



I. GESTIÓN DE REDES EN LINUX

II. OBJETIVOS

Al finalizar la presente práctica el estudiante:

1. Conoce los mecanismos de comunicación de redes TCP/IP en Linux.
2. Configura una red de computadores en Linux y Windows utilizando IPv6
3. Comparte carpetas con el sistema operativo Windows.
4. Comparte archivos utilizando protocolos SMB y NFS.

III. PRE-REQUISITOS

Para un mejor resultado, es recomendable que, para realizar esta práctica, el estudiante previamente:

1. Conozca a nivel básico la operación del sistema operativo Linux
2. Conozca los aspectos teóricos básicos de una red de computadores TCP/IP

IV. MATERIALES Y EQUIPOS.

Los materiales que utilizaremos en los trabajos de laboratorio son:

1. Sistema operativo Linux (La presente guía se desarrolló utilizando Debian. Si utiliza otra distribución, deberá documentarse sobre las particularidades de la misma)
2. Cable UTP cruzado
3. Computador con sistema operativo Windows 7 o superior.



V. MARCO TEORICO.

REDES TCP/IP

La tecnología de redes TCP/IP es la base de Internet y en la actualidad es el estándar de facto para el desarrollo de aplicaciones de red.

COMANDOS PARA LA GESTION DE REDES EN LINUX

En Linux, existen diversas herramientas que nos pueden ayudar en la configuración de los elementos de una red, desde su interfaz, a servicios que pueden ejecutarse en un equipo servidor.

Para la gestión básica de red podemos utilizar, desde la línea de comandos los comandos `ifconfig`/`ip`/`route`/`iwcd`, etc.

ASIGNACION DE DIRECCION IPv4 A LA INTERFAZ DE UN HOST

Por ejemplo, para agregar una dirección IP a una interfaz de red, podemos utilizar el comando:

```
ifconfig eth0 add 192.168.1.30 netmask 255.255.255.0 up
```

La misma funcionalidad se puede conseguir con el comando:

```
ip a add 192.168.1.30/24 dev eth0
```

CONFIGURACIÓN DE LA PUERTA DE ENLACE (GATEWAY)

Para definir la ruta de salida de la red a la que pertenece un equipo, utilizamos el comando

```
route add default gw 192.168.1.1 eth0
```

El comando `route` utilizado específica que cuando un paquete este destinado a una red distinta a la local, este se envía al equipo cuya dirección IP es 192.168.1.1 y que está conectado al equipo local a través del puerto `eth0` de este último.

CONFIGURACIÓN DEL SERVICIO DE NOMBRES DE DOMINIO (DNS)

Para que el equipo local pueda conectarse a equipos remotos, cuyas direcciones IP desconocemos, pero cuyos nombres si conocemos, se debe utilizar un servidor de nombres de dominio o DNS (por sus siglas en inglés).



UNIVERSIDAD ANDINA DEL CUSCO

ADMINISTRACIÓN DE SERVICIOS DE REDES

GUÍA DE LABORATORIO

ECP 3 de 15

La configuración del servidor de nombres de dominio se puede hacer mediante la edición directa del archivo /etc/resolv.conf, el cual se utiliza para indicar la dirección IP del servidor de nombres de dominio como se aprecia en la siguiente figura:

```
GNU nano 2.2.6      Fichero: /etc/resolv.conf      Modificado
nameserver 200.48.225.130 ■
```

Se puede conseguir el mismo resultado si, desde la línea de comandos, utilizamos el siguiente comando:

```
echo 'nameserver 200.48.225.130' > /etc/resolv.conf
```

Este comando escribe en el archivo resolv.conf la cadena

```
nameserver 200.48.225.130
```

SERVICIO NETWORK MANAGER

Para facilitar la configuración de red, se utiliza el servicio Network Manager, que ajusta los parámetros de red como la dirección IP, el Gateway, el DNS y otros de forma automática.

Una consecuencia de la configuración automática, es que los parámetros de red pueden cambiar y afectar la provisión de servicios de red.

Es recomendable desactivar el servicio Network Manager y gestionar manualmente los parámetros de red, mas aun, si estamos configurando un servidor de red.

Para detener el servicio Network Manager ejecutamos:

```
/etc/init.d/network-manager stop
```

Para desactivar completamente Network Manager ejecutamos la orden:

```
systemctl stop NetworkManager.service
```

y

```
systemctl disable NetworkManager.service
```

Para que los parámetros de configuración de la red no cambien, podemos registrarlos en el archivo interfaces, que se encuentra en la carpeta /etc/network/.



En el archivo interfaces guardamos los parámetros de red como se muestra en el siguiente texto:

```
source /etc/network/interfaces.d/*  
  
#The loopback network interface  
auto lo  
iface lo inet loopback  
  
#The primary network interface  
auto eth0  
iface eth0 inet static  
    address 192.168.1.254  
    netmask 255.255.255.0  
    gateway 192.168.1.1  
    nameserver 8.8.8.8
```

COMANDOS PARA EL MONITOREO Y DIAGNOSTICO DE REDES LINUX

Para verificar la conectividad entre un par de equipos, es suficiente utilizar el comando ping, cuya sintaxis se muestra a continuación:

```
ping <dirección IPv4>
```

Por ejemplo, para probar la conectividad desde un equipo de la red 192.168.1.0/24 al gateway 192.168.1.1, debemos utilizar el comando:

```
ping 192.168.1.1
```

Si la conexión entre el gateway y el equipo de red está operativa, el resultado de realizar la consulta será:

```
root@base:~# ping 192.168.1.1  
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.  
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.565 ms  
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.873 ms  
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.895 ms  
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=0.770 ms  
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=0.849 ms  
^C  
--- 192.168.1.1 ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 22ms  
rtt min/avg/max/mdev = 0.565/0.790/0.895/0.122 ms
```



ASIGNACION DE DIRECCION IPv6 A UNA INTERFAZ DE HOST

Antes de asignar la dirección IPv6 a un host (más específicamente, a una interfaz), se debe generar la dirección IPv6 siguiendo los siguientes pasos:

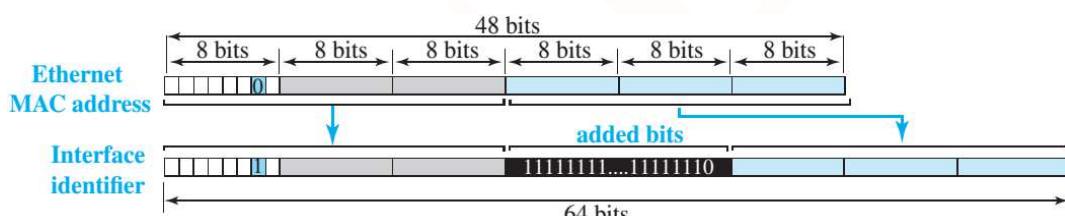
1. Obtener la dirección MAC de la interfaz

Como ejemplo, tomaremos la siguiente dirección MAC:

08:00:27:0A:2C:C5

2. Formar el identificador de interfaz

El identificador de interfaz se obtiene aplicando la siguiente regla, expresada gráficamente:



Escribimos la dirección MAC como un número binario:

0	8	0	0	2	7	0	A	2	C	C	5
0000	1000	0000	0000	0010	0111	0000	1010	0010	1100	1100	0101

Cambiamos el segundo bit menos significativo del byte más significativo de 0 a 1 e insertamos el número hexadecimal FFFE a la mitad de la dirección de 48 bits. Como resultado obtenemos el identificador de interfaz de 64 bits:

0	A	0	0	2	7	F	F	F	E	0	A	2	C	C	5
0000	1010	0000	0000	0010	0111	1111	1111	1111	1110	0000	0101	0010	1100	1100	0101

0A00:27FF:FE0A:2CC5

De forma abreviada, eliminando los ceros a la izquierda, este identificador puede escribirse así:

A00:27FF:FE0A:2CC5

3. Determinar el prefijo de enrutamiento global (GRP):

El GRP es asignado por IANA (Internet Assigned Numbers Authority) o dependencias de jerarquía inferior (RIR, NIR, ISP/LIR), del bloque 2000::/3 como se muestra en la tabla:



UNIVERSIDAD ANDINA DEL CUSCO
ADMINISTRACIÓN DE SERVICIOS DE REDES
GUÍA DE LABORATORIO

ECP 6 de 15

Block prefix	CIDR	Block assignment	Fraction
0000 0000	0000::/8	Special addresses	1/256
001	2000::/3	Global unicast	1/8
1111 1110	FC00::/7	Unique local unicast	1/128
1111 1110 10	FE80::/10	Link local addresses	1/1024
1111 1111	FF00::/8	Multicast addresses	1/256

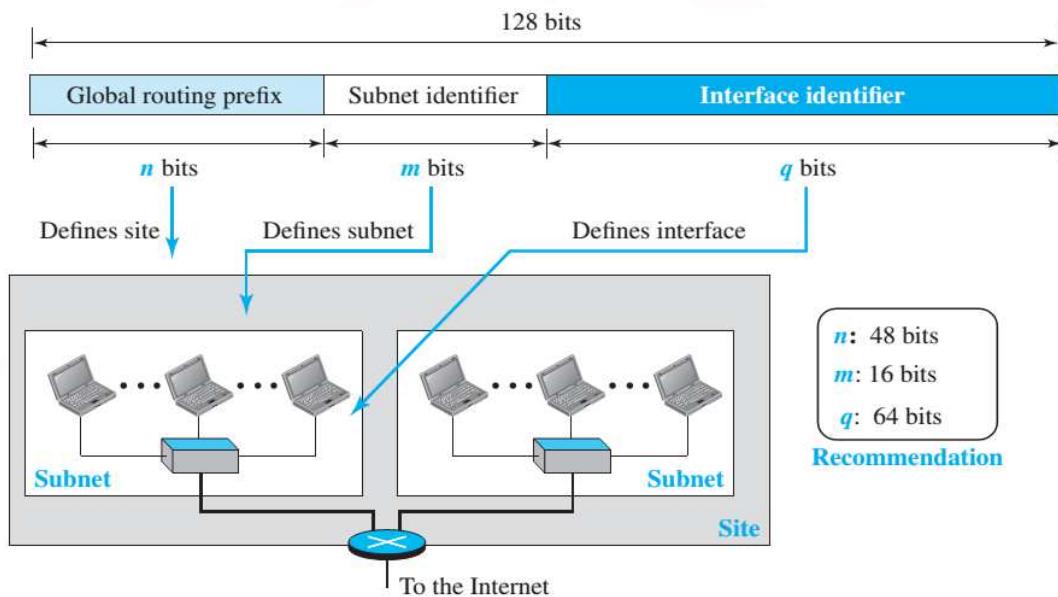
Como ejemplo usaremos el GRP de 48 bits:

2134:A0B1:C2D3

4. La dirección de subred se define localmente dentro de la organización, de acuerdo a sus necesidades. Para el ejemplo usaremos la subred (de 16 bits):

E4F5

5. Concatenar el prefijo de enrutamiento global con la dirección de subred y el identificador de interfaz de acuerdo al siguiente diagrama:



Al concatenar estos tres componentes, obtenemos la dirección IPv6 de la interfaz de red:

2134:A0B1:C2D3:E4F5:A00:27FF:FE0A:2CC5

La máscara será igual a la longitud del prefijo de red y el identificador de subred:

$$48 + 16 = 64$$

Entonces la dirección, incluyendo la máscara, en notación CIDR será:

2134:A0B1:C2D3:E4F5:A00:27FF:FE0A:2CC5/64



UNIVERSIDAD ANDINA DEL CUSCO
ADMINISTRACIÓN DE SERVICIOS DE REDES
GUÍA DE LABORATORIO

ECP 7 de 15

6. Una vez obtenida la dirección IPv6, asignamos la misma a la interfaz de red mediante el comando:

```
ifconfig <Interfaz de red> inet6 add <Dirección IPv6>/<Máscara CIDR> up
```

para el ejemplo:

```
ifconfig eth0 inet6 add 2134:A0B1:C2D3:E4F5:A00:27FF:FE0A:2CC5/64 up
```

finalmente, verificamos la dirección IPv6 asignada:

```
demo@antix1:~  
$ ifconfig  
eth0: flags=28605<UP,BROADCAST,RUNNING,MULTICAST,DYNAMIC> mtu 1500  
      inet 192.168.1.14 netmask 255.255.255.0 broadcast 192.168.1.255  
      inet6 2134:a0b1:c2d3:e4f5:a00:27ff:fe0a:2cc5 prefixlen 64 scopeid 0x0<global>  
      inet6 fe80::a0b1:c2d3:e4f5:a00:27ff:fe0a:2cc5 prefixlen 64 scopeid 0x20<link>  
      ether 08:00:27:0a:2c:c5 txqueuelen 1000 (Ethernet)  
      RX packets 667 bytes 50186 (49.0 KiB)  
      RX errors 0 dropped 0 overruns 0 frame 0  
      TX packets 416 bytes 37928 (37.0 KiB)  
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Podemos ver que, efectivamente, se asignó la dirección:

```
2134:A0B1:C2D3:E4F5:A00:27FF:FE0A:2CC5/64
```



VI. PRACTICAS DE LABORATORIO.

1. Asigne una dirección IP del rango 192.168.1.0/24 a dos computadores con el sistema operativo Linux y verifique la conectividad entre ellos.

SOLUCIÓN

- a) Asignamos la dirección IP a PC01 mediante el comando:

```
ifconfig eth0 192.168.1.2 netmask 255.255.255.0 up
```

alternativamente, puede utilizarse el comando `ip` con la siguiente sintaxis:

```
ip a add 192.168.1.2 dev eth0
```

- b) Verificamos que la dirección IP se haya asignado correctamente mediante el comando

```
ifconfig
```

```
root@dog:~# ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:15:a0:f0
          inet addr:192.168.1.2 Bcast:192.168.1.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe15:a0f0/64 Scope:Link
                  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                  RX packets:591 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:52 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:1000
                  RX bytes:52041 (50.8 KiB) TX bytes:4951 (4.8 KiB)
```

O con el comando

```
ip a
```

- c) Siguiendo el mismo procedimiento, asignamos la dirección IP 192.168.1.3/255.255.255.0 al segundo equipo (PC02)

```
ifconfig eth0 192.168.1.3 netmask 255.255.255.0 up
```

```
user: bash - Konsole
Archivo Editar Ver Marcadores Preferencias Ayuda
root@jessie:/home/user# ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:b8:8c:97
          inet addr:192.168.1.3 Bcast:192.168.1.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:feb8:8c97/64 Scope:Link
                  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                  RX packets:66 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:59 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:1000
                  RX bytes:5321 (5.1 KiB) TX bytes:8302 (8.1 KiB)
```



UNIVERSIDAD ANDINA DEL CUSCO
ADMINISTRACIÓN DE SERVICIOS DE REDES
GUÍA DE LABORATORIO

ECP 9 de 15

- d) Para verificar la conectividad entre los equipos utilizamos el comando ping. Por ejemplo, para verificar la conectividad desde el segundo equipo (192.168.1.3):

```
ping 192.168.1.2
```

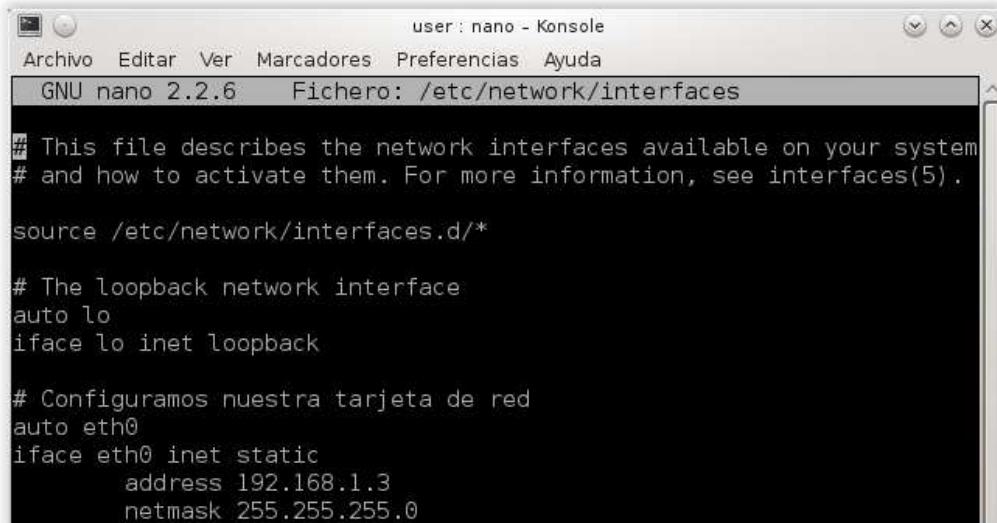


A screenshot of a terminal window titled "user : bash - Konsole". The window shows the command "root@jessie:/home/user# ping 192.168.1.2" followed by several lines of output from the ping command. The output shows 64 bytes being sent to 192.168.1.2 with TTL=64 and times ranging from 0.563 ms to 0.739 ms.

```
user : bash - Konsole
Archivo Editar Ver Marcadores Preferencias Ayuda
root@jessie:/home/user# ping 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=0.563 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.565 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.739 ms
64 bytes from 192.168.1.2: icmp_seq=4 ttl=64 time=0.804 ms
64 bytes from 192.168.1.2: icmp_seq=5 ttl=64 time=0.561 ms
```

Si la respuesta no es similar a la mostrada, verifique nuevamente la dirección IP y la máscara asignada a las interfaces de red.

- e) Alternativamente, se puede asignar una dirección IP editando el archivo /etc/network/interfaces. La diferencia de esta alternativa es que cuando el equipo reinicie, tendrá la dirección IP establecida en este archivo, en tanto que, en la primera alternativa, luego de reiniciarse el equipo, la dirección IP asignada no se conservará.



A screenshot of a terminal window titled "user : nano - Konsole". The window shows the "/etc/network/interfaces" file being edited with the nano editor. The file contains configuration for the "lo" loopback interface and the "eth0" network interface, setting its IP address to 192.168.1.3 and netmask to 255.255.255.0.

```
user : nano - Konsole
Archivo Editar Ver Marcadores Preferencias Ayuda
GNU nano 2.2.6 Fichero: /etc/network/interfaces
#
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
#
source /etc/network/interfaces.d/*
#
# The loopback network interface
auto lo
iface lo inet loopback
#
# Configuramos nuestra tarjeta de red
auto eth0
iface eth0 inet static
    address 192.168.1.3
    netmask 255.255.255.0
```

Para activar el servicio de red utilizamos el comando:

```
/etc/init.d/networking reload
```



UNIVERSIDAD ANDINA DEL CUSCO

ADMINISTRACIÓN DE SERVICIOS DE REDES

GUÍA DE LABORATORIO

ECP 10 de 15

2. Conecte dos equipos mediante IPv6. El prefijo de red que debe utilizar es 2134:A0B1:C2D3 y la subred E4F5 . Verifique la conectividad entre los equipos.

SOLUCIÓN

Para asignar una dirección IP a la interfaz de un host, como, por ejemplo, a la interfaz enp0s3 debemos obtener la dirección MAC de la interfaz.

Utilizando el comando **ifconfig**, obtenemos la dirección MAC del equipo que configuraremos:

```
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.1.24 netmask 255.255.255.0 broadcast 192.168.1.255
      inet6 fe80::a00:27ff:fe75:cb0 prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:75:0c:b0 txqueuelen 1000 (Ethernet)
          RX packets 99 bytes 7074 (6.9 KiB)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 18 bytes 1942 (1.8 KiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
          RX packets 0 bytes 0 (0.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 0 bytes 0 (0.0 B)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

La dirección MAC es:

08:00:27:75:0C:B0

A partir de la dirección MAC, obtenemos el Identificador de Interfaz siguiendo la regla de obtención del identificador de interfaz:

MAC-48	0	8	0	0	2	7	7	5	0	C	B	0	
	0000	1000	0000	0000	0010	0111	0111	0101	0000	1100	1011	0000	
II-64	0	A	0	0	2	7	F	F	F	E	7	5	0
	0000	1010	0000	0000	0010	0111	1111	1111	1111	1110	0111	0101	0000

El identificador de interfaz resultante es:

0A00:27FF:FE75:0CB0

Que en forma abreviada se escribe como:

A00:27FF:FE75:CB0



UNIVERSIDAD ANDINA DEL CUSCO

ADMINISTRACIÓN DE SERVICIOS DE REDES

GUÍA DE LABORATORIO

ECP 11 de 15

Con este dato y las especificaciones sobre el prefijo de enrutamiento global y la subred, establecemos la dirección unicast del host en:

2134:A0B1:C2D3:E4F5:A00:27FF:FE75:CB0

La máscara será igual a la longitud del prefijo de red y el identificador de subred:

$$48 + 16 = 64$$

Entonces la dirección, incluyendo la máscara, en notación CIDR será:

2134:A0B1:C2D3:E4F5:A00:27FF:FE75:CB0/64

Una vez obtenida la dirección IPv6 para la interfaz de red, asignamos la misma mediante la siguiente orden:

```
root@base:~# ifconfig enp0s3 inet6 add 2134:A0B1:C2D3:E4F5:A00:27FF:FE75:CB0/64 up
root@base:~#
```

Verificamos que la dirección IPv6 se haya asignado correctamente:

```
root@base:~# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.1.24 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::a0b1:c2d3%enp0s3 prefixlen 64 scopeid 0x20<link>
        inet6 2134:a0b1:c2d3:e4f5:a00:27ff%enp0s3 prefixlen 64 scopeid 0x0<global>
          ether 08:00:27:75:0c:b0 txqueuelen 1000 (Ethernet)
            RX packets 911 bytes 60054 (58.6 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 48 bytes 4466 (4.3 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Podemos ver que, efectivamente, se asignó la dirección:

2134:A0B1:C2D3:E4F5:A00:27FF:FE75:CB0 con máscara /64

VERIFICACIÓN DE CONECTIVIDAD

A continuación, verificaremos que la configuración funcione. Para este fin, utilizaremos el equipo configurado previamente:

2134:A0B1:C2D3:E4F5:A00:27FF:FE0A:2CC5/64



UNIVERSIDAD ANDINA DEL CUSCO

ADMINISTRACIÓN DE SERVICIOS DE REDES

GUÍA DE LABORATORIO

ECP 12 de 15

```
demo@antix1:~$ ifconfig
eth0: flags=28605<UP,BROADCAST,RUNNING,MULTICAST,DYNAMIC> mtu 1500
        inet 192.168.1.14 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 2134:a0b1:c2d3:e4f5:a00:27ff:fe0a:2cc5 prefixlen 64 scopeid 0x0<global>
        inet6 fe80::a00:27ff:fe0a:2cc5 prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:0a:2c:c5 txqueuelen 1000 (Ethernet)
        RX packets 667 bytes 50186 (49.0 KiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 416 bytes 37928 (37.0 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Probamos la conectividad mediante el comando ping6 o simplemente ping (el soporte de la variante del comando ping puede variar de una distribución de Linux a otra. En el caso de la distribución Debian utilizada, ambas opciones son válidas).

```
root@base:~# ping 2134:A0B1:C2D3:E4F5:0A00:27FF:FE0A:2CC5 -c3
PING 2134:A0B1:C2D3:E4F5:0A00:27FF:FE0A:2CC5(2134:a0b1:c2d3:e4f5:a00:27ff:fe0a:2cc5) 56 data bytes
64 bytes from 2134:a0b1:c2d3:e4f5:a00:27ff:fe0a:2cc5: icmp_seq=1 ttl=64 time=0.291 ms
64 bytes from 2134:a0b1:c2d3:e4f5:a00:27ff:fe0a:2cc5: icmp_seq=2 ttl=64 time=0.854 ms
64 bytes from 2134:a0b1:c2d3:e4f5:a00:27ff:fe0a:2cc5: icmp_seq=3 ttl=64 time=0.879 ms

--- 2134:A0B1:C2D3:E4F5:0A00:27FF:FE0A:2CC5 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 4ms
rtt min/avg/max/mdev = 0.291/0.674/0.879/0.273 ms
```

Podemos ver que, efectivamente, los equipos pueden comunicarse entre sí.



I. TRABAJOS DE LABORATORIO.

1. Elabore cartillas de documentación para los comandos mostrados a continuación: arp, bmon, dig, dog, drill, ethtools, fping, getent, host, hostname, ifplugstatus, iftop, ip, iperf, mtr, nc, netstat, networkctl, nmap, nmcli, nslookup, oping, ping, route, speedtest-cli, ss, tcpdump, tracepath, traceroute, vnstat, whois.

Cada cartilla debe mostrar los modificadores y un ejemplo de uso del comando.

Las cartillas se entregarán a través de la plataforma virtual.

2. En un computador con sistema operativo Linux Debian, comparta una carpeta con derechos de lectura y escritura utilizando SMB y configure otro computador con sistema operativo Linux Debian desde el cual se pueda acceder a la carpeta compartida en el primer computador. Ambos equipos deben utilizar direcciones IPv6 del bloque 2030:40:DF/48. Para realizar las pruebas de conectividad, desactive las direcciones IPv4.
3. En un computador con sistema operativo Linux Debian, comparta una carpeta con derechos de lectura y escritura utilizando NFS y configure otro computador con sistema operativo Linux Debian desde el cual se pueda acceder a la carpeta compartida en el primer computador. Ambos equipos deben utilizar direcciones IPv6 del bloque 2022:E9:A/48. Para realizar las pruebas de conectividad, desactive las direcciones IPv4.



**UNIVERSIDAD ANDINA DEL CUSCO
ADMINISTRACIÓN DE SERVICIOS DE REDES
GUÍA DE LABORATORIO**

ECP 14 de 15

II. EVALUACION

La evaluación de las actividades realizadas en la presente guía de práctica se hará en función de la siguiente tabla:

ACTIVIDAD	Procedimental		
	SESION 01	SESION 02	SESION 03
Réplica de los ejercicios de ejemplo.	05	--	--
Resolución del ejercicio propuesto 01	--	05	--
Resolución del ejercicio propuesto 02	--	--	05
Resolución del ejercicio propuesto 03	--	--	05
TOTAL	--	--	20



III. REFERENCIAS

1. Carrasco, E. “*Pre-P-20211-Cusco-Ingenieria de Sistemas-Redes y Comunicacion de Datos II-9A*”. <https://campus.uandina.edu.pe/course/view.php?id=9767>
2. Comer, D. “*Internetworking With TCP/IP Volumen 1*”. Ed. Prentice Hall 4ed.
3. Henry-Stocker S. “*More Ways to Examine Network Connections on Linux*” <https://www.networkworld.com/article/3233306/linux/more-ways-to-examine-network-connections-on-linux.html>
4. Henry-Stocker S. “*Using Ss To Examine Network And Socket Connections*”. <https://www.networkworld.com/article/3327557/linux/using-ss-to-examine-network-connections.html>
5. ITzGeek “*Change Default Network Name (Ens33) to Old “Eth0” on Debian 9*” <https://www.itzgeek.com/how-tos/linux/debian/change-default-network-name-ens33-to-old-eth0-on-debian-9.html>
6. Kili A. “*A Linux Sysadmin’s Guide to Network Management, Troubleshooting and Debugging*”. <https://www.tecmint.com/linux-networking-commands/> (21/07/2018)
7. Schoder, C. “*Redes En Linux. Guía de Referencia*”. Ed. Anaya Multimedia 2008.
8. Schoder C. “*Calculating IPv6 Subnets in Linux*” <https://www.linux.com/learn/intro-to-linux/2017/10/calculating-ipv6-subnets-linux>
9. Shah, S., Soyinka, W. “*Manual de Administración de Linux*”. Ed. McGraw Hill. 4º edición. 2005.