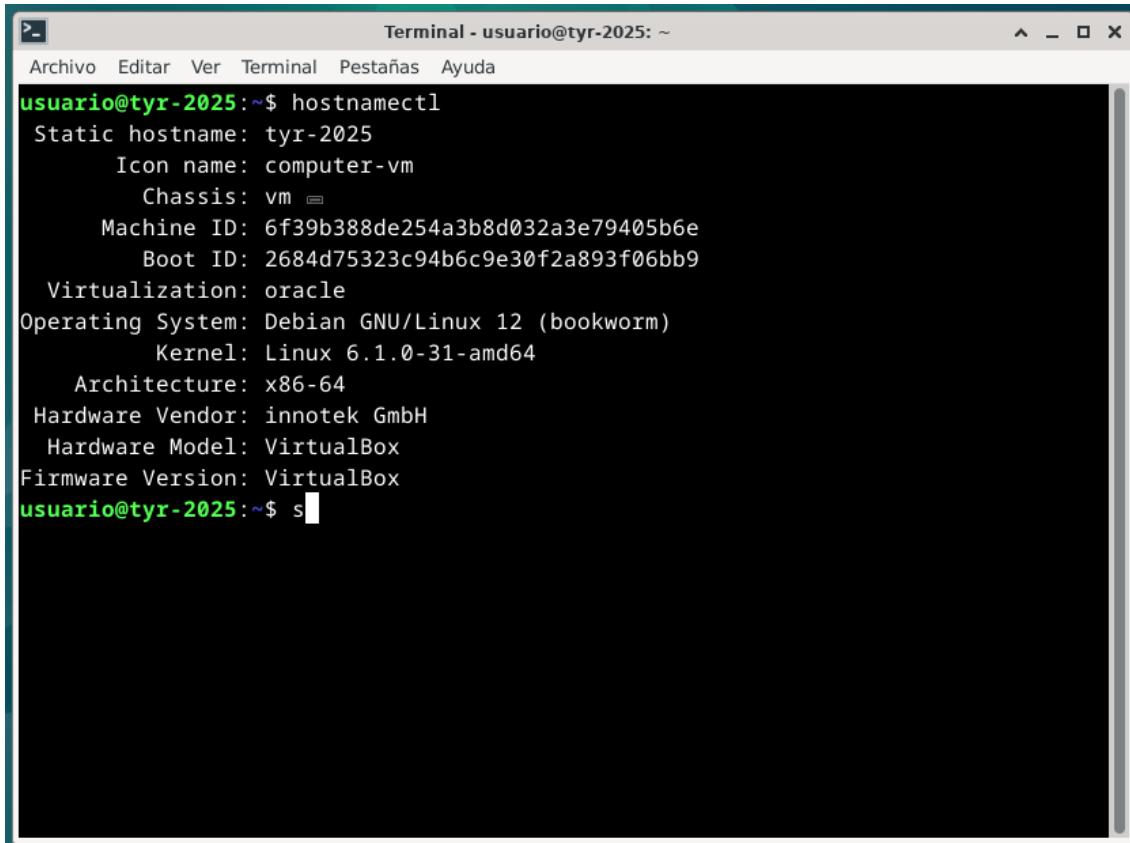


TPL 1 - Configuración inicial de la red del laboratorio

Alumno: Joaquin Dominguez

Primero instalamos la MV facilitada por el equipo docente

A terminal window titled "Terminal - usuario@tyr-2025: ~" with a menu bar containing "Archivo", "Editar", "Ver", "Terminal", "Pestañas", and "Ayuda". The terminal shows the command "hostnamectl" being executed, which outputs system information. The prompt "usuario@tyr-2025:~\$" is shown in green. The output is as follows:

```
usuario@tyr-2025:~$ hostnamectl
Static hostname: tyr-2025
Icon name: computer-vm
Chassis: vm
Machine ID: 6f39b388de254a3b8d032a3e79405b6e
Boot ID: 2684d75323c94b6c9e30f2a893f06bb9
Virtualization: oracle
Operating System: Debian GNU/Linux 12 (bookworm)
Kernel: Linux 6.1.0-31-amd64
Architecture: x86-64
Hardware Vendor: innotek GmbH
Hardware Model: VirtualBox
Firmware Version: VirtualBox
usuario@tyr-2025:~$ s
```

Usamos el comando hostnamectl para ver la distribución sobre la que se va a trabajar (la de la MV facilitada)

Usamos los siguientes comando para instalar kathara:

Debian 12

1. Add Kathará public key to your keyring:

```
8e6cbba6b991abe76646193862b759810" | sudo gpg --dearmor -o /usr/share/keyrings/ppa-kathara-archive-keyring.gpg
```

2. Add the Kathará repository:

- o Create a `kathara.list` file in `/etc/apt/sources.list.d` and put inside the following lines:

```
deb [ signed-by=/usr/share/keyrings/ppa-kathara-archive-keyring.gpg ] http://ppa.launchpad.net/kathara-lab/ppa/ubuntu-22.04 main  
deb-src [ signed-by=/usr/share/keyrings/ppa-kathara-archive-keyring.gpg ] http://ppa.launchpad.net/kathara-lab/ppa/ubuntu-22.04 main
```

- o Or run the following commands:

```
echo "deb [ signed-by=/usr/share/keyrings/ppa-kathara-archive-keyring.gpg ] http://ppa.launchpad.net/kathara-lab/ppa/ubuntu-22.04 main" > /etc/apt/sources.list.d/kathara.list  
echo "deb-src [ signed-by=/usr/share/keyrings/ppa-kathara-archive-keyring.gpg ] http://ppa.launchpad.net/kathara-lab/ppa/ubuntu-22.04 main" > /etc/apt/sources.list.d/kathara.list
```

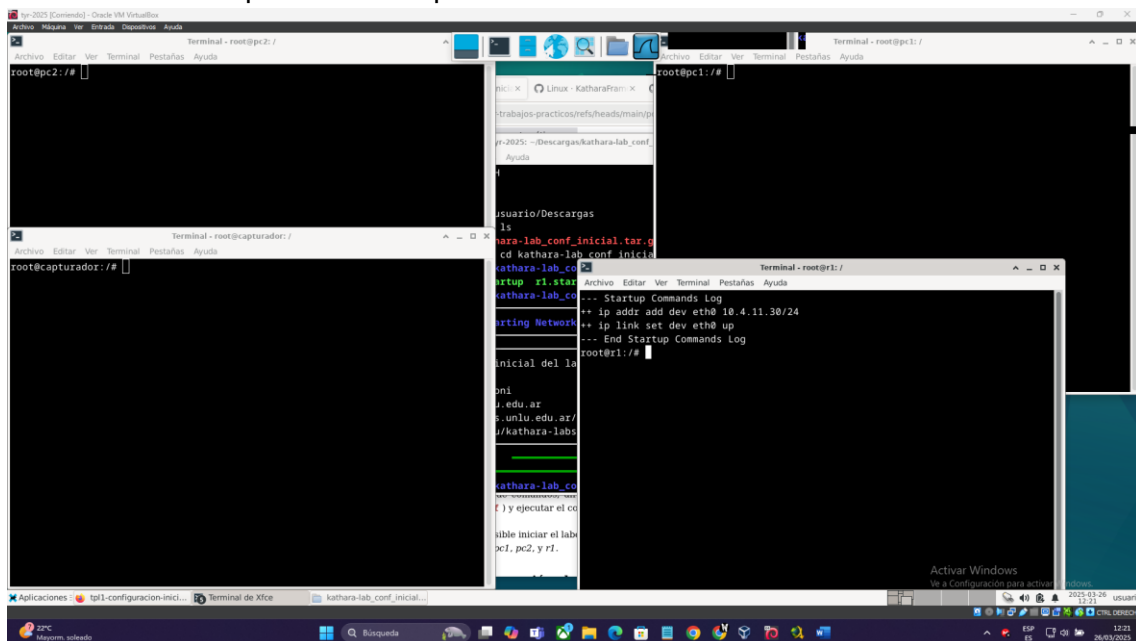
3. Update your apt cache by running:

```
sudo apt update
```

4. Install Kathará:

```
sudo apt install kathara
```

Iniciamos kathara para verificar que todo funcione correctamente

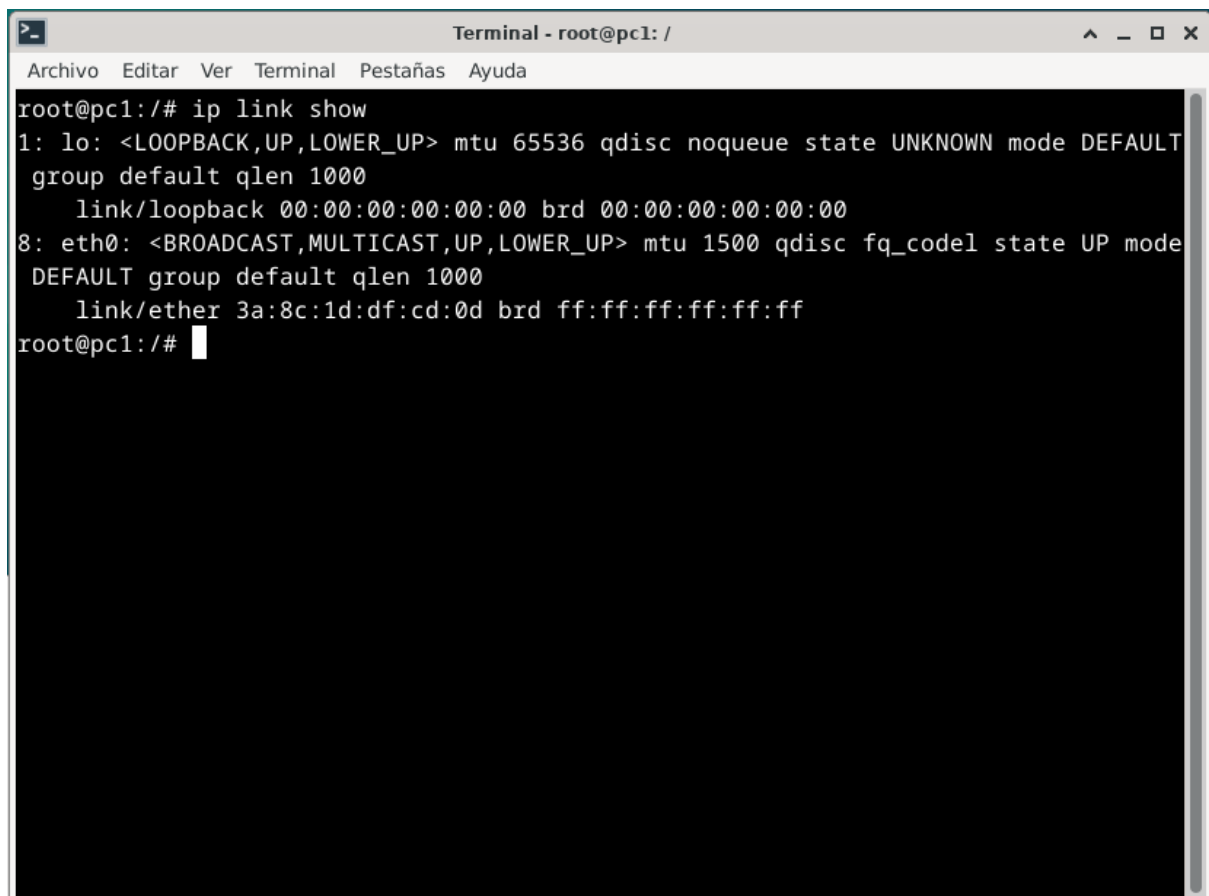


Ahora si comenzamos con el TP:

1. Verificar la/s interfaces de red (comúnmente llamada placa de red o NIC) que el sistema operativo haya detectado en pc1 y pc2, y listar su información en pantalla.

¿Que comando utilizó? ¿Cual es el nombre de las interfaces? ¿Que parte de la salida le indicó cual es la interfaz que se encuentra conectada? ¿Cual es el nombre de la interfaz que se encuentra conectada a la red?

- **Comando Utilizado:** `ip link show`
- **Nombre de las Interfaces:** En la salida de `ip link show`, se observa una lista de interfaces de red junto con sus nombres. En mi caso `eth0` y `lo` (de loopback)
- **Interfaz Conectada:** Lo que indica el estado de la interfaz conectada es el state UP
- **Nombre de la Interfaz Conectada a la Red:** La única interfaz conectada a la red es `eth0`.



```
Terminal - root@pc1: /
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
root@pc1:/# ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
    group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
8: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode
    DEFAULT group default qlen 1000
    link/ether 3a:8c:1d:df:cd:0d brd ff:ff:ff:ff:ff:ff
root@pc1:/#
```

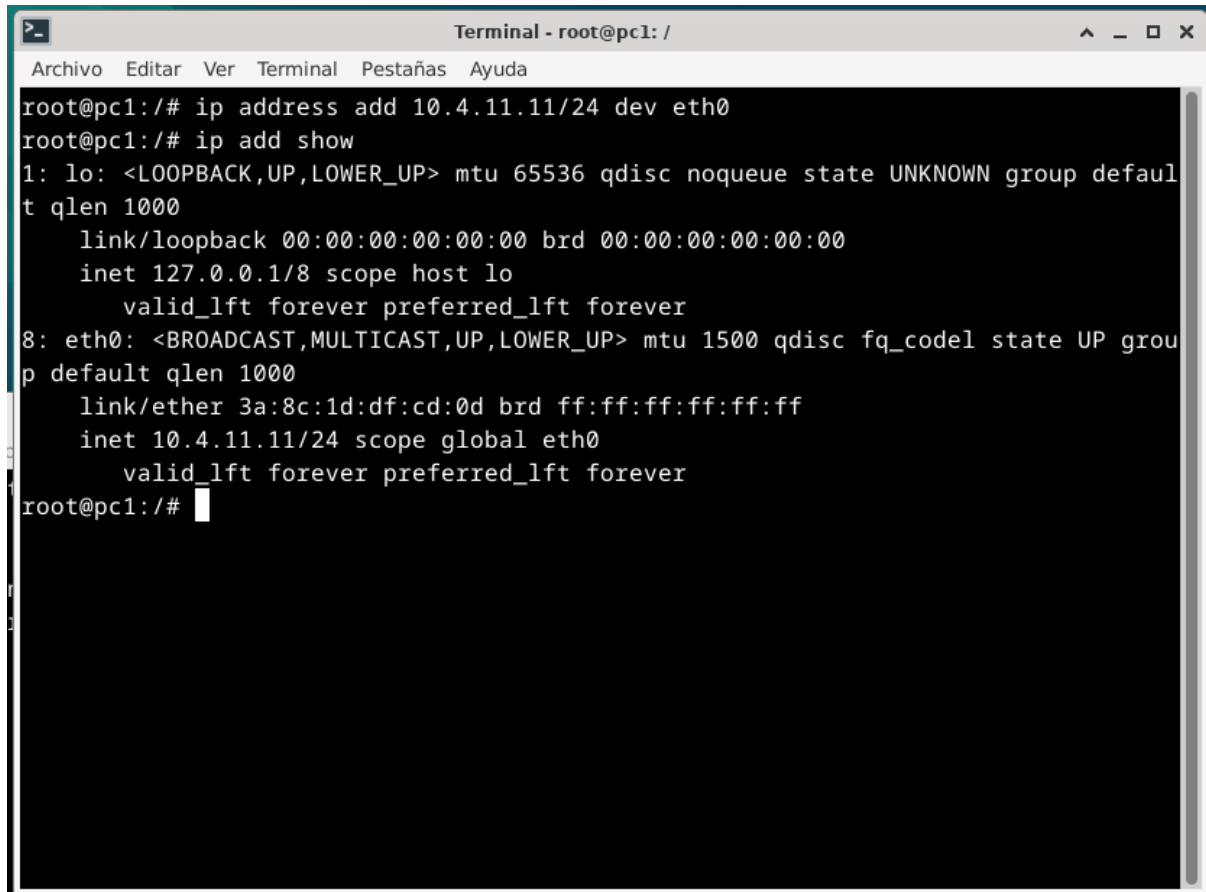
2. Configuración de interfaces de red para utilizar el protocolo TCP/IP. El paso siguiente es asignar las direcciones IP 10.4.11.11 y 10.4.11.12 a pc1 y pc2 respectivamente (la máscara de red es /24 o 255.255.255.0).

Para asignar las direcciones IP 10.4.11.11 y 10.4.11.12 a **pc1** y **pc2** respectivamente utilizo los siguientes comandos:

Ip addr add 10.4.11.11/24 dev eth0 (para pc1)

Ip addr add 10.4.11.12/24 dev eth0 (para pc2)

después verifico que se haya configurado correctamente con: *ip addr show*. De esta manera le asigno una ip a la interfaz eth0 de cada maquina para que puedan identificarse en la red.

A terminal window titled "Terminal - root@pc1: /" with a menu bar containing "Archivo", "Editar", "Ver", "Terminal", "Pestañas", and "Ayuda". The terminal shows the following commands and output:

```
root@pc1:/# ip address add 10.4.11.11/24 dev eth0
root@pc1:/# ip add show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
8: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 3a:8c:1d:df:cd:0d brd ff:ff:ff:ff:ff:ff
    inet 10.4.11.11/24 scope global eth0
        valid_lft forever preferred_lft forever
root@pc1:/#
```

3. Verificar que es posible contactar ambos equipos de la red.

lo verifico con ping 10.4.11.11, desde pc2 a pc1 y, después con ping 10.4.11.12 desde pc1 hacia pc2.

```
Terminal - root@pc1: /
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda

root@pc1:/# ip address add 10.4.11.11/24 dev eth0
root@pc1:/# ip add show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
8: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 3a:8c:1d:df:cd:0d brd ff:ff:ff:ff:ff:ff
    inet 10.4.11.11/24 scope global eth0
        valid_lft forever preferred_lft forever
root@pc1:/# ping -c 4 10.4.11.12
PING 10.4.11.12 (10.4.11.12) 56(84) bytes of data.
64 bytes from 10.4.11.12: icmp_seq=1 ttl=64 time=75.9 ms
64 bytes from 10.4.11.12: icmp_seq=2 ttl=64 time=0.230 ms
64 bytes from 10.4.11.12: icmp_seq=3 ttl=64 time=0.246 ms
64 bytes from 10.4.11.12: icmp_seq=4 ttl=64 time=0.252 ms

--- 10.4.11.12 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3037ms
rtt min/avg/max/mdev = 0.230/19.159/75.910/32.764 ms
root@pc1:/#
```

Ahora pruebo la conexión desde pc2 hacia pc1

```
Terminal - root@pc2: /
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda

root@pc2:/# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
9: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 1a:3a:15:c4:d4:26 brd ff:ff:ff:ff:ff:ff
    inet 10.4.11.12/24 scope global eth0
        valid_lft forever preferred_lft forever
root@pc2:/# ping -c 4 10.4.11.11
PING 10.4.11.11 (10.4.11.11) 56(84) bytes of data.
64 bytes from 10.4.11.11: icmp_seq=1 ttl=64 time=0.059 ms
64 bytes from 10.4.11.11: icmp_seq=2 ttl=64 time=0.039 ms
64 bytes from 10.4.11.11: icmp_seq=3 ttl=64 time=0.046 ms
64 bytes from 10.4.11.11: icmp_seq=4 ttl=64 time=0.058 ms

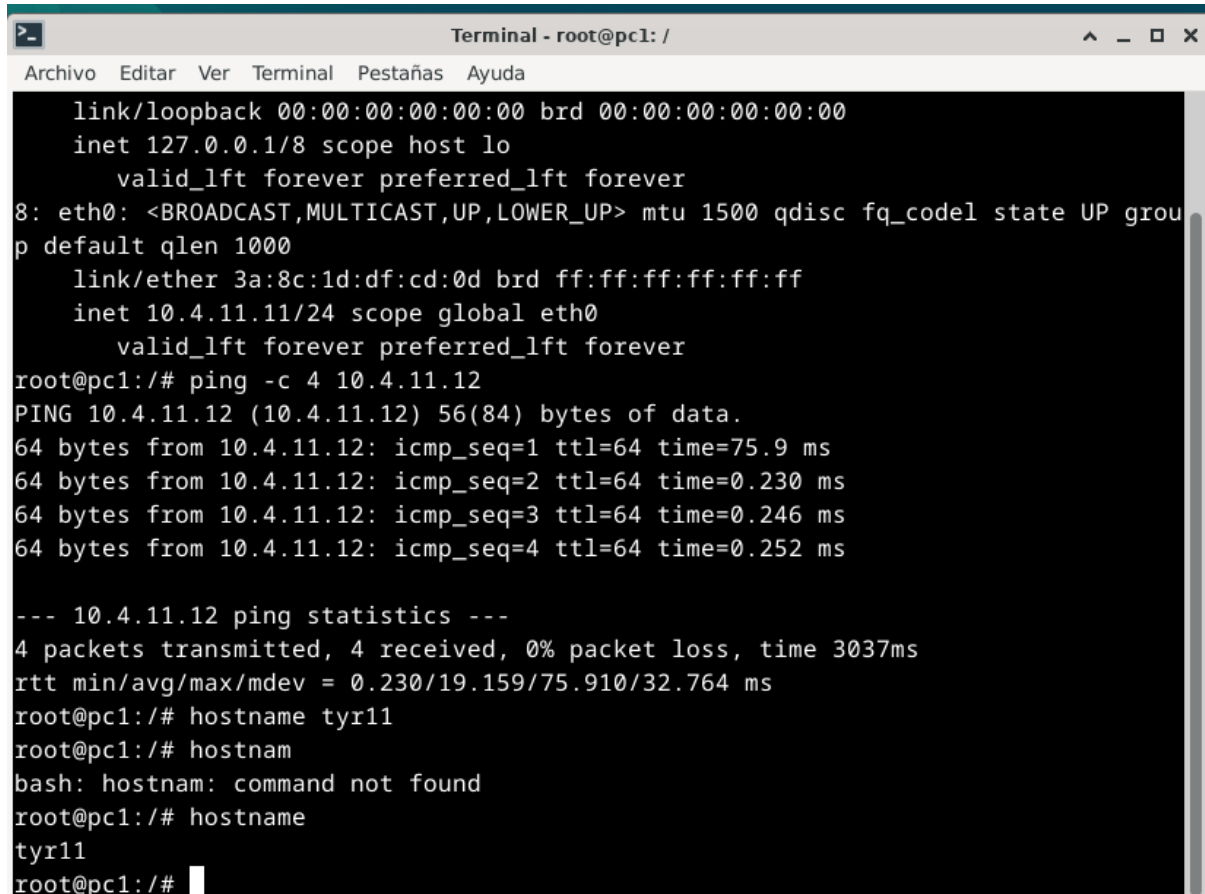
--- 10.4.11.11 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3063ms
rtt min/avg/max/mdev = 0.039/0.050/0.059/0.008 ms
root@pc2:/#
```

4. Cambiar la configuración de los nombres de los equipos asignándoles tyr11 y tyr12 a pc1 y pc2 respectivamente.

Hacemos uso del comando: *hostname* para asignarle nombres a los equipos:

```
hostname tyr11
```

```
hostname tyr12
```



```
Terminal - root@pc1: /
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
    valid_lft forever preferred_lft forever
8: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
p default qlen 1000
    link/ether 3a:8c:1d:df:cd:0d brd ff:ff:ff:ff:ff:ff
    inet 10.4.11.11/24 scope global eth0
        valid_lft forever preferred_lft forever
root@pc1:/# ping -c 4 10.4.11.12
PING 10.4.11.12 (10.4.11.12) 56(84) bytes of data.
64 bytes from 10.4.11.12: icmp_seq=1 ttl=64 time=75.9 ms
64 bytes from 10.4.11.12: icmp_seq=2 ttl=64 time=0.230 ms
64 bytes from 10.4.11.12: icmp_seq=3 ttl=64 time=0.246 ms
64 bytes from 10.4.11.12: icmp_seq=4 ttl=64 time=0.252 ms

--- 10.4.11.12 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3037ms
rtt min/avg/max/mdev = 0.230/19.159/75.910/32.764 ms
root@pc1:/# hostname tyr11
root@pc1:/# hostnam
bash: hostnam: command not found
root@pc1:/# hostname
tyr11
root@pc1:/#
```

5. Resolución de nombres de hosts a direcciones IP.

a. Configurar la resolución de nombres locales en ambos hosts con la información contenida en el punto 4.

```
sudo nano /etc/hosts
```

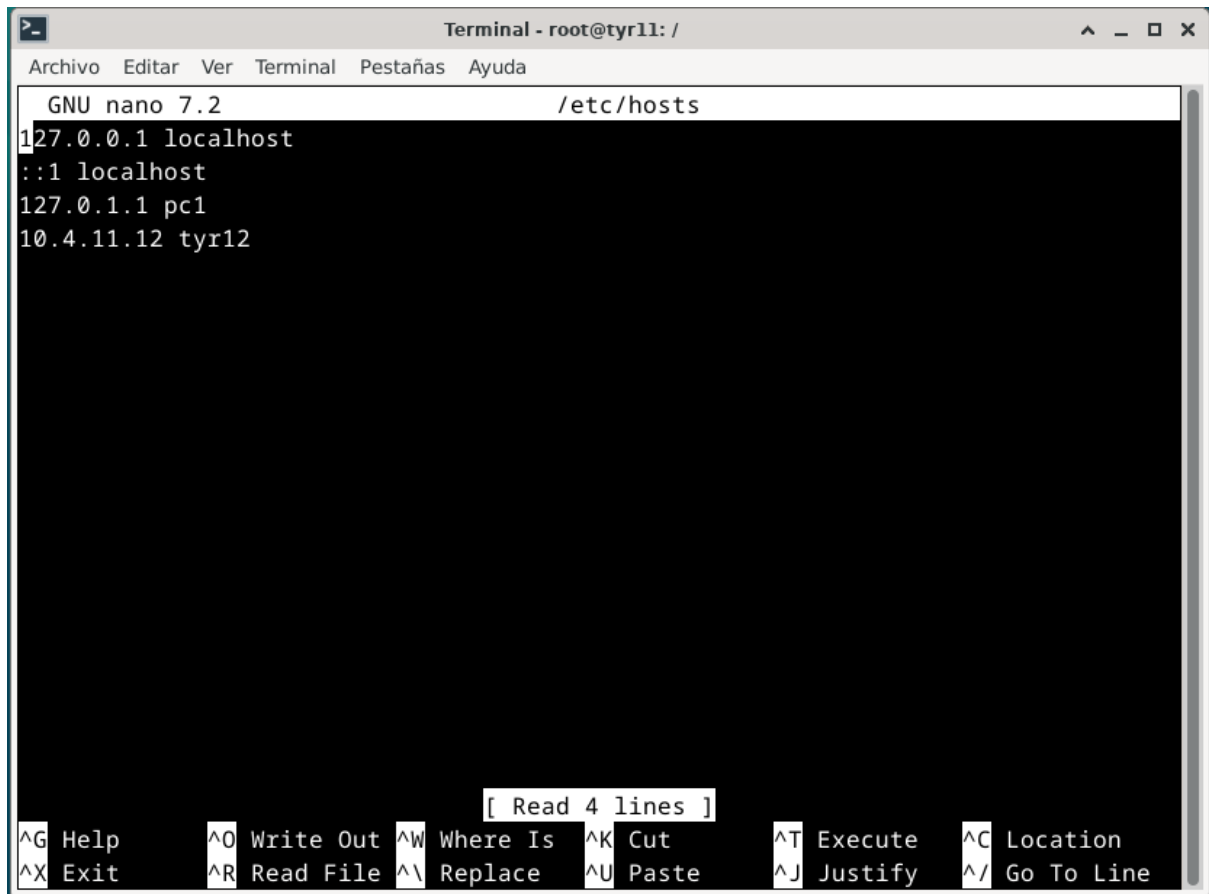
b. Verificar que es posible contactar ambos equipos de la red utilizando nombres de host.

a) para configurar la resolución de nombres locales en pc1 y pc2, hay que editar el archivo de configuración de hosts del sistema (de cada uno, respectivamente) y agregar:

```
10.4.11.11 tyr11 (en el archivo de pc2)
```

```
10.4.11.12 tyr12 (en el archivo de pc1)
```

Para abrir el archivo y agregar esas líneas se usa el comando:
sudo nano /etc/hosts



```
Terminal - root@tyr11: /
GNU nano 7.2 /etc/hosts
1 127.0.0.1 localhost
2 ::1 localhost
3 127.0.1.1 pc1
4 10.4.11.12 tyr12
[ Read 4 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line
```

b) Para verificar que es posible conectar ambos equipos. Hago un ping tyr11 desde pc2, y después un ping tyr12 desde pc1.

```
Terminal - root@tyr11: /
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
root@tyr11:/# sudo nano /etc/hosts
bash: sudo: command not found
root@tyr11:/# nano /etc/hosts
root@tyr11:/# nano /etc/hosts
root@tyr11:/# nano /etc/hosts
root@tyr11:/# ping -c 4 tyr12
PING tyr12 (10.4.11.12) 56(84) bytes of data.
64 bytes from tyr12 (10.4.11.12): icmp_seq=1 ttl=64 time=0.398 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=2 ttl=64 time=0.309 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=3 ttl=64 time=0.411 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=4 ttl=64 time=0.314 ms

--- tyr12 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3066ms
rtt min/avg/max/mdev = 0.309/0.358/0.411/0.046 ms
root@tyr11:/#
```

6. Ver la tabla de ruteo definida en el equipo. ¿Cuáles son las redes accesibles?

Uso el comando *ip route*:

```
Terminal - root@tyr11: /
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
root@tyr11:/# sudo nano /etc/hosts
bash: sudo: command not found
root@tyr11:/# nano /etc/hosts
root@tyr11:/# nano /etc/hosts
root@tyr11:/# nano /etc/hosts
root@tyr11:/# ping -c 4 tyr12
PING tyr12 (10.4.11.12) 56(84) bytes of data.
64 bytes from tyr12 (10.4.11.12): icmp_seq=1 ttl=64 time=0.398 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=2 ttl=64 time=0.309 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=3 ttl=64 time=0.411 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=4 ttl=64 time=0.314 ms

--- tyr12 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3066ms
rtt min/avg/max/mdev = 0.309/0.358/0.411/0.046 ms
root@tyr11:/# ip route
10.4.11.0/24 dev eth0 proto kernel scope link src 10.4.11.11
root@tyr11:/#
```


Las redes accesibles directamente son las locales a través de la interfaz eth0

La salida del comando `ip route` indica la siguiente ruta en la tabla de enrutamiento:

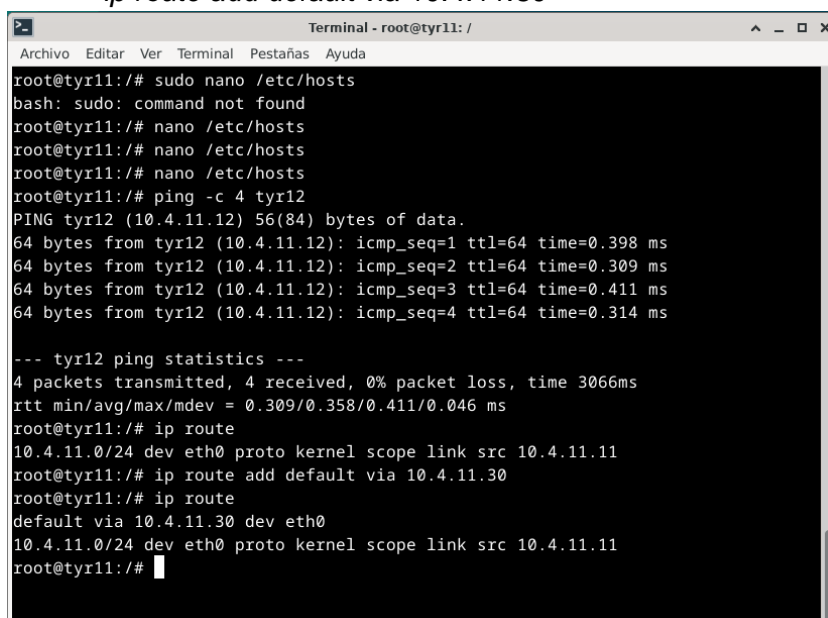
- **10.4.11.0/24**: Esta es la red de destino, representada en formato CIDR. La notación "/24" indica que se trata de una subred con una máscara de red de 24 bits, lo que equivale a una máscara de red de 255.255.255.0. Esta subred incluye todas las direcciones IP que comparten los primeros 24 bits, lo que significa que incluye direcciones desde 10.4.11.0 hasta 10.4.11.255.
- **dev eth0**: Esto indica la interfaz de red a través de la cual se puede alcanzar la red de destino. En este caso, la interfaz se llama `eth0`
- **proto kernel**: Significa que esta ruta fue añadida al kernel por algún proceso del kernel, como parte de la configuración de red automática o dinámica.
- **scope link**: Indica el alcance de la ruta. "link" significa que la ruta es válida solo para el enlace local (la misma subred), por lo tanto, los paquetes destinados a direcciones en esta red serán enviados directamente a través de la interfaz especificada.
- **src 10.4.11.11**: Indica la dirección IP de origen que se utilizará cuando se envíen paquetes a esta red. En este caso, la dirección IP de origen es 10.4.11.12, lo que significa que los paquetes se enviarán desde esta dirección cuando se comuniquen con hosts en la red de destino.

En resumen, la red 10.4.11.0/24 es accesible a través de la interfaz `eth0` en tu sistema, utilizando la dirección IP 10.4.11.11 como la dirección de origen cuando se envían paquetes a esa red.

7. Agregar la dirección 10.4.11.30 como ruta por defecto para acceder a otras redes. Verificar nuevamente la tabla de ruteo.

Para agregar un default Gateway uso el comando:

`ip route add default via 10.4.11.30`



```
Terminal - root@tyr11: /
Archivo Editar Ver Terminal Pestañas Ayuda
root@tyr11:/# sudo nano /etc/hosts
bash: sudo: command not found
root@tyr11:/# nano /etc/hosts
root@tyr11:/# nano /etc/hosts
root@tyr11:/# nano /etc/hosts
root@tyr11:/# ping -c 4 tyr12
PING tyr12 (10.4.11.12) 56(84) bytes of data:
64 bytes from tyr12 (10.4.11.12): icmp_seq=1 ttl=64 time=0.398 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=2 ttl=64 time=0.309 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=3 ttl=64 time=0.411 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=4 ttl=64 time=0.314 ms

--- tyr12 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3066ms
rtt min/avg/max/mdev = 0.309/0.358/0.411/0.046 ms
root@tyr11:/# ip route
10.4.11.0/24 dev eth0 proto kernel scope link src 10.4.11.11
root@tyr11:/# ip route add default via 10.4.11.30
root@tyr11:/# ip route
default via 10.4.11.30 dev eth0
10.4.11.0/24 dev eth0 proto kernel scope link src 10.4.11.11
root@tyr11:/#
```

Si no agregamos una ruta por defecto entonces la única red que estaría directamente accesible es la red **10.4.11.0/24** a través de la interfaz **eth0**. La adición del default gateway permite el acceso a otras redes externas a través de la puerta de enlace especificada.

8. Realizar una captura de las PDU intercambiadas mientras se utiliza el comando ping para verificar conectividad con el otro equipo. Las acciones que debe realizar son:

- a. En el dispositivo capturador inicie la captura utilizando el comando tcpdump o tshark sobre la interfaz eth0 y redirigir la salida a un archivo en el directorio /shared para su posterior análisis.

- b. En pc1 ejecutar el comando ping para enviar a pc2 exactamente 3 mensajes ICMP Echo Request (consulte el manual de ping).

- c. Una vez obtenida la respuesta del comando ping (deberán recibirse tres respuestas), detener la captura (finalizar el proceso tcpdump o tshark presionando Ctrl+C)

- d. Analizar el volcado del programa de captura utilizando la aplicación wireshark (o cualquier otro analizador de tráfico que permita leer archivos en formato pcap), representando en un gráfico ideado por usted el intercambio de mensajes. Indicar cuál es la función de cada uno identificando los datos de encabezados mas relevantes.

Inicio la captura con `tshark -i eth0 -w /shared/captura.pcap` y hago un ping desde `tyr11` a `tyr12`, después detengo la captura con `ctrl+c`

```
Terminal - root@tyr11: /
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
64 bytes from tyr12 (10.4.11.12): icmp_seq=3 ttl=64 time=0.325 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=4 ttl=64 time=0.313 ms

--- tyr12 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3031ms
rtt min/avg/max/mdev = 0.233/0.277/0.325/0.042 ms
root@tyr11:/# ping -c 3 tyr12
PING tyr12 (10.4.11.12) 56(84) bytes of data.
64 bytes from tyr12 (10.4.11.12): icmp_seq=1 ttl=64 time=0.246 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=2 ttl=64 time=0.225 ms
64 bytes from tyr12 (10.4.11.12): icmp_seq=3 ttl=64 time=0.293 ms

--- tyr12 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2036ms
rtt min/avg/max/mdev = 0.225/0.254/0.293/0.028 ms
root@tyr11:/#
```

```
Terminal - root@capturador: /
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
root@capturador:/# tshark -i eth0 -w /shared/captura.pcap
Running as user "root" and group "root". This could be dangerous.
Capturing on 'eth0'
** (tshark:62) 19:55:35.687663 [Main MESSAGE] -- Capture started.
** (tshark:62) 19:55:35.689670 [Main MESSAGE] -- File: "/shared/captura.pcap"
12 ^C
tshark:
root@capturador:/# tshark -i eth0 -w /shared/captura.pcap
Running as user "root" and group "root". This could be dangerous.
Capturing on 'eth0'
** (tshark:84) 20:03:27.375765 [Main MESSAGE] -- Capture started.
** (tshark:84) 20:03:27.376667 [Main MESSAGE] -- File: "/shared/captura.pcap"
10 ^C
tshark:
root@capturador:/#
```

Analizamos la captura

captura.pcap

ArchivoEdiciónVisualizaciónIrCapturaAnalizarEstadísticasTelefoníaWirelessHerramientasAyuda

Aplique un filtro de visualización ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.4.11.11	10.4.11.12	ICMP	98	Echo (ping) request id=0x0006, seq=1/256, ttl=64 (reply in 2)
2	0.000000000	10.4.11.12	10.4.11.11	ICMP	98	Echo (ping) reply id=0x0006, seq=1/256, ttl=64 (request in 1)
3	1.011261256	10.4.11.11	10.4.11.12	ICMP	98	Echo (ping) request id=0x0006, seq=2/512, ttl=64 (reply in 4)
4	1.011345158	10.4.11.12	10.4.11.11	ICMP	98	Echo (ping) reply id=0x0006, seq=2/512, ttl=64 (request in 3)
5	2.035599002	10.4.11.11	10.4.11.12	ICMP	98	Echo (ping) request id=0x0006, seq=3/768, ttl=64 (reply in 6)
6	2.035677924	10.4.11.12	10.4.11.11	ICMP	98	Echo (ping) reply id=0x0006, seq=3/768, ttl=64 (request in 5)
7	5.139549762	3a:8c:1d:df:cd:0d	1a:3a:15:c4:d4:26	ARP	60	Who has 10.4.11.12? Tell 10.4.11.11
8	5.139619864	1a:3a:15:c4:d4:26	3a:8c:1d:df:cd:0d	ARP	60	Who has 10.4.11.11? Tell 10.4.11.12
9	5.139680615	3a:8c:1d:df:cd:0d	1a:3a:15:c4:d4:26	ARP	60	10.4.11.11 is at 3a:8c:1d:df:cd:0d
10	5.139739036	1a:3a:15:c4:d4:26	3a:8c:1d:df:cd:0d	ARP	60	10.4.11.12 is at 1a:3a:15:c4:d4:26

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface eth0, Internet II, Src: 3a:8c:1d:df:cd:0d (3a:8c:1d:df:cd:0d), Dst: 1a:3a:15:c4:d4:26 (1a:3a:15:c4:d4:26), Internet Protocol Version 4, Src: 10.4.11.11, Dst: 10.4.11.12, Internet Control Message Protocol

00001a3a15c4d4263a8c1ddfcd0d08004500

0001005401a340000048010e080a0400000a04

0002000c000009b2000600019d00c7670000

000300000000000000000000101112131415

0004161718191a1b1c1d1e1f202122232425

0005262728292a2b2c2d2e2f303132333435

00063637

.....E
T:0:0
.....0
.....
.....
.....
.....
.....

captura.pcap

Paquetes: 10 - Mostrado: 10 (100.0%)

Perfil: Default

10.4.11.11

10.4.11.12

ARP Who has 10.4.11.12?

ARP Reply: 10.4.11.12 is at

ICMP Echo Request (seq=1)

ICMP Echo Reply (seq=1)

ICMP Echo Request (seq=2)

ICMP Echo Reply (seq=2)

ICMP Echo Request (seq=3)

ICMP Echo Reply (seq=3)

Análisis del tráfico capturado

La captura muestra el intercambio de paquetes ICMP y ARP entre **10.4.11.11** y **10.4.11.12**.

- **ICMP Echo Request (ping request)**: Mensajes enviados por **10.4.11.11** a **10.4.11.12** para comprobar conectividad.
- **ICMP Echo Reply (ping reply)**: Respuesta de **10.4.11.12** confirmando la comunicación exitosa.
- **ARP "Who has"**: Consultas para obtener la dirección MAC de la IP de destino antes de enviar los paquetes ICMP.
- **ARP "Is at"**: Respuestas indicando la dirección MAC correspondiente a la IP consultada.

•

Datos clave de los encabezados

Encabezado de Ethernet II

- **Src MAC**: 3a:8c:1d:df:cd:0d (origen)
- **Dst MAC**: 1a:3a:15:c4:d4:26 (destino)
- **Tipo**: 0x0800 (IPv4) o 0x0806 (ARP)

Encabezado IP

- **Source IP**: 10.4.11.11
- **Destination IP**: 10.4.11.12
- **TTL**: 64
- **Protocolo**: ICMP (1)

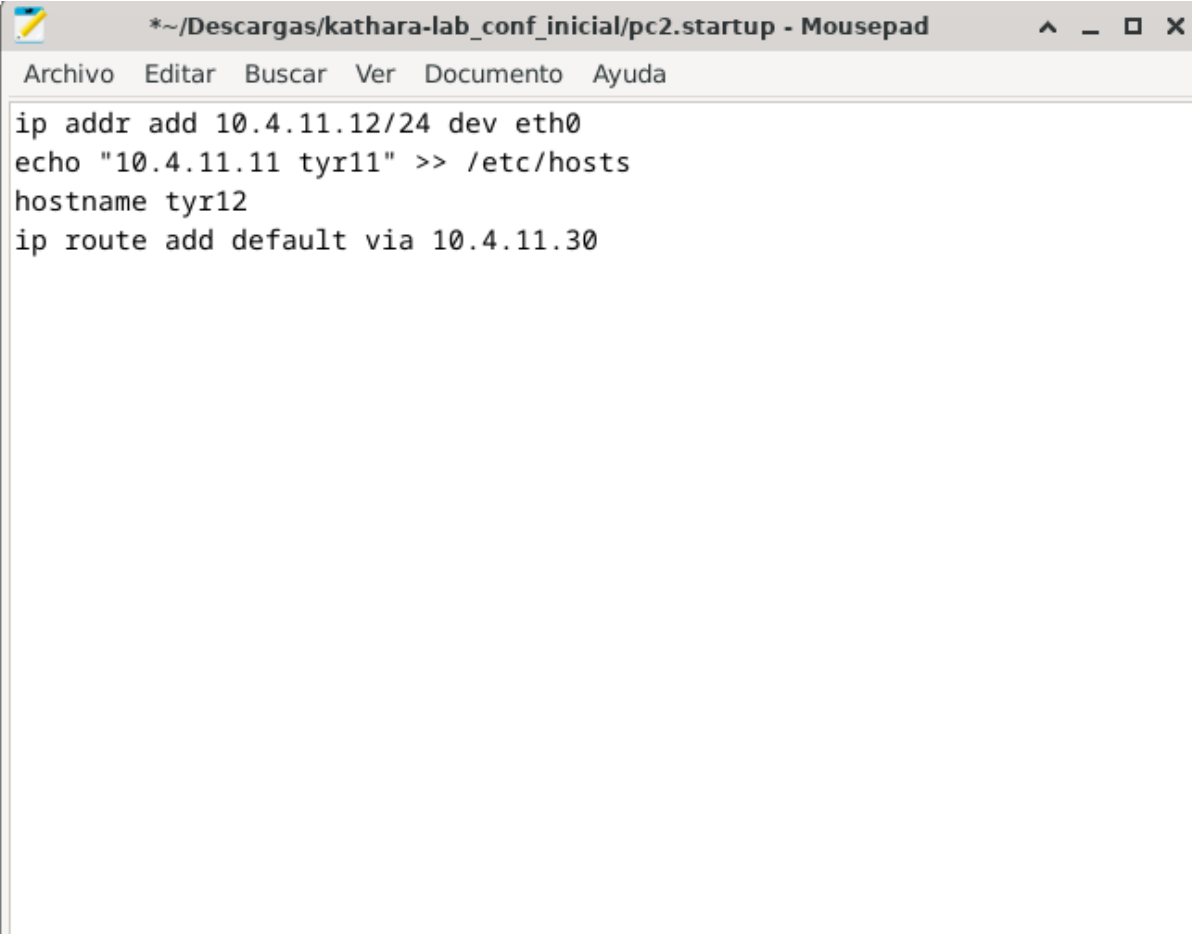
Encabezado ICMP

- **Tipo**: 8 (Echo Request) / 0 (Echo Reply)
- **Código**: 0
- **Identificador**: 0x0006
- **Número de secuencia**: 1, 2, 3

Paso a paso del proceso:

1. **Resolución ARP**: 10.4.11.11 pregunta por la MAC de 10.4.11.12 y obtiene la respuesta.
2. **Envío de paquetes ICMP**: Se envían tres ping request.
3. **Recepción de respuestas**: 10.4.11.12 responde con tres ping reply.
4. **Confirmación de conectividad**: Los ping son exitosos, lo que indica que los equipos están conectados correctamente.

9. Escribir los comandos de configuración que ejecutó en los puntos 2, 4, 5 y 7 en pc1 y pc2 a los archivos pc1.startup y pc2.startup respectivamente, que están dentro del directorio del laboratorio, de manera tal que los nodos se configuren automáticamente al reiniciar el laboratorio.



The image shows a screenshot of a text editor window titled "*/~/Descargas/kathara-lab_conf_inicial/pc2.startup - Mousepad". The window has a menu bar with "Archivo", "Editar", "Buscar", "Ver", "Documento", and "Ayuda". The text area contains the following commands:

```
ip addr add 10.4.11.12/24 dev eth0
echo "10.4.11.11 tyr11" >> /etc/hosts
hostname tyr12
ip route add default via 10.4.11.30
```

Pruebo que esto funcione con:

Ip route

Ip a s

Hostname

Cat /etc/hosts

Recursos

- Resumen de comandos: <http://bit.ly/tyr-comandos>
- Semestre Perdido. Linea de Comandos: <http://bit.ly/tyr-ms-cli> (versión traducida por el equipo de TyR)

- Semetre Perdido. Shell: <http://bit.ly/tyr-ms-shell> (versión traducida por el equipo de TyR)
- Manual de uso de Kathará <https://www.kathara.org/man-pages/kathara.1.html>