req=16 ttl=62 time=0.100 ms
64 bytes from 172.26.4.1: icmp_req=17 ttl=62 time=0.207 ms
64 bytes from 172.26.4.1: icmp_req=18 ttl=62 time=0.106 ms
64 bytes from 172.26.4.1: icmp_req=19 ttl=62 time=0.104 ms
64 bytes from 172.26.4.1: icmp_req=20 ttl=62 time=0.114 ms
64 bytes from 172.26.4.1: icmp_req=21 ttl=62 time=0.106 ms
64 bytes from 172.26.4.1: icmp_req=22 ttl=62 time=0.093 ms
64 bytes from 172.26.4.1: icmp_req=23 ttl=62 time=0.103 ms
64 bytes from 172.26.4.1: icmp_req=24 ttl=62 time=0.102 ms
```

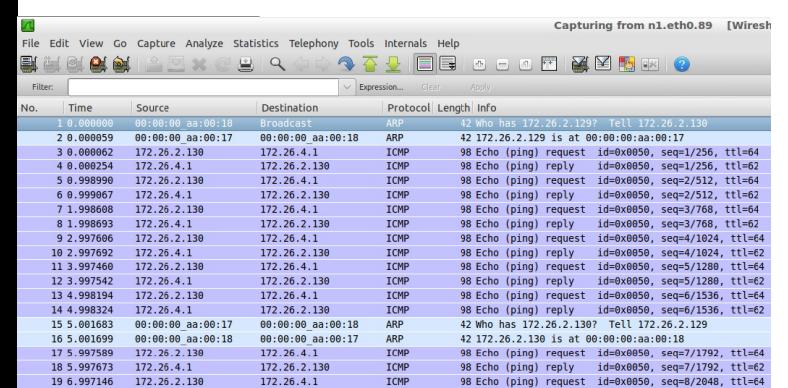


Imagen del Wireshark (Router 6)

Hay paquetes que van y vienen de PC1 y Router 2

## Imagen del ping desde PC1 hacia Router 2 (privada)

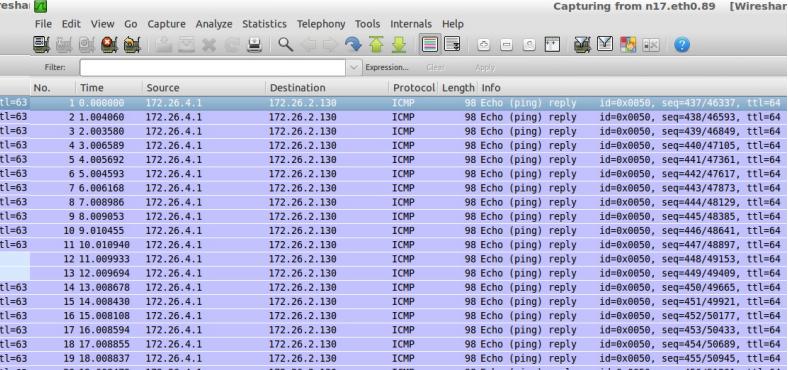


Imagen del Wireshark (Router 4)

Imagen del Wireshark (Router 3)

Los paquetes se dirigen de PC1 al Router 2

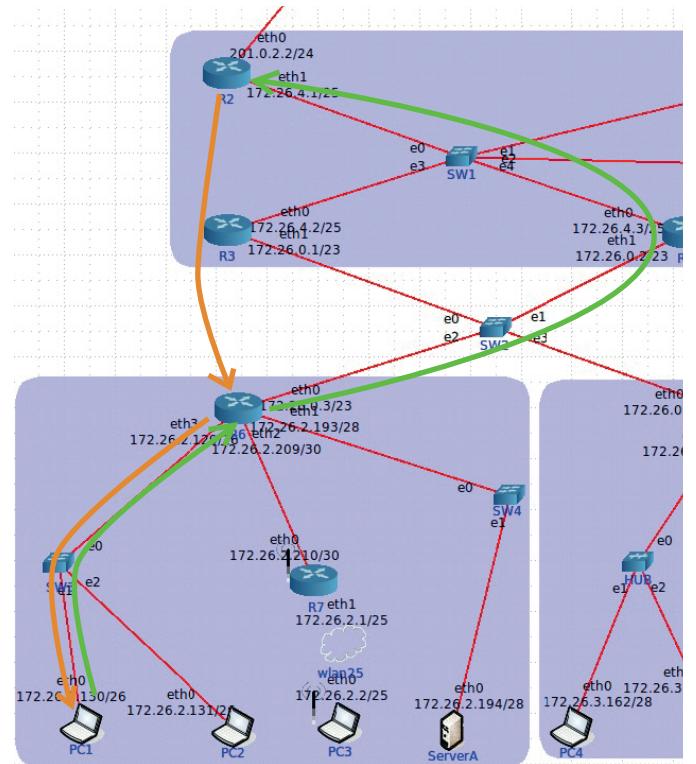
Los paquetes se dirigen de Router 2 a PC1

Esto debido a lo realizado en el ejercicio 8 (el tráfico de las Fabricas al Datacenter siempre subira por el Router 4)

# Ejercicio 12

Capturing from n12.eth1.89 [Wireshark]

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	00:00:00 aa:00:04	Broadcast	ARP	42	42 Who has 172.26.4.1? Tell 172.26.4.3
2	0.000013	00:00:00 aa:00:00	00:00:00 aa:00:04	ARP	42	42 Who has 172.26.4.1 is at 00:00:00:aa:00:00
3	0.000026	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) request id=0x0050, seq=1/256, ttl=64
4	0.000032	00:00:00 aa:00:00	Broadcast	ARP	42	Who has 172.26.4.2? Tell 172.26.4.1
5	0.000045	00:00:00 aa:00:03	00:00:00 aa:00:00	ARP	42	42 172.26.4.2 is at 00:00:00:aa:00:03
6	0.000064	172.26.4.1	172.26.2.130	ICMP	98	Echo (ping) reply id=0x0050, seq=1/256, ttl=64
7	0.000072	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) request id=0x0050, seq=2/512, ttl=64
8	0.000081	172.26.4.1	172.26.2.130	ICMP	98	Echo (ping) reply id=0x0050, seq=2/512, ttl=64
9	0.000082	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) request id=0x0050, seq=3/768, ttl=64
10	0.000083	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) reply id=0x0050, seq=3/768, ttl=64
11	0.000093	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) request id=0x0050, seq=4/1024, ttl=64
12	0.000094	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) reply id=0x0050, seq=4/1024, ttl=64
13	0.000095	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) request id=0x0050, seq=5/1280, ttl=64
14	0.000096	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) reply id=0x0050, seq=5/1280, ttl=64
15	0.000098	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) request id=0x0050, seq=6/1536, ttl=64
16	0.000099	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) reply id=0x0050, seq=6/1536, ttl=64
17	0.000100	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) request id=0x0050, seq=7/1792, ttl=64
18	0.000101	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) reply id=0x0050, seq=7/1792, ttl=64
19	0.000102	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) request id=0x0050, seq=8/2048, ttl=64
20	0.000103	172.26.2.130	172.26.4.1	ICMP	98	Echo (ping) reply id=0x0050, seq=8/2048, ttl=64



## Imagen del Wireshark (Router 2)

Hay paquetes que van y vienen de PC1 y Router 2

Esquema de la ruta del tráfico de los paquetes

## B) Ping desde equipo de Fábrica A hasta dirección pública del R2

File Edit Tabs Help

```
root@PC2:/tmp/pycore.44018/PC2.conf# ping 201.0.2.2
PING 201.0.2.2 (201.0.2.2) 56(84) bytes of data.
64 bytes from 201.0.2.2: icmp_req=1 ttl=62 time=0.167 ms
64 bytes from 201.0.2.2: icmp_req=2 ttl=62 time=0.113 ms
64 bytes from 201.0.2.2: icmp_req=3 ttl=62 time=0.097 ms
64 bytes from 201.0.2.2: icmp_req=4 ttl=62 time=0.159 ms
64 bytes from 201.0.2.2: icmp_req=5 ttl=62 time=0.097 ms
64 bytes from 201.0.2.2: icmp_req=6 ttl=62 time=0.098 ms
64 bytes from 201.0.2.2: icmp_req=7 ttl=62 time=0.071 ms
64 bytes from 201.0.2.2: icmp_req=8 ttl=62 time=0.117 ms
64 bytes from 201.0.2.2: icmp_req=9 ttl=62 time=0.114 ms
64 bytes from 201.0.2.2: icmp_req=10 ttl=62 time=0.096 ms
64 bytes from 201.0.2.2: icmp_req=11 ttl=62 time=0.097 ms
64 bytes from 201.0.2.2: icmp_req=12 ttl=62 time=0.097 ms
64 bytes from 201.0.2.2: icmp_req=13 ttl=62 time=0.107 ms
64 bytes from 201.0.2.2: icmp_req=14 ttl=62 time=0.097 ms
64 bytes from 201.0.2.2: icmp_req=15 ttl=62 time=0.096 ms
64 bytes from 201.0.2.2: icmp_req=16 ttl=62 time=0.114 ms
64 bytes from 201.0.2.2: icmp_req=17 ttl=62 time=0.097 ms
64 bytes from 201.0.2.2: icmp_req=18 ttl=62 time=0.102 ms
64 bytes from 201.0.2.2: icmp_req=19 ttl=62 time=0.097 ms
64 bytes from 201.0.2.2: icmp_req=20 ttl=62 time=0.157 ms
64 bytes from 201.0.2.2: icmp_req=21 ttl=62 time=0.099 ms
64 bytes from 201.0.2.2: icmp_req=22 ttl=62 time=0.097 ms
```

## Imagen del ping desde PC1 hacia Router 2 (pública)

Hay paquetes que van y vienen de PC1 y Router 2

## Imagen del Wireshark (Router 3)

Hay paquetes que van y vienen de PC1 y Router 2

## Imagen del Wireshark (Router 6)

# Ejercicio 12

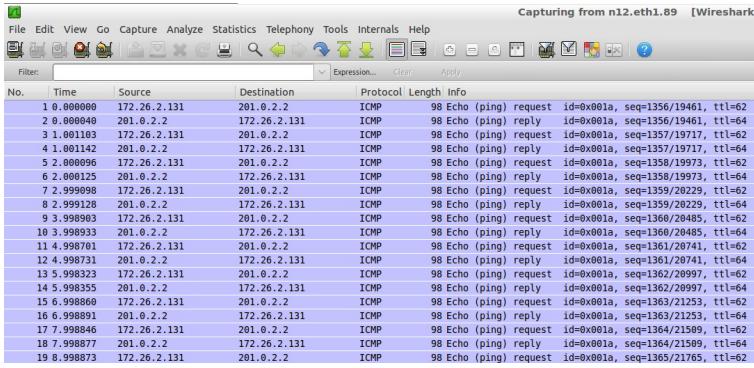


Imagen del Wireshark  
(Router 2 - IP pública)

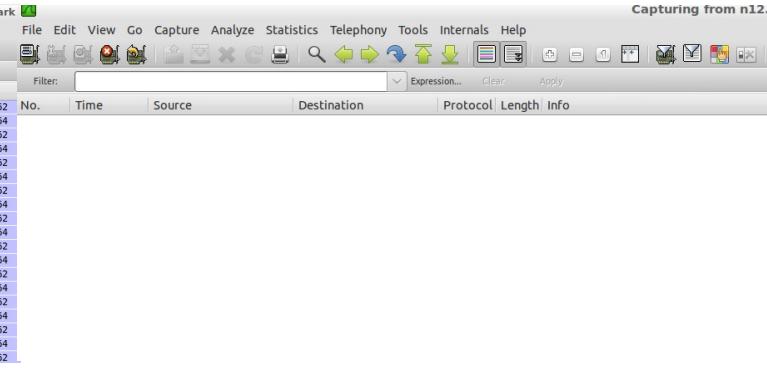


Imagen del Wireshark  
(Router 2 - IP privada)

Hay paquetes que van y vienen de PC1 y Router 2

No recibe paquetes

## C) Ping desde PC4 hasta la interfaz pública del R8

```
LXTerminal
File Edit Tabs Help
root@PC4:/tmp/pycore.44020/PC4.conf# ping 201.0.1.2
PING 201.0.1.2 (201.0.1.2) 56(84) bytes of data.
From 172.26.3.161 icmp_seq=1 Destination Port Unreachable
From 172.26.3.161 icmp_seq=2 Destination Port Unreachable
From 172.26.3.161 icmp_seq=3 Destination Port Unreachable
From 172.26.3.161 icmp_seq=4 Destination Port Unreachable
From 172.26.3.161 icmp_seq=5 Destination Port Unreachable
From 172.26.3.161 icmp_seq=6 Destination Port Unreachable
From 172.26.3.161 icmp_seq=7 Destination Port Unreachable
From 172.26.3.161 icmp_seq=8 Destination Port Unreachable
From 172.26.3.161 icmp_seq=9 Destination Port Unreachable
```

Imagen del ping desde PC4  
hacia la Interfaz pública del R8

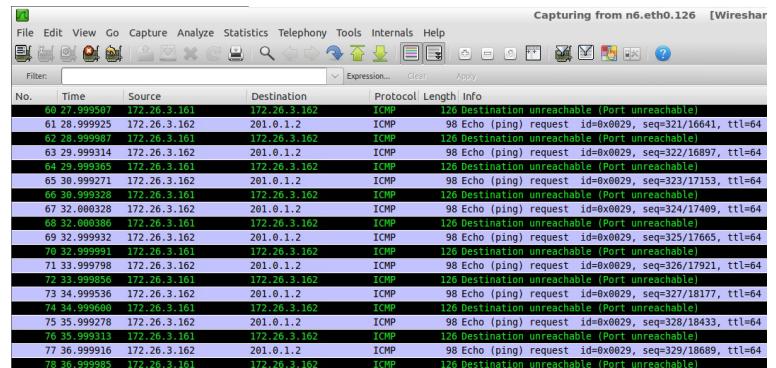


Imagen del Wireshark (PC4)



Hay paquetes que van y vienen de PC1 y Router 2

Imagen del Wireshark (R5)

Hay paquetes que van y vienen de PC1 y Router 2

Hay paquetes que van y vienen de PC1 y Router 2

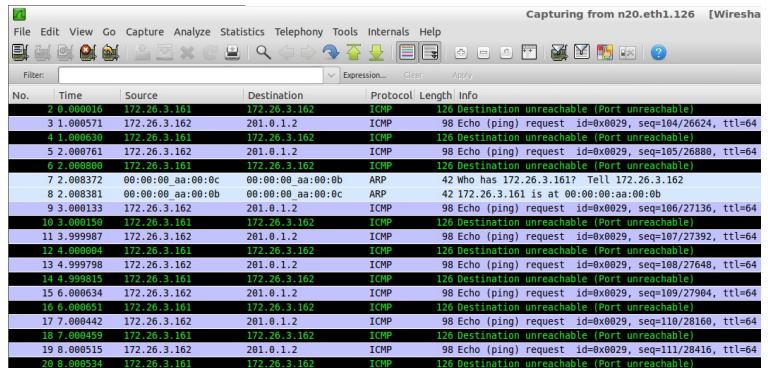


Imagen del Wireshark  
(IP R5 privada)

# Ejercicio 13

Modifique las tablas de ruteo del router 7 de manera que se generen paquetes ICMP con los siguientes códigos de error:

- Destination network unreachable.
- Time Exceeded.

Importante: No se podrán utilizar las opciones del comando ping para generar los errores.

## A) Destination Network Unreachable

### Resolución:

Primero se modificó la tabla de ruteo del router 7 borrando la ruta por defecto. Al hacer ping desde la PC3 hacia la PC1 se genera el error al no poder encontrar la dirección.

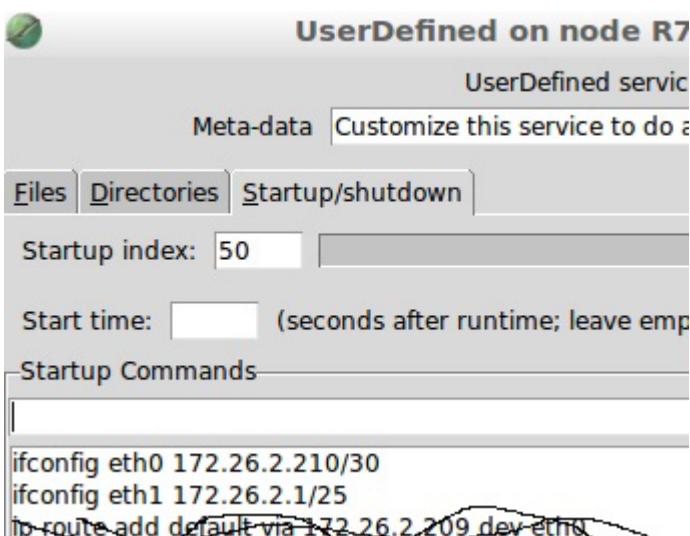


Imagen del router 7

```
LXTerminal
File Edit Tabs Help
root@PC3:/tmp/pycore.60883/PC3.conf# ping 172.26.2.130
PING 172.26.2.130 (172.26.2.130) 56(84) bytes of data.
From 172.26.2.1 icmp_seq=1 Destination Net Unreachable
From 172.26.2.1 icmp_seq=2 Destination Net Unreachable
From 172.26.2.1 icmp_seq=3 Destination Net Unreachable
From 172.26.2.1 icmp_seq=4 Destination Net Unreachable
From 172.26.2.1 icmp_seq=5 Destination Net Unreachable
From 172.26.2.1 icmp_seq=6 Destination Net Unreachable
From 172.26.2.1 icmp_seq=7 Destination Net Unreachable
From 172.26.2.1 icmp_seq=8 Destination Net Unreachable
From 172.26.2.1 icmp_seq=9 Destination Net Unreachable
From 172.26.2.1 icmp_seq=10 Destination Net Unreachable
From 172.26.2.1 icmp_seq=11 Destination Net Unreachable
From 172.26.2.1 icmp_seq=12 Destination Net Unreachable
From 172.26.2.1 icmp_seq=13 Destination Net Unreachable
From 172.26.2.1 icmp_seq=14 Destination Net Unreachable
From 172.26.2.1 icmp_seq=15 Destination Net Unreachable
From 172.26.2.1 icmp_seq=16 Destination Net Unreachable
From 172.26.2.1 icmp_seq=17 Destination Net Unreachable
From 172.26.2.1 icmp_seq=18 Destination Net Unreachable
From 172.26.2.1 icmp_seq=19 Destination Net Unreachable
From 172.26.2.1 icmp_seq=20 Destination Net Unreachable
From 172.26.2.1 icmp_seq=21 Destination Net Unreachable
From 172.26.2.1 icmp_seq=22 Destination Net Unreachable
```

Imagen de ping desde PC3 a PC1

Wireshark capture from interface n24.eth1.62. The table shows several ICMP Destination unreachable messages (code 126) between 172.26.2.1 and 172.26.2.2, indicating network unreachable errors.

No.	Time	Source	Destination	Protocol	Length	Info
12 5.018197	172.26.2.1	172.26.2.2		ICMP	126	Destination unreachable (Network unreachable)
13 6.015236	172.26.2.2	172.26.2.130		ICMP	98	Echo (ping) request id=0x0025, seq=196/50176, ttl=64
14 6.015272	172.26.2.1	172.26.2.2		ICMP	126	Destination unreachable (Network unreachable)
15 7.017497	172.26.2.2	172.26.2.130		ICMP	98	Echo (ping) request id=0x0025, seq=197/50432, ttl=64
16 7.017441	172.26.2.1	172.26.2.2		ICMP	126	Destination unreachable (Network unreachable)
17 8.024361	172.26.2.2	172.26.2.130		ICMP	98	Echo (ping) request id=0x0025, seq=198/50688, ttl=64
18 8.024395	172.26.2.1	172.26.2.2		ICMP	126	Destination unreachable (Network unreachable)
19 9.022433	172.26.2.2	172.26.2.130		ICMP	98	Echo (ping) request id=0x0025, seq=199/50944, ttl=64
20 9.022463	172.26.2.1	172.26.2.2		ICMP	126	Destination unreachable (Network unreachable)
21 10.024198	172.26.2.2	172.26.2.130		ICMP	98	Echo (ping) request id=0x0025, seq=200/51200, ttl=64
22 10.024236	172.26.2.1	172.26.2.2		ICMP	126	Destination unreachable (Network unreachable)
23 11.026809	172.26.2.2	172.26.2.130		ICMP	98	Echo (ping) request id=0x0025, seq=201/51456, ttl=64
24 11.026835	172.26.2.1	172.26.2.2		ICMP	126	Destination unreachable (Network unreachable)
25 12.028293	172.26.2.2	172.26.2.130		ICMP	98	Echo (ping) request id=0x0025, seq=202/51712, ttl=64
26 12.028329	172.26.2.1	172.26.2.2		ICMP	126	Destination unreachable (Network unreachable)
27 13.031526	172.26.2.2	172.26.2.130		ICMP	98	Echo (ping) request id=0x0025, seq=203/51968, ttl=64
28 13.031549	172.26.2.1	172.26.2.2		ICMP	126	Destination unreachable (Network unreachable)
29 14.032463	172.26.2.2	172.26.2.130		ICMP	98	Echo (ping) request id=0x0025, seq=204/52224, ttl=64
30 14.032494	172.26.2.1	172.26.2.2		ICMP	126	Destination unreachable (Network unreachable)

Imagen de Wireshark (R7 IP privada)

# Ejercicio 13

## B) Time Exceeded

### Resolución:

Primero se modificó la tabla de rutreo del router 6 modificando la ruta por defecto y eliminando los route add indicados para poder generar un bucle. Al hacer ping desde la PC3 hacia el router 1 (eth1) se genera el error ya que el mensaje queda atrapado en las mismas rutas una y otra vez hasta que excede el TTL del paquete.

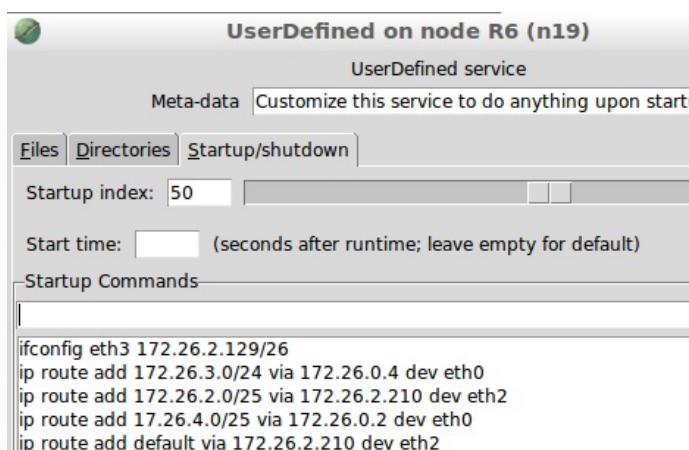


Imagen del router 6

```
File Edit Tabs Help
root@PC3:/tmp/pycore.41143/PC3.conf# ping 201.0.2.1
PING 201.0.2.1 (201.0.2.1) 56(84) bytes of data.
From 172.26.2.209 icmp_seq=1 Time to live exceeded
From 172.26.2.209 icmp_seq=2 Time to live exceeded
From 172.26.2.209 icmp_seq=3 Time to live exceeded
From 172.26.2.209 icmp_seq=4 Time to live exceeded
From 172.26.2.209 icmp_seq=5 Time to live exceeded
From 172.26.2.209 icmp_seq=6 Time to live exceeded
From 172.26.2.209 icmp_seq=7 Time to live exceeded
From 172.26.2.209 icmp_seq=8 Time to live exceeded
From 172.26.2.209 icmp_seq=9 Time to live exceeded
From 172.26.2.209 icmp_seq=10 Time to live exceeded
From 172.26.2.209 icmp_seq=11 Time to live exceeded
From 172.26.2.209 icmp_seq=12 Time to live exceeded
From 172.26.2.209 icmp_seq=13 Time to live exceeded
From 172.26.2.209 icmp_seq=14 Time to live exceeded
^C
--- 201.0.2.1 ping statistics ---
14 packets transmitted, 0 received, +14 errors, 100% packet loss, time 13094ms
root@PC3:/tmp/pycore.41143/PC3.conf#
```

Imagen de ping desde PC3 al router 1 eth1

No.	Time	Source	Destination	Protocol	Length	Info
1 0.000000		172.26.2.2	201.0.2.1	ICMP	98	Echo (ping) request id=0x001b, seq=26/6656, ttl=64
2 0.043968		172.26.2.209	172.26.2.2	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
3 1.002624		172.26.2.2	201.0.2.1	ICMP	98	Echo (ping) request id=0x001b, seq=27/6912, ttl=64
4 1.045948		172.26.2.209	172.26.2.2	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
5 2.005117		172.26.2.2	201.0.2.1	ICMP	98	Echo (ping) request id=0x001b, seq=28/7168, ttl=64
6 2.046880		172.26.2.209	172.26.2.2	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
7 3.008422		172.26.2.2	201.0.2.1	ICMP	98	Echo (ping) request id=0x001b, seq=29/7424, ttl=64
8 3.052376		172.26.2.209	172.26.2.2	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
9 4.009639		172.26.2.2	201.0.2.1	ICMP	98	Echo (ping) request id=0x001b, seq=30/7680, ttl=64
10 4.051120		172.26.2.209	172.26.2.2	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
11 5.013227		172.26.2.2	201.0.2.1	ICMP	98	Echo (ping) request id=0x001b, seq=31/7936, ttl=64
12 5.056735		172.26.2.209	172.26.2.2	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
13 6.015177		172.26.2.2	201.0.2.1	ICMP	98	Echo (ping) request id=0x001b, seq=32/8192, ttl=64
14 6.056145		172.26.2.209	172.26.2.2	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
15 7.018583		172.26.2.2	201.0.2.1	ICMP	98	Echo (ping) request id=0x001b, seq=33/8448, ttl=64
16 7.065938		172.26.2.209	172.26.2.2	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
17 8.020930		172.26.2.2	201.0.2.1	ICMP	98	Echo (ping) request id=0x001b, seq=34/8704, ttl=64
18 8.067554		172.26.2.209	172.26.2.2	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
19 9.024753		172.26.2.2	201.0.2.1	ICMP	98	Echo (ping) request id=0x001b, seq=35/8960, ttl=64

Imagen de Wireshark (PC3)

# Ejercicio 14

Realice los siguientes traceroute y explique lo observado:

- Desde la Pc6 a la ip pública de R8.
- Desde la Pc4 a la ip pública de R8.
- Desde la Pc1 a la ip del Server 1.

Indique los saltos de routers generados para cada punto.

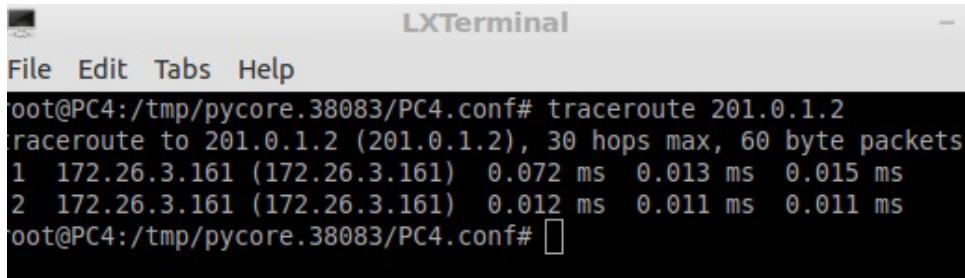
Para los caso en que interviene el Router 2, observe con Wireshark los paquetes que se generen en ambas interfaces.

## A) Traceroute desde PC6 a la IP pública de R8

```
root@PC6:/tmp/pycore.38083/PC6.conf# traceroute 201.0.1.2
traceroute to 201.0.1.2 (201.0.1.2), 30 hops max, 60 byte packets
 1  172.26.3.1 (172.26.3.1)  0.050 ms  0.031 ms  0.024 ms
 2  172.26.0.1 (172.26.0.1)  0.033 ms  0.024 ms  0.027 ms
 3  * * *
 4  201.0.2.1 (201.0.2.1)  0.043 ms  0.032 ms  0.030 ms
 5  201.0.1.2 (201.0.1.2)  0.045 ms  0.042 ms  0.036 ms
root@PC6:/tmp/pycore.38083/PC6.conf#
```

Podemos observar cuánto tardó en llegar al router default del host, luego el salto hacia el router 3, el enmascaramiento en el router 2 seguido del salto hacia el router 1 y por último el router 8.

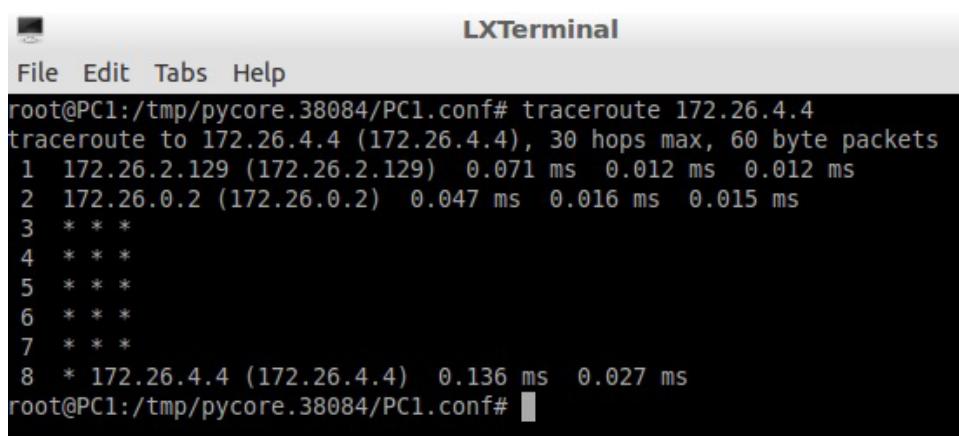
## B) Traceroute desde PC4 a la IP pública de R8



```
LXTerminal
File Edit Tabs Help
root@PC4:/tmp/pycore.38083/PC4.conf# traceroute 201.0.1.2
traceroute to 201.0.1.2 (201.0.1.2), 30 hops max, 60 byte packets
 1  172.26.3.161 (172.26.3.161)  0.072 ms  0.013 ms  0.015 ms
 2  172.26.3.161 (172.26.3.161)  0.012 ms  0.011 ms  0.011 ms
root@PC4:/tmp/pycore.38083/PC4.conf#
```

En este caso solo podemos observar el salto al router default del host debido al REJECT aplicado en anteriormente en otro ejercicio.

## C) Traceroute desde PC1 a la IP del Server 1



```
LXTerminal
File Edit Tabs Help
root@PC1:/tmp/pycore.38084/PC1.conf# traceroute 172.26.4.4
traceroute to 172.26.4.4 (172.26.4.4), 30 hops max, 60 byte packets
 1  172.26.2.129 (172.26.2.129)  0.071 ms  0.012 ms  0.012 ms
 2  172.26.0.2 (172.26.0.2)  0.047 ms  0.016 ms  0.015 ms
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
 8  * 172.26.4.4 (172.26.4.4)  0.136 ms  0.027 ms
root@PC1:/tmp/pycore.38084/PC1.conf#
```

En el último caso primero se dirige hacia el router default del host y hace el salto hacia el router 4 y por último al Servidor 1

# Ejercicio 14

Ejercicio  
A

Capturing from n12.eth0.87 [Wireshark 1.6.7]						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	201.0.2.2	201.0.1.2	UDP	74	Source port: 35948 Destination port: 33443
2	0.000015	201.0.2.1	201.0.2.2	ICMP	102	Time-to-live exceeded (Time to live exceeded in transit)
3	0.000049	201.0.2.2	201.0.1.2	UDP	74	Source port: 38291 Destination port: 33444
4	0.000057	201.0.2.1	201.0.2.2	ICMP	102	Time-to-live exceeded (Time to live exceeded in transit)
5	0.000087	201.0.2.2	201.0.1.2	UDP	74	Source port: 55594 Destination port: 33445
6	0.000095	201.0.2.1	201.0.2.2	ICMP	102	Time-to-live exceeded (Time to live exceeded in transit)
7	0.000126	201.0.2.2	201.0.1.2	UDP	74	Source port: 60445 Destination port: 33446
8	0.000147	201.0.1.2	201.0.2.2	ICMP	102	Destination unreachable (Port unreachable)
9	0.000183	201.0.2.2	201.0.1.2	UDP	74	Source port: 54388 Destination port: 33447
10	0.000197	201.0.1.2	201.0.2.2	ICMP	102	Destination unreachable (Port unreachable)
11	0.000228	201.0.2.2	201.0.1.2	UDP	74	Source port: 53166 Destination port: 33448
12	0.000241	201.0.1.2	201.0.2.2	ICMP	102	Destination unreachable (Port unreachable)
13	0.000271	201.0.2.2	201.0.1.2	UDP	74	Source port: 43913 Destination port: 33449
14	5.011176	00:00:00_aa:00:1e	00:00:00_aa:00:1f	ARP	42	Who has 201.0.2.1? Tell 201.0.2.2
15	5.011210	00:00:00_aa:00:1f	00:00:00_aa:00:1e	ARP	42	Who has 201.0.2.2? Tell 201.0.2.1
16	5.011238	00:00:00_aa:00:1f	00:00:00_aa:00:1e	ARP	42	201.0.2.1 is at 00:00:00_aa:00:1f
17	5.011230	00:00:00_aa:00:1e	00:00:00_aa:00:1f	ARP	42	201.0.2.2 is at 00:00:00_aa:00:1e

Imagen de Wireshark(R2 IP pública)

Ejercicio  
A

Capturing from n12.eth1.87 [Wireshark 1.6.7]						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.26.3.2	201.0.1.2	UDP	74	Source port: 36145 Destination port: 33440
2	0.000029	172.26.3.2	201.0.1.2	UDP	74	Source port: 54578 Destination port: 33441
3	0.000055	172.26.3.2	201.0.1.2	UDP	74	Source port: 42024 Destination port: 33442
4	0.000075	172.26.3.2	201.0.1.2	UDP	74	Source port: 35948 Destination port: 33443
5	0.000100	201.0.2.1	172.26.3.2	ICMP	102	Time-to-live exceeded (Time to live exceeded in transit)
6	0.000127	172.26.3.2	201.0.1.2	UDP	74	Source port: 38291 Destination port: 33444
7	0.000146	201.0.2.1	172.26.3.2	ICMP	102	Time-to-live exceeded (Time to live exceeded in transit)
8	0.000165	172.26.3.2	201.0.1.2	UDP	74	Source port: 55594 Destination port: 33445
9	0.000178	201.0.2.1	172.26.3.2	ICMP	102	Time-to-live exceeded (Time to live exceeded in transit)
10	0.000204	172.26.3.2	201.0.1.2	UDP	74	Source port: 60445 Destination port: 33446
11	0.000230	201.0.1.2	172.26.3.2	ICMP	102	Destination unreachable (Port unreachable)
12	0.000256	172.26.3.2	201.0.1.2	UDP	74	Source port: 54388 Destination port: 33447
13	0.000280	201.0.1.2	172.26.3.2	ICMP	102	Destination unreachable (Port unreachable)
14	0.000306	172.26.3.2	201.0.1.2	UDP	74	Source port: 53166 Destination port: 33448
15	0.000324	201.0.1.2	172.26.3.2	ICMP	102	Destination unreachable (Port unreachable)
16	0.000349	172.26.3.2	201.0.1.2	UDP	74	Source port: 43913 Destination port: 33449
17	5.011280	00:00:00_aa:00:03	00:00:00_aa:00:00	ARP	42	Who has 172.26.4.1? Tell 172.26.4.2
18	5.011262	00:00:00_aa:00:00	00:00:00_aa:00:03	ARP	42	Who has 172.26.4.2? Tell 172.26.4.1
19	5.011309	00:00:00_aa:00:00	00:00:00_aa:00:03	ARP	42	172.26.4.1 is at 00:00:00_aa:00:00
20	5.011321	00:00:00_aa:00:03	00:00:00_aa:00:00	ARP	42	172.26.4.2 is at 00:00:00_aa:00:03

Imagen de Wireshark(R2 IP privada)

Ejercicio  
B

Capturing from n12.eth1.87 [Wireshark 1.6.7]						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.26.3.2	201.0.1.2	UDP	74	Source port: 36145 Destination port: 33440
2	0.000029	172.26.3.2	201.0.1.2	UDP	74	Source port: 54578 Destination port: 33441

Capturing from n12.eth0.87 [Wireshark 1.6.7]						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.26.3.2	201.0.1.2	UDP	74	Source port: 36145 Destination port: 33440
2	0.000029	172.26.3.2	201.0.1.2	UDP	74	Source port: 54578 Destination port: 33441

Imagen de Wireshark(R2 IP pública y privada)

## Observaciones:

En Wireshark se puede observar como cambia la IP original de los paquetes enmascarandola con la de R2, también como en algunos de los paquetes el TTL expira y retorna a la IP en donde expiró.

# Conclusión

Aprendimos mucho desarrollando este trabajo, principalmente a manejar un entorno virtual en otro sistema operativo, luego definir ips para cada red y subredes con sus respectivas máscaras, definir las rutas para que los routers puedan conectarse entre las redes, enmascarar los datos de ip privada a pública, darle permisos o denegarlos dependiendo la red requerida, mandar mensajes desde la consola por puertos específicos entre las redes, ver qué rutas utilizó el paquete para llegar a su destino, derivar tráfico por diferentes routers, utilizar Wireshark, analizar los datos provistos por dicha herramienta, y típicos errores entre las conexiones (Destination network unreachable, Time Exceeded).

## Experiencias personales

En primer lugar nos llevó mucho tiempo poder hacer correr el virtualbox ya que no nos andaba bien la virtualización, luego de buscar por días pudimos encontrar un post sobre configuración en la bios y descargamos un archivo que modificaba el registro en el cual luego de aplicarlo comenzó a funcionar y pudimos acceder a virtualbox, nos dimos cuenta que nos cambiaba toda la configuración de la computadora al utilizarlo y teníamos que reiniciar la pc relativamente de 10 a 15 veces por sesión de estudio en el tpe. Recurrimos a buscar otras alternativas y nos encontramos con VMWARE, impecable, es más fácil conectar usb o pasar datos del escritorio a la máquina virtual en un santiamén, luego tuvimos otro inconveniente respecto al tamaño de la pantalla el cual hacia que no podíamos ver los botones para aceptar o cancelar las modificaciones en la configuración de los routers, específicamente en el service donde prácticamente era lo más importante para hacer los primeros ejercicios de configuración para luego ver cómo se transmitían y circulaban los paquetes de un lado a otro.