

**Universidad Nacional Autónoma de Nicaragua**  
**UNAN - León**  
**Facultad de Ciencias y Tecnologías**



**Practica 4**

**Componente:**

o **Redes de Computadoras**

**Integrante:**

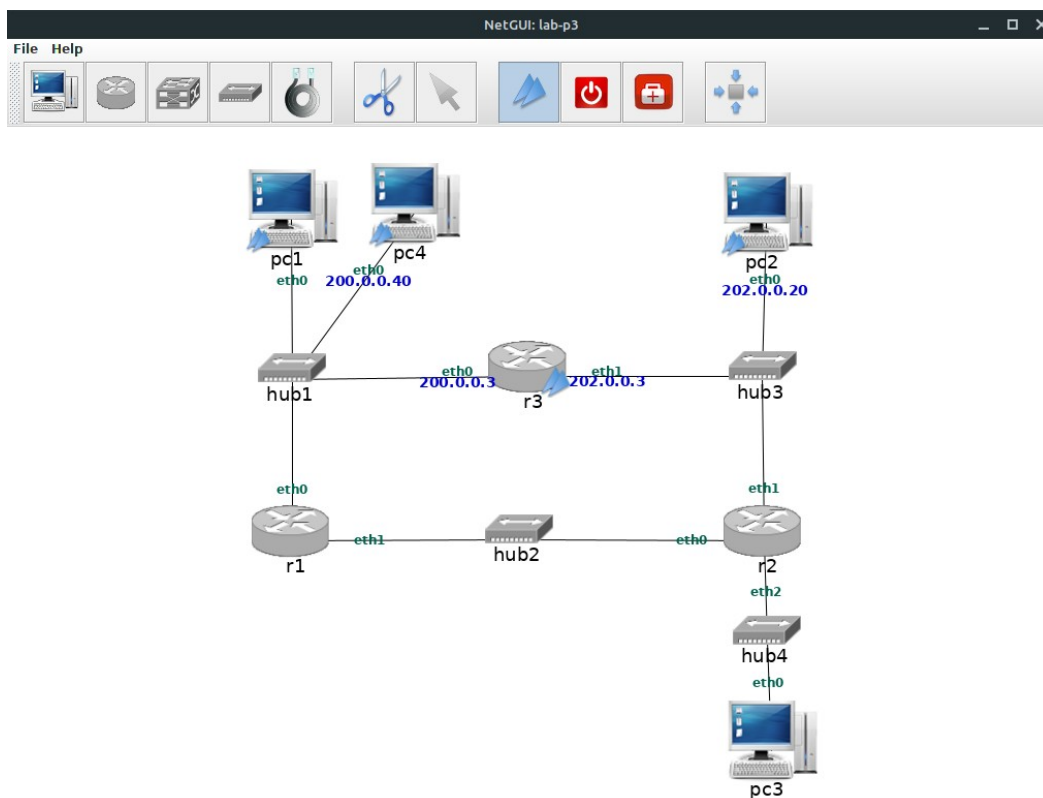
➤ **Bismarck Antonio Berrios Lopez**

## 1. Configuración de tablas de encaminamiento con route

Lanza ahora NetGUI. En el menú, elige File → Open y selecciona la carpeta lab-p3 en la que está el escenario. Verás aparecer la red de la figura 1.

Arranca únicamente las siguientes máquinas: pc1 , pc4 , r3 y pc2 .

Este escenario realiza una configuración asignando direcciones IP a todas las interfaces de las máquinas, excepto a pc1 . Esta configuración inicial está almacenada en el fichero /etc/network/interfaces de cada una de las máquinas, tal y como se ha visto en la práctica anterior.



Teniendo en cuenta que sólo están estas máquinas arrancadas, responde a las siguientes cuestiones:

1. Escriba en pc1 la orden `ping 127.0.0.1`. Obtienes mensajes de respuesta? Quien esta enviando esos mensajes de respuesta? Con la orden `route` puedes consultar la tabla de encaminamiento. Comprueba la tabla de encaminamiento de pc1 para ayudarte a entender lo que esta pasando.

Si se obtiene mensaje de respuesta, los mensajes los envia la propia pc1.

```
pc1
pc1:~# route -n
Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref    Use Iface
pc1:~#
```

2. Modifica el fichero `/etc/network/interfaces` de `pc1` para que `pc1` tenga una dirección IP acorde a la subred a la que está conectado. (Nota: Deberás consultar previamente la máscara de la subred que tienen las otras maquinas conectadas a la misma subred que `pc1`). Reinicia la red en `pc1` para que se aplique la configuración que has escrito en el fichero `/etc/network/interfaces`

```
pc1
GNU nano 2.0.7 File: /etc/network/interfaces
#####
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
    address 200.0.0.30
    netmask 255.255.255.0

[ Read 11 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^T To Spell
```

```
pc1
pc1:~# nano /etc/network/interfaces
pc1:~# /etc/init.d/networking restart
Reconfiguring network interfaces...done.
pc1:~#
```

3. Comprueba con route cómo en pc1 , tras asignar la dirección IP a su interfaz de red, se ha añadido automáticamente una entrada en la tabla de encaminamiento. Con esta tabla de encaminamiento en Pc1. A que otras direcciones IP crees que pc1 podrá enviar datagramas IP?

```

pc1
pc1:~# nano /etc/network/interfaces
pc1:~# /etc/init.d/networking restart
Reconfiguring network interfaces...done.
pc1:~# route -n
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
200.0.0.0      0.0.0.0         255.255.255.0   U        0      0        0 eth0
pc1:~#

```

Podrá enviarme datagramas a todos los equipos que pertenezcan a la red 200.0.0.0

4. Dado que el resto de las máquinas tienen ya configurada una dirección IP, podrás suponer fácilmente cuál es el contenido de su tabla de encaminamiento:

- Cuál crees que será la tabla de encaminamiento de pc4 ?. Compruébalo consultando su tabla. Con esta tabla de encaminamiento, A que otras direcciones IP crees que Pc4 podrá enviar datagramas IP?

Creo que pc4 tendra en su tabla de encaminamiento a la red 200.0.0.0.

```

pc4
--- Starting Netkit phase 2 init script ---
#####
Lab directory (host): /home/bisanb1/Documents/Redes_Computadoras/Laboratorio/4.I
P-ARP-ICMP/lab-p3
Version: <none>
Author: <none>
Email: <none>
Web: <none>
Description:
<none>
#####
--- Netkit phase 2 initialization terminated ---

pc4 login: root (automatic login)
pc4:~# route -n
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
200.0.0.0      0.0.0.0         255.255.255.0   U        0      0        0 eth0
pc4:~#

```

Creo que solamente a las direcciones ip que pertenezcan la red 200.0.0.0 es decir a 200.0.0.30 y 200.0.0.3

- Cuál crees que será la tabla de encaminamiento de pc2 ?. Compruébalo consultando su tabla. Con esta tabla de encaminamiento, a que otras direcciones IP crees que pc2 podrá enviar datagramas IP?

La tabla de encaminamiento de pc2 mostrara que puede enviar datagramas a la red 202.0.0.20.

```
pc2:~# route -n
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
202.0.0.0      0.0.0.0         255.255.255.0   U        0      0          0 eth0
pc2:~#
```

creo que podra enviar a todas las direcciones dentro del rango de la red 202.0.0.0 que por el momento en la simulacion es a la 202.0.0.3

- Cuál crees que será la tabla de encaminamiento de r3 ?. Compruébalo consultando su tabla. Con esta tabla de encaminamiento, Crees que r3 puede enviar datagramas IP a pc1 y pc4 ? Y a pc2 .

Creo que r3 tendra en su tabla de encaminamiento las redes 200.0.0.0 y 202.0.0.0

```

r3
Version: <none>
Author: <none>
Email: <none>
Web: <none>
Description:
<none>

=====
— Netkit phase 2 initialization terminated —

r3 login: root (automatic login)
Last login: Sun Sep 15 23:39:11 UTC 2019 on tty1
r3:~#
r3 login: root (automatic login)
Last login: Sun Sep 15 23:39:12 UTC 2019 on tty0
r3:~# route -n
route -n
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
202.0.0.0      0.0.0.0         255.255.255.0   U        0      0        0 eth1
200.0.0.0      0.0.0.0         255.255.255.0   U        0      0        0 eth0
r3:~#

```

Si podrá enviar datagramas a pc1 y pc4 incluso a pc2.

5. Haz ping desde pc1 a pc4 y haz ping desde pc1 a la dirección r3(eth0) . Ten en cuenta que no puedes utilizar los nombres pc1 , pc4 , etc. en el ping , sino que debes usar las direcciones IP correspondientes. Funcionan estos ping ? Qué entradas de las tablas de encaminamiento se consultan en cada caso?

Todos los pines funcionaron perfectamente y en cada caso se consulta la entra #1 correspondiente a la red 200.0.0.0

```

pc1
— Netkit phase 2 initialization terminated —

pc1 login: root (automatic login)
Last login: Sun Sep 15 23:39:03 UTC 2019 on tty1
pc1:~# ping 200.0.0.40
PING 200.0.0.40 (200.0.0.40) 56(84) bytes of data.
64 bytes from 200.0.0.40: icmp_seq=1 ttl=64 time=11.1 ms
64 bytes from 200.0.0.40: icmp_seq=2 ttl=64 time=0.461 ms
64 bytes from 200.0.0.40: icmp_seq=3 ttl=64 time=0.464 ms

--- 200.0.0.40 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.461/4.028/11.159/5.042 ms
pc1:~# ping 200.0.0.3
PING 200.0.0.3 (200.0.0.3) 56(84) bytes of data.
64 bytes from 200.0.0.3: icmp_seq=1 ttl=64 time=10.4 ms
64 bytes from 200.0.0.3: icmp_seq=2 ttl=64 time=0.721 ms

--- 200.0.0.3 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.721/5.589/10.458/4.869 ms
pc1:~#

```

6. Haz un ping de pc1 a pc2 y haz un ping de pc1 a la dirección r3(eth1) .  
Funcionan estos ping ? Por qué?

```
pc1:~# ping 202.0.0.20
connect: Network is unreachable
pc1:~# ping 202.0.0.3
connect: Network is unreachable
pc1:~#
```

No funcionan porque no hay en la tabla de encaminamiento de pc1 una forma de llegar a la red 202.0.0.0. ni un gateway al cual preguntarle.

7. Agrega una ruta con el comando route en pc1 para que los datagramas IP que no sean para su propia subred los envíe a través del router r3.

```
pc1:~# ping 202.0.0.20
connect: Network is unreachable
pc1:~# ping 202.0.0.3
connect: Network is unreachable
pc1:~# route add default gw 200.0.0.3
pc1:~# route -n
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
200.0.0.0      0.0.0.0         255.255.255.0   U      0      0      0 eth0
0.0.0.0        200.0.0.3       0.0.0.0         UG     0      0      0 eth0
pc1:~#
```

8. Haz ahora ping desde pc1 a r3(eth1). Funciona este ping? Que estradas de las tablas de encaminamiento se consultan?

```
pc1:~# ping 202.0.0.20
connect: Network is unreachable
pc1:~# ping 202.0.0.3
connect: Network is unreachable
pc1:~# route add default gw 200.0.0.3
pc1:~# route -n
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
200.0.0.0        0.0.0.0         255.255.255.0    U      0      0      0 eth0
0.0.0.0          200.0.0.3       0.0.0.0          UG     0      0      0 eth0
pc1:~# ping 202.0.0.3
PING 202.0.0.3 (202.0.0.3) 56(84) bytes of data.
64 bytes from 202.0.0.3: icmp_seq=1 ttl=64 time=10.6 ms
64 bytes from 202.0.0.3: icmp_seq=2 ttl=64 time=0.491 ms
64 bytes from 202.0.0.3: icmp_seq=3 ttl=64 time=0.358 ms
64 bytes from 202.0.0.3: icmp_seq=4 ttl=64 time=0.361 ms
^X64 bytes from 202.0.0.3: icmp_seq=5 ttl=64 time=0.260 ms

--- 202.0.0.3 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 0.260/2.425/10.659/4.117 ms
pc1:~#
```

Si funciona se consulta ambas entradas y al comprobar que en la primera no puede encontrar el destino lo realiza con la segunda mediante la cual si llega a la ip especificada.

9. haz un ping de pc1 a pc2. Por que no funciona este ping?

```
pc1:~# route add default gw 200.0.0.3
pc1:~# route -n
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
200.0.0.0        0.0.0.0         255.255.255.0    U      0      0      0 eth0
0.0.0.0          200.0.0.3       0.0.0.0          UG     0      0      0 eth0
pc1:~# ping 202.0.0.3
PING 202.0.0.3 (202.0.0.3) 56(84) bytes of data.
64 bytes from 202.0.0.3: icmp_seq=1 ttl=64 time=10.6 ms
64 bytes from 202.0.0.3: icmp_seq=2 ttl=64 time=0.491 ms
64 bytes from 202.0.0.3: icmp_seq=3 ttl=64 time=0.358 ms
64 bytes from 202.0.0.3: icmp_seq=4 ttl=64 time=0.361 ms
^X64 bytes from 202.0.0.3: icmp_seq=5 ttl=64 time=0.260 ms

--- 202.0.0.3 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 0.260/2.425/10.659/4.117 ms
pc1:~# ping 202.0.0.20
PING 202.0.0.20 (202.0.0.20) 56(84) bytes of data.

--- 202.0.0.20 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2008ms
pc1:~#
```

No funciona debido a que r3 no sabe que hacer con ese paquete ni a donde enviarlo.



10. En función del contenido actual de las tablas de encaminamiento de las máquinas y del router , explica qué máquinas podrán comunicarse entre sí.

- Pc1 podría comunicarse con PC4 y r3(eth0 y1).
- Pc4 podría comunicarse con PC1 y r3(eth0).
- Pc2 podría comunicarse con r3(eth1).
- r3 puede comunicarse con PC1, PC2 y PC4.

11. Agrega las rutas que consideres necesarias utilizando el comando route para que funcione un ping de pc1 a pc2 y de pc4 a pc2. Te en cuenta que podras utilizar , rutas de maquina, rutas de subred o rutas por defecto.

Se agregaron rutas de subredes y sus puertas de enlace en el router r3 y se agrego la ruta por defecto a cada maquina .

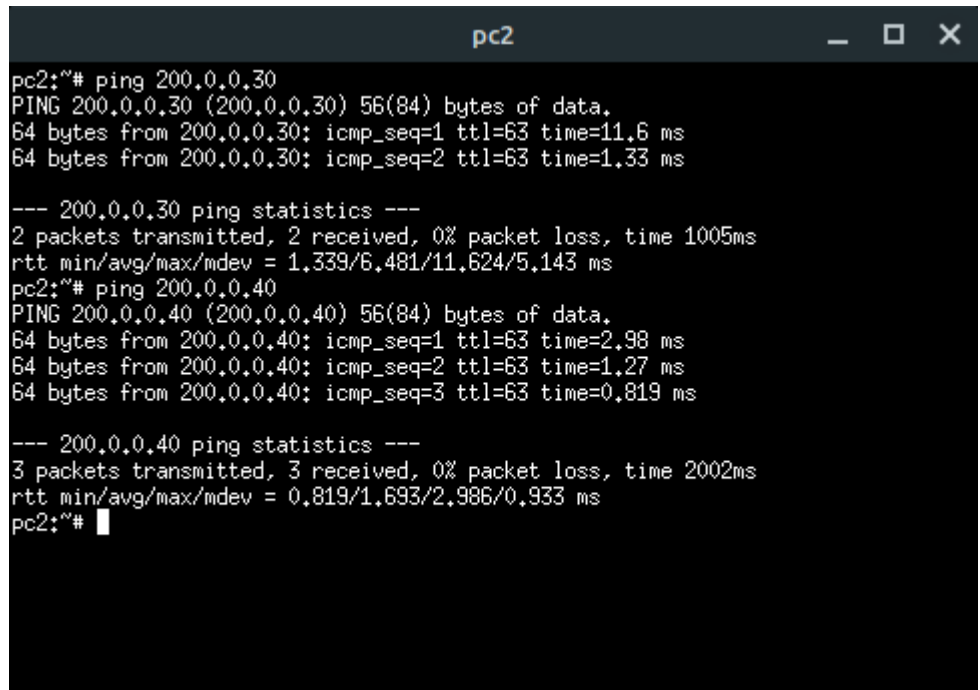
Pc1 y Pc4 tienen como gw r3(eth0) y Pc2 tiene a r3(eth1).

```

r3
r3:~# route-n
-bash: route-n: command not found
r3:~# route -n
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
202.0.0.0        202.0.0.3      255.255.255.0   UG    0      0        0 eth1
202.0.0.0        0.0.0.0        255.255.255.0   U      0      0        0 eth1
200.0.0.0        200.0.0.3      255.255.255.0   UG    0      0        0 eth0
200.0.0.0        0.0.0.0        255.255.255.0   U      0      0        0 eth0
r3:~#
```

12. Indica si crees que con la configuración que has realizado funcionará un ping de pc2 a pc1 y de pc2 a pc4 . Compruébalo.

Si creo que funcionara.



```
pc2:~# ping 200.0.0.30
PING 200.0.0.30 (200.0.0.30) 56(84) bytes of data:
64 bytes from 200.0.0.30: icmp_seq=1 ttl=63 time=11.6 ms
64 bytes from 200.0.0.30: icmp_seq=2 ttl=63 time=1.33 ms

--- 200.0.0.30 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1005ms
rtt min/avg/max/mdev = 1.339/6.481/11.624/5.143 ms
pc2:~# ping 200.0.0.40
PING 200.0.0.40 (200.0.0.40) 56(84) bytes of data:
64 bytes from 200.0.0.40: icmp_seq=1 ttl=63 time=2.98 ms
64 bytes from 200.0.0.40: icmp_seq=2 ttl=63 time=1.27 ms
64 bytes from 200.0.0.40: icmp_seq=3 ttl=63 time=0.819 ms

--- 200.0.0.40 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 0.819/1.693/2.986/0.933 ms
pc2:~#
```

### 1.1 Captura de Trafico.

Antes de comenzar a realizar los siguientes ejercicios, espera al menos 10 minutos después de haber ejecutado el último ping del apartado anterior.

1. consulta el estado de las caches ARP en los pcs y en el router. Explica su contenido.

Las caches arp estan vacias porque tiene un tiempo de vida inferior a 10 minutos x lo que luego de ese tiempo las entradas se borran.

2. Arranca en pc4 un tcpdump para c capturar trafico en su interfaz eth0, guardando la captura en un fichero (tal y como lo hiciste en la practica 0).

3. Ejecuta en pc1 un ping a pc4 que envíe sólo 1 paquete ICMP Echo Request ( ping -c 1 <máquinaDestino> ).

4. Interrumpe la captura en pc4 (Ctrl+C).

```
pc4:~# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
200.0.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
0.0.0.0 200.0.0.3 0.0.0.0 UG 0 0 0 eth0
pc4:~# tcpdump -i eth0 -s 0 -w /hosthome/pc4.cap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
6 packets captured
6 packets received by filter
0 packets dropped by kernel
pc4:~# arp -n
pc4:~#
```

5. Comprueba el estado de las cachés de ARP en pc1 , pc4 , pc2 y r3 .Explica su contenido.

La cache arp de pc1 obtiene la mac address de pc4 al preguntar por ella al broadcast. Y pc4 la obtiene al recibir la solicitud de ping de pc1. R3 y pc2 no reciben ninguna

```
pc1:~# ping -c 1 200.0.0.40
PING 200.0.0.40 (200.0.0.40) 56(84) bytes of data:
64 bytes from 200.0.0.40: icmp_seq=1 ttl=64 time=11.0 ms

--- 200.0.0.40 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 11.096/11.096/11.096/0.000 ms
pc1:~# arp
Address Hwtype Hwaddress Flags Mask Iface
200.0.0.40 ether A2:54:94:B3:A0:02 C eth0
pc1:~#
```

```
pc2:~# arp
pc2:~#
```

```
pc4:~# tcpdump -i eth0 -s 0 -w /hosthome/pc4.cap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
6 packets captured
6 packets received by filter
0 packets dropped by kernel
pc4:~# arp
address Hwtype Hwaddress Flags Mask Iface
200.0.0.30 ether 1E:CB:8E:A4:30:E2 C eth0
pc4:~#
```

```
r3:~# arp
r3:~#
```

comunicación porque el hub le envió el paquete a todos los dispositivos conectados en la red y el router no tiene necesidad de responder a una solicitud que se realiza dentro de la misma red y tampoco envía la solicitud de broadcast a la red de pc2.

6. Arranca en un terminal de la máquina real la aplicación aplicación wireshark para cargar el fichero de captura que has obtenido. Observa los siguientes campos en los mensajes de la captura:

Mensaje de solicitud de ARP que envía pc1 a pc4 .

- Dirección Ethernet destino **Broadcast**
- Dirección Ethernet origen **1e:cb8e:a4:30:e2**
- Tipo en la cabecera Ethernet **ARP**
- Contenido del mensaje de solicitud de ARP: localiza el campo que contiene la dirección IP de la máquina sobre la que se está preguntando su dirección Ethernet.

```
Address Resolution Protocol (request)
  Hardware type: Ethernet (1)
  Protocol type: IPv4 (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: request (1)
  Sender MAC address: 1e:cb:8e:a4:30:e2 (1e:cb:8e:a4:30:e2)
  Sender IP address: 200.0.0.30
  Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
  Target IP address: 200.0.0.40
```

Mensaje de respuesta de ARP que envía pc4 a pc1 .

- Dirección Ethernet destino **1e:cb8e:a4:30:e2**
- Dirección Ethernet origen **a2:54:94:b3:a0:02**
- Tipo en la cabecera Ethernet **ARP**
- Contenido del mensaje de respuesta de ARP: localiza el campo que contiene la dirección Ethernet solicitada.

```
Address Resolution Protocol (reply)
  Hardware type: Ethernet (1)
  Protocol type: IPv4 (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: reply (2)
  Sender MAC address: a2:54:94:b3:a0:02 (a2:54:94:b3:a0:02)
  Sender IP address: 200.0.0.40
  Target MAC address: 1e:cb:8e:a4:30:e2 (1e:cb:8e:a4:30:e2)
  Target IP address: 200.0.0.30
```

Datagrama IP que envía pc1 a pc4 .

- Dirección Ethernet destino **a2:54:94:b3:a0:02**
- Dirección Ethernet origen **1e:cb8e:a4:30:e2**
- Tipo en la cabecera Ethernet **IPV4**
- Dirección IP origen **200.0.0.30**
- Dirección IP destino **200.0.0.40**
- Campo TTL **64**

Datagrama IP que envía pc4 a pc1 .

- Dirección Ethernet destino **1e:cb8e:a4:30:e2**
- Dirección Ethernet origen **a2:54:94:b3:a0:02**
- Tipo en la cabecera Ethernet **IPV4**
- Dirección IP origen **200.0.0.40**
- Dirección IP destino **200.0.0.30**
- Campo TTL **64**

7. Espera a que la caché de ARP de pc1 esté vacía. Ahora vamos a analizar el tráfico desde pc1 a pc2 . Cuántas capturas de tráfico crees que son necesarias para ver todos los paquetes que se generan en el escenario cuando se comunican pc1 y pc2 ? 2 .

8. Arranca un tcpdump en r3(eth0) y en pc2 para ver todos los paquetes que se generan cuando pc1 y pc2 se comunican, guardando las capturas de tráfico en dos ficheros diferentes.

9. Ejecuta en pc1 un ping a pc2 que envíe sólo 1 paquete ( ping 3 -c 1 <máquinaDestino> ).

10. Interrumpe las capturas ( Ctrl+C ).

11. Comprueba el estado de las cachés de ARP en pc1 , pc2 , pc4 y r3 . Explica su contenido.

La pc1 solo conoce a r3 debido a que es de quien recibe informacion al igual que pc2 solo reconoce al router por el mismo motivo. Sin embargo el r3 que es el encargado de comunicar ambas computadoras en distintas redes las conoce a ambas pc.

```
pc1:~# arp
Address      Hwtype      Hwaddress    Flags Mask    Iface
200.0.0.3    ether      26:F3:B0:24:E4:02  C              eth0
pc1:~#

pc2:~# tcpdump -i eth0 -s 0 -w /home/pc2.cap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
6 packets captured
6 packets received by filter
0 packets dropped by kernel
pc2:~# arp
Address      Hwtype      Hwaddress    Flags Mask    Iface
202.0.0.3    ether      02:21:3F:66:81:CA  C              eth0
pc2:~#

pc4:~# arp
pc4:~#

r3:~# tcpdump -s 0 -w /home/r3.cap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
6 packets captured
6 packets received by filter
0 packets dropped by kernel
r3:~# arp
Address      Hwtype      Hwaddress    Flags Mask    Iface
202.0.0.20   ether      F8:F9A7:8F:C8:FE  C              eth1
200.0.0.30   ether      1E:1C:8E:A4:30:E2  C              eth0
r3:~#
```

12. Arranca en un terminal de la máquina real la aplicación wireshark para cargar el/los fichero/s de captura que has obtenido. Observa las capturas y construye una tabla por cada uno de los paquetes que se han capturado indicando el tipo y su contenido atendiendo a los siguientes campos:

Mensajes ARP	Dirección Ethernet destino Dirección Ethernet origen Tipo en la cabecera Ethernet Contenido del mensaje Ethernet
Datagramas IP	Dirección Ethernet destino Dirección Ethernet origen Tipo en la cabecera Ethernet Dirección IP origen Dirección IP destino Campo TTL

## PC2

Tipo Mensaje	ETH D	ETH O	Tipo C. ETH	Conten. Mensaje	IP O	IP D	TTL
ARP	Broadcast	02:21:ef:66:81:ca	ARP	Who has 202.0.0.20? Tell 202.0.0.3			
ARP	02:21:ef:66:81:ca	f6:fa:d7:8f:cb:fe	ARP	202.0.0.20 is at f6:fa:d7:8f:cb:fe			
IP	f6:fa:d7:8f:cb:fe	02:21:ef:66:81:ca	IPV4		200.0.0.30	202.0.0.20	63
IP	02:21:ef:66:81:ca	f6:fa:d7:8f:cb:fe	IPV4		202.0.0.20	200.0.0.30	64
ARP	02:21:ef:66:81:ca	f6:fa:d7:8f:cb:fe	ARP	Who has 202.0.0.3? Tell 202.0.0.20			
ARP	f6:fa:d7:8f:cb:fe	02:21:ef:66:81:ca	ARP	202.0.0.3 is at 02:21:3f:66:81:ca			

### R3

Tipo Mensaje	ETH D	ETH O	Tipo C. ETH	Conten. Mensaje	IP O	IP D	TTL
ARP	Broadcast	1e:cb:8e:a4:30:e2	ARP	Who has 200.0.0.3? Tell 200.0.0.30			
ARP	1e:cb:8e:a4:30:e2	26:f3:bd:24:e4:02	ARP	200.0.0.3 is at 26:f3:bd:24:e4:02			
IP	26:f3:bd:24:e4:02	1e:cb:8e:a4:30:e2	IPV4		200.0.0.30	202.0.0.20	64
IP	1e:cb:8e:a4:30:e2	26:f3:bd:24:e4:02	IPV4		202.0.0.20	200.0.0.30	63
ARP	1e:cb:8e:a4:30:e2	26:f3:bd:24:e4:02	ARP	Who has 200.0.0.30? Tell 200.0.0.3			
ARP	26:f3:bd:24:e4:02	1e:cb:8e:a4:30:e2	ARP	200.0.0.30 is at 1e:cb:8e:a4:30:e2			

13. Observa cómo un datagrama IP viaja por cada una de las subredes hacia el destino en tramas Ethernet diferentes. Qué únicos campos de la cabecera IP cambian al atravesar diferentes subredes? El TTL y Ethernet Origen.

14. Espera a que la caché de ARP de interfaz eth1 ,pc2 esté vacía. Arranca en r3 un tcpdump para capturar tráfico en su guardando la captura en otro fichero diferente.

15. Ejecuta en pc2 un ping a pc1 de forma que envíe sólo 1 paquete ICMP Echo Request ( ping -c 1 <máquinaDestino> ). A continuación ejecuta en pc2 un ping a pc4 de forma que envíe sólo 1 paquete ICMP Echo Request.

16. Interrumpe la captura en r3(eth1) ( Ctrl+C ).

17. Cuántos mensajes ARP crees que se habrán capturado en el fichero? Compruébalo cargando el fichero de captura en el wireshark . Creo que seran 8.

1	0.000000	26:f3:bd:24:e4:02	Broadcast	ARP	42 Who has 200.0.0.30? Tell 200.0.0.3
2	0.000052	1e:cb:8e:a4:30:e2	26:f3:bd:24:e4:02	ARP	42 200.0.0.30 is at 1e:cb:8e:a4:30:e2
3	0.000058	202.0.0.20	200.0.0.30	ICMP	98 Echo (ping) request id=0xe804, seq=1/256, ttl=63 (
4	0.000121	200.0.0.30	202.0.0.20	ICMP	98 Echo (ping) reply id=0xe804, seq=1/256, ttl=64 (
5	4.997539	1e:cb:8e:a4:30:e2	26:f3:bd:24:e4:02	ARP	42 Who has 200.0.0.3? Tell 200.0.0.30
6	4.997578	26:f3:bd:24:e4:02	1e:cb:8e:a4:30:e2	ARP	42 200.0.0.3 is at 26:f3:bd:24:e4:02
7	9.741859	26:f3:bd:24:e4:02	Broadcast	ARP	42 Who has 200.0.0.40? Tell 200.0.0.3
8	9.742319	a2:54:94:b3:a0:02	26:f3:bd:24:e4:02	ARP	42 200.0.0.40 is at a2:54:94:b3:a0:02
9	9.742340	202.0.0.20	200.0.0.40	ICMP	98 Echo (ping) request id=0xea04, seq=1/256, ttl=63 (
10	9.742770	200.0.0.40	202.0.0.20	ICMP	98 Echo (ping) reply id=0xea04, seq=1/256, ttl=64 (
11	14.730998	a2:54:94:b3:a0:02	26:f3:bd:24:e4:02	ARP	42 Who has 200.0.0.3? Tell 200.0.0.40
12	14.731036	26:f3:bd:24:e4:02	a2:54:94:b3:a0:02	ARP	42 200.0.0.3 is at 26:f3:bd:24:e4:02



## 2. Configuración de tablas de encaminamiento mediante ficheros de configuración

Arranca el resto de las máquinas y routers de la figura y obtendrás un diagrama similar a la figura 2. Ten en cuenta que en pc1 ya tendrás configurada una dirección IP, mantén esta configuración.

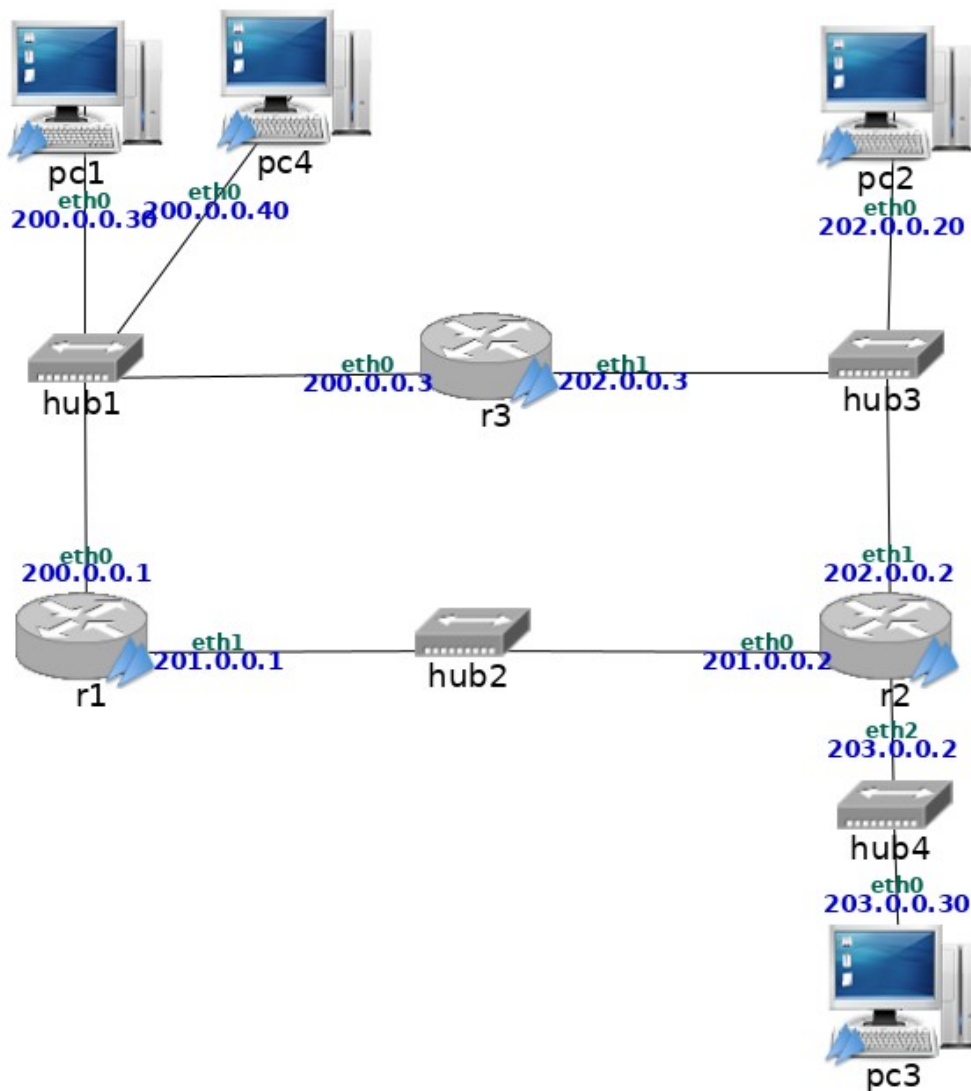


Figura 2: Todas las máquinas arrancadas

1. Cuántas subredes observas en la figura? Escribe la dirección de cada una de estas subredes junto con su máscara. 4 subredes.

SubRed	Mascara
200.0.0.0	255.255.255.0
201.0.0.0	255.255.255.0
202.0.0.0	255.255.255.0
203.0.0.0	255.255.255.0

2. Reinicia las máquinas pc1 , pc2 y pc4 .

3. Consulta las tablas de encaminamiento en todas las máquinas y routers , comprobarás que las rutas que configuraste en el apartado anterior han desaparecido. Los pcs y routers sólo tienen ruta a las subredes a las que están directamente conectados. Por tanto sólo se podrán comunicar con las máquinas con las que son vecinas.

4. Con la configuración actual, indica qué máquinas se pueden comunicar entre sí, especificando sus direcciones IP.

Pc1 y pc4 200.0.0.30 y 200.0.0.40

5. Modifica el fichero /etc/network/interfaces en los ordenadores y en los routers de la red de forma que funcionen las siguientes rutas. Podrás utilizar rutas de máquina, de subred o rutas por defecto:

a ) Conectividad entre pc1 y pc2 en los dos sentidos, a través de las siguientes rutas:

pc1 ⇒ r3 ⇒ pc2

pc2 ⇒ r3 ⇒ pc1

Ejecuta en pc1 la orden ping -c 3 <dirIPpc2> para comprobar si hay conectividad entre las máquinas. Para verificar que el camino es el descrito previamente, deberás ejecutar route en pc1 y observar qué entrada de la tabla de encaminamiento se utiliza cuando se desea alcanzar pc2 . Esta entrada debería indicar que el siguiente salto es r3 . A continuación en r3 deberás ejecutar route y observar qué entrada de la tabla de encaminamiento se utiliza cuando se desea alcanzar pc2 . Esta entrada debería indicar que r3 no necesita ningún router adicional para alcanzar pc2 .

De forma análoga deberás consultar las tablas de encaminamiento en el sentido pc2 ⇒ r3 ⇒ pc1.

```

pc1
Version: <none>
Author: <none>
Email: <none>
Web: <none>
Description:
<none>

*****

--- Netkit phase 2 initialization terminated ---

pc1 login: root (automatic login)
Last login: Wed Oct 2 15:14:06 UTC 2019 on tty1
pc1:~# ping -c 3 202.0.0.20
PING 202.0.0.20 (202.0.0.20) 56(84) bytes of data:
64 bytes from 202.0.0.20: icmp_seq=1 ttl=63 time=13.3 ms
64 bytes from 202.0.0.20: icmp_seq=2 ttl=63 time=0.989 ms
64 bytes from 202.0.0.20: icmp_seq=3 ttl=63 time=1.11 ms

--- 202.0.0.20 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.989/5.157/13.373/5.809 ms
pc1:~#

```

**Pc1 route:**

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
202.0.0.0	200.0.0.3	255.255.255.0	UG	0	0	0	eth0

**R3 route:**

Destination	Gateway	Genmask
202.0.0.0	202.0.0.3	255.255.255.0
202.0.0.0	*	255.255.255.0
200.0.0.0	200.0.0.3	255.255.255.0
200.0.0.0	*	255.255.255.0

R3 no necesita un router adicional para llegar a pc2 ni a Pc1 ya que esta conectado directamente a las redes de ambas pc.

**Pc2 route:**

200.0.0.0	202.0.0.3	255.255.255.0
-----------	-----------	---------------

B ) Conectividad entre pc2 y pc3 en los dos sentidos, a través de la siguientes rutas:

pc2 ⇒ r2 ⇒ pc3

pc3 ⇒ r2 ⇒ pc2

Ejecuta en pc2 la orden ping -c 3 <dirIPpc3> para comprobar si hay conectividad entre las máquinas. Para verificar que el camino es el descrito previamente, deberás ejecutar route en pc2 y observar qué entrada de la tabla de encaminamiento se utiliza cuando se desea alcanzar pc3 . Esta entrada debería indicar que el siguiente salto es r2 . A continuación en r2 deberás ejecutar route y observar qué entrada de la tabla de encaminamiento se utiliza cuando se desea alcanzar pc3 . Esta entrada debería indicar que r2 no necesita ningún router adicional para alcanzar pc3 .

De forma análoga deberás consultar las tablas de encaminamiento en el sentido pc3 ⇒ r2 ⇒ pc2.

```

pc2:~# ping -c3 203.0.0.30
PING 203.0.0.30 (203.0.0.30) 56(84) bytes of data.
64 bytes from 203.0.0.30: icmp_seq=1 ttl=63 time=12.8 ms
64 bytes from 203.0.0.30: icmp_seq=2 ttl=63 time=0.803 ms
64 bytes from 203.0.0.30: icmp_seq=3 ttl=63 time=1.02 ms

--- 203.0.0.30 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.803/4.889/12.842/5.624 ms
pc2:~#

```

**Pc2** route:

203.0.0.0	202.0.0.2	255.255.255.0
-----------	-----------	---------------

**R2** route: para alcanzar pc3 se utiliza

202.0.0.0	202.0.0.2	255.255.255.0
202.0.0.0	*	255.255.255.0
203.0.0.0	203.0.0.2	255.255.255.0

la ultima linea de esta tabla y para llegar a pc2 la primera. Ambas desde el mismo router r2.

C ) Conectividad entre pc1y pc3 en los dos sentidos, a través de las siguientes rutas:

pc1 ⇒ r1 ⇒ r2 ⇒ pc3

pc3 ⇒ r2 ⇒ r3 ⇒ pc1

Ejecuta enpc1 la orden ping -c 3 <dirIPpc3> para comprobar si hay conectividad entre las máquinas. Para verificar que el camino es el descrito previamente, deberás ejecutar route en pc1 y observar qué entrada de la tabla de encaminamiento se utiliza cuando se desea alcanzar pc3 . Esta entrada debería indicar que el siguiente salto es r1 . Después, deberás ejecutar route en r1 y observar qué entrada de la tabla de encaminamiento se utiliza cuando se desea alcanzar pc3 . Esta entrada debería indicar que el siguiente salto es r2 . A continuación en r2 deberás ejecutar route y observar qué entrada de la tabla de encaminamiento se utiliza cuando se desea alcanzar pc3 . Esta entrada debería indicar que r2 no necesita ningún router adicional para alcanzar pc3 .

De forma análoga deberás consultar las tablas de encaminamiento en el sentido pc3 ⇒ r2 ⇒ r3 ⇒ pc1.

```

pc1:~# ping -c3 203.0.0.30
PING 203.0.0.30 (203.0.0.30) 56(84) bytes of data.
64 bytes from 203.0.0.30: icmp_seq=1 ttl=62 time=43.3 ms
64 bytes from 203.0.0.30: icmp_seq=2 ttl=62 time=1.62 ms
64 bytes from 203.0.0.30: icmp_seq=3 ttl=62 time=1.21 ms

--- 203.0.0.30 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2005ms
rtt min/avg/max/mdev = 1.218/15.394/43.339/19.760 ms
pc1:~#

```

**Pc1** route:

203.0.0.0	200.0.0.1	255.255.255.0
-----------	-----------	---------------

**r1** route: r1 indica que cuando se quiere llegar a pc3 se envian los paquetes por r2.

203.0.0.0	201.0.0.2	255.255.255.0
-----------	-----------	---------------

**r2** route: la tabla especifica que no necesita router adicional para

203.0.0.0	203.0.0.2	255.255.255.0
203.0.0.0	*	255.255.255.0
200.0.0.0	202.0.0.3	255.255.255.0

llegar a pc3. Y que cuando se desea comunicar pc3 a pc1 se envia el trafico hacia r3.

**R3** route: no necesita un router extra para llegar a pc1.

Destination	Gateway	Genmask
202.0.0.0	202.0.0.3	255.255.255.0
202.0.0.0	*	255.255.255.0
200.0.0.0	200.0.0.3	255.255.255.0
200.0.0.0	*	255.255.255.0

6. Ejecuta en pc1 un traceroute hacia pc2 y en pc2 uno hacia pc1 para comprobar que las rutas son las especificadas.

```

pc2:~# traceroute 200.0.0.30
traceroute to 200.0.0.30 (200.0.0.30), 64 hops max, 40 byte packets
 1  202.0.0.3 (202.0.0.3)  1 ms  1 ms  0 ms
 2  200.0.0.30 (200.0.0.30)  1 ms  1 ms  1 ms
pc2:~#

pc1:~# traceroute 202.0.0.20
traceroute to 202.0.0.20 (202.0.0.20), 64 hops max, 40 byte packets
 1  200.0.0.3 (200.0.0.3)  11 ms  1 ms  0 ms
 2  202.0.0.20 (202.0.0.20)  10 ms  1 ms  1 ms
pc1:~#

```

7. Ejecuta en pc2 un traceroute hacia pc3 y en pc3 uno hacia pc2 para comprobar que las rutas son las especificadas.

```
pc2:~# traceroute 203.0.0.30
traceroute to 203.0.0.30 (203.0.0.30), 64 hops max, 40 byte packets
 1  202.0.0.2 (202.0.0.2)  11 ms  1 ms  1 ms
 2  203.0.0.30 (203.0.0.30)  9 ms  1 ms  1 ms
pc2:~#
```

```
pc3:~# traceroute 202.0.0.20
traceroute to 202.0.0.20 (202.0.0.20), 64 hops max, 40 byte packets
 1  203.0.0.2 (203.0.0.2)  1 ms  1 ms  0 ms
 2  202.0.0.20 (202.0.0.20)  1 ms  1 ms  1 ms
pc3:~#
```

8. Ejecuta en pc1 un traceroute hacia pc3 y en pc3 uno hacia pc1 para comprobar que las rutas son las especificadas. En este último traceroute observarás que aparecen unos \*. A qué crees que se debe? En la Práctica 3 se estudiarán más en detalle este tipo de casos.

```
pc1:~# traceroute 203.0.0.30
traceroute to 203.0.0.30 (203.0.0.30), 64 hops max, 40 byte packets
 1  200.0.0.1 (200.0.0.1)  12 ms  1 ms  1 ms
 2  202.0.0.2 (202.0.0.2)  21 ms  1 ms  1 ms
 3  203.0.0.30 (203.0.0.30)  1 ms  1 ms  1 ms
pc1:~#
```

```
pc3:~# traceroute 200.0.0.30
traceroute to 200.0.0.30 (200.0.0.30), 64 hops max, 40 byte packets
 1  203.0.0.2 (203.0.0.2)  1 ms  1 ms  0 ms
 2  * * *
 3  200.0.0.30 (200.0.0.30)  2 ms  1 ms  1 ms
pc3:~#
```

## 2.1. Capturas de tráfico

Antes de comenzar a realizar los siguientes ejercicios asegúrate de que las rutas entre las máquinas son las especificadas en el apartado anterior.

1. Lanza tcpdump , almacenando los paquetes capturados en ficheros diferentes, capturando tráfico en las siguientes interfaces: r1(eth0) , r2(eth0) ,en r3(eth1) y pc3(eth0) .
2. Ejecuta en pc1 un ping a pc3 que envíe sólo 2 paquetes ( ping -c 2 <máquinaDestino> ).
3. Interrumpe las 4 capturas ( Ctrl+C ).

4. En un terminal de la máquina real lanza la aplicación wireshark 4 veces, una con cada fichero, para poder ver simultáneamente las distintas capturas. Observa en las capturas cómo los datagramas IP que se envían y reciben con la orden ping contienen un mensaje de ICMP. Comprueba en estos datagramas:

- Dirección IP origen
- Dirección IP destino
- TTL en la cabecera IP
- Tipo de Protocolo en la cabecera IP
- Tipo y Código en la cabecera ICMP.

5. Consultando las capturas, responde a las siguientes cuestiones:

- a) En qué se distinguen los mensajes de ida del ping de los mensajes de vuelta? En los router por los que pasan en sus viajes. Y sus ttl.
- b) En qué capturas se pueden ver los mensajes de ida del ping ? Y los mensajes de vuelta? Por qué?
  - Los mensajes de ida se pueden ver en la captura de r1, r2 y pc3.
  - Los mensajes de vuelta se muestran en r1, r3 y pc3.

Esto se debe a que toman caminos diferentes de ida y de vuelta. En el caso de r1 recibe ambos mensajes porque al llegar a r3 y enviarlo al hub este se lo envía a todos los dispositivos conectados incluyendo r1.

- c) Comprueba los valores del campo TTL de la cabecera IP de todos los datagramas de todas las capturas y explica dichos valores.

R1.

# de paquete	TTL	Explicacion
3	64	Al ser el paquete inicial de solicitud este es el primer router al que llega. Por eso posee el valor inicial 64.
6	62	A este router llega luego de pasar por r2 y r3 producto de la conexión con el hub que posee por lo tanto obtiene el valor 62.
7	64	Al ser el paquete inicial de solicitud este es el primer router al que llega. Por eso posee el valor inicial 64.
8	62	A este router llega luego de pasar por r2 y r3 producto de esto obtiene el valor 62.

## R2

# de paquete	TTL	Explicacion
3	63	Tiene el ttl de 63 ya que el mensaje paso por el R1 previamente
4	63	Tiene el ttl de 63 ya que el mensaje paso por el R1 previamente

## R3

# de paquete	TTL	Explicacion
3	63	Los mensajes de respuesta tiene un ttl de 63
4	63	debido a que previamente pasaron por R2 unicamente.

## Pc3

# de paquete	TTL	Explicacion
3	62	El valor es 62 debido a que el mensaje paso anteriormente por 2 router (r1 y r2).
4	64	El ttl es de 64 ya que no ha pasado por ningun router puesto que pc3 es quien lo genera.
5	62	El valor es 62 debido a que el mensaje paso anteriormente por 2 router (r1 y r2).
6	64	El ttl es de 64 ya que no ha pasado por ningun router puesto que pc3 es quien lo genera.

6. Arranca de nuevo tcpdump en las mismas máquinas e interfaces que lo has hecho anteriormente pero guardando las capturas en otros ficheros diferentes: en r1(eth0) , en r2(eth0) , en r3(eth1) y en pc3(eth0) .



7. Ejecuta en pc1 la orden traceroute a pc3 .

8. Cuando la orden anterior haya terminado, interrumpe las capturas ( Ctrl+C ).

9. A la vista del resultado que se ha obtenido en llegar de pc1: qué saltos intermedios ha atravesado un paquete para llegar de pc1 a pc3 ?

```
pc1:~# traceroute 203.0.0.30
traceroute to 203.0.0.30 (203.0.0.30), 64 hops max, 40 byte packets
 1  200.0.0.1 (200.0.0.1)  11 ms  1 ms  0 ms
 2  202.0.0.2 (202.0.0.2)  33 ms  1 ms  1 ms
 3  203.0.0.30 (203.0.0.30)  12 ms  1 ms  1 ms
pc1:~#
```

2 saltos el 1 por 200.0.0.1 y el segundo por 202.0.0.2

10. Abre con wireshark los ficheros de captura que has obtenido. Identifica en los ficheros de capturas los siguientes paquetes:

- Los 3 mensajes enviados por pc1 con TTL=1
- Los 3 ICMP de TTL excedido enviados por r1
- Los 3 mensajes enviados por pc1 con TTL=2
- Los 3 ICMP de TTL excedido enviados por r2
- Los 3 mensajes enviados por pc1 con TTL=3
- Los 3 ICMP de puerto inalcanzable enviados por pc3

11. Consultando las capturas, responde a las siguientes cuestiones:

- Por qué ruta van viajando los mensajes enviados por pc1 con TTL creciente? Viajan de r1 a r2.
- Por qué ruta viajan los ICMP enviados por r1 ? Qué dirección IP usa r1 como IP de origen el enviar esos ICMP? Viajan por la red 200.0.0.0 utilizando como origen la ip 200.0.0.1
- Por qué ruta viajan los ICMP enviados por r2 ? Qué dirección IP usa r2 como IP de origen el enviar esos ICMP? En mis capturas R2 no envia ningun ICMP
- Por qué ruta viajan los ICMP enviados por pc3 ? Viajan por r2 hacia r3.