

Administración de redes Linux

Iria Estévez Ayres & Andrés Marín López

Abril 2025

Resumen

Vamos a trabajar con las interfaces de red, a manipular sus nombres y ver las tablas de enrutamiento.

1. Interfaces de red

1.1. net-tools

Comencemos instalando el paquete net-tools para utilizar varios comandos como netstat ifconfig, arp, route, iptunnel, etc. Ya instalado, como casi todos los comandos están en /sbin y /usr/sbin los añadimos al PATH (si no estuviesen):

```
~$ sudo apt install net-tools
...
~$ export PATH=/sbin:/usr/sbin:$PATH
~$ cat >> .bashrc
PATH=/sbin:/usr/sbin:$PATH
CTRL-D
```

Si consultamos las interfaces levantadas, veremos algo similar a esto:

```
~$ sudo ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe37:25b3  prefixlen 64  scopeid 0x20<link>
    ether 08:00:27:37:25:b3  txqueuelen 1000  (Ethernet)
    RX packets 56024  bytes 80950748 (77.2 MiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 5680  bytes 383342 (374.3 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 28  bytes 2796 (2.7 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 28  bytes 2796 (2.7 KiB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

La primera interfaz es la interfaz ethernet y su nombre sigue la convención de comenzar con el prefijo **enp** (ethernet network peripheral). El número a continuación es el número de periférico, y después viene **s** seguido del número de serie de la interfaz.

Esta es la forma estándar que tiene el kernel de nombrar a los interfaces cuando arranca.

1.2. Cambiando el nombre de las interfaces

Si queremos cambiar al nombrado tradicional (lo vamos a hacer como ejercicio y para aprender los comandos de cómo se haría, NO debéis hacerlo por defecto en vuestras máquinas), basta con seguir los siguientes pasos:

1. Informa al kernel de que cuando arranque no utilice este sistema de nombrado.

Edita `/etc/default/grub` y cambia:

```
GRUB_CMDLINE_LINUX="net.ifnames=0 biosdevname=0"
```

2. Actualizamos el menú de arranque (grub):

```
sudo update-grub
```

3. Configura la interfaz `eth0`.

Edita `/etc/network/interfaces` y añade la interfaz `eth0`, configurándolo para que utilice `inet` y `dhcp`.

```
auto eth0
iface eth0 inet dhcp
```

4. Rearranca la máquina.

```
sudo shutdown -r now
```

Ahora, cuando ejecutamos `ifconfig` vemos la interfaz ethernet como `eth0`.

```
~$ sudo ifconfig
[sudo] contraseña para astt:
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe37:25b3  prefixlen 64  scopeid 0x20<link>
    ether 08:00:27:37:25:b3  txqueuelen 1000  (Ethernet)
    RX packets 83  bytes 34304 (33.5 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 137  bytes 15862 (15.4 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 25  bytes 2606 (2.5 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 25  bytes 2606 (2.5 KiB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

1.3. Viendo las rutas y demás

El comando `route -n` nos permite ver las rutas configuradas. El flag `-n` es opcional, sirve para mostrar las direcciones en formato numérico y evita resolver el nombre de las direcciones.

```
$ sudo route -n
Tabla de rutas IP del núcleo

```

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	10.0.2.2	0.0.0.0	UG	0	0	0	eth0
10.0.2.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
169.254.0.0	0.0.0.0	255.255.0.0	U	1000	0	0	eth0

Donde:

- **Destination:** es el destino de red de la ruta.
- **Gateway** (pasarela): es el siguiente salto IP
- **Genmask:** la máscara de red (255.255.255.255 para un destino que sea un host y 0.0.0.0 para la ruta por defecto).
- Existen diferentes **flags**. **G** que se use la pasarela, **U**, la ruta es *up*.
- **Metric:** *coste* de la ruta (entre 0 y 9999). No lo usa el kernel Linux.
- **Ref:** número de referencias a esa ruta. No lo usa el kernel Linux.
- **Use:** número de lookups para la ruta. Puede ser el número de fracasos o de éxitos.
- **Iface:** es la interfaz por la que se van a enrutar los paquetes de esa ruta.

El comando `netstat` permite ver todas las conexiones del sistema, tanto sockets tcp/ip como sockets de unix, y nos permite también permite evitar resolver los nombres de las direcciones y hasta mostrar los procesos responsables del punto de la conexión.

Por ejemplo, podemos chequear los puertos abiertos en un servidor de esta manera:

```
$ sudo netstat -tulp
Active Internet connections (only servers)

```

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State	PID/Program name
tcp	0	0	0.0.0.0:ssh	0.0.0.0:*	LISTEN	830/sshd
tcp	0	0	localhost:smtp	0.0.0.0:*	LISTEN	838/exim4
tcp6	0	0	:::http	:::*	LISTEN	840/apache2
tcp6	0	0	:::ssh	:::*	LISTEN	830/sshd
tcp6	0	0	ip6-localhost:smtp	:::*	LISTEN	838/exim4
tcp6	0	0	:::https	:::*	LISTEN	840/apache2