

20. NETSTAT

netstat

- netstat es una herramienta básica en cualquier SSOO (Unix, Linux, macOS, Windows...)
- Muestra información sobre conexiones de red, tablas de encaminamiento, estadísticas de los interfaces, NAT y multicast

Nos permitirá comprobar qué demonios están funcionando en nuestra máquina

- Es una herramienta básica para depurar cualquier servicio
- Un principio básico de seguridad en cualquier sistema es tener activos solo los servicios necesarios, cualquier nuevo servicio siempre implica un cierto riesgo

Información sobre conexiones de red, Linux

- -tu
Muestra información sobre TCP y UDP (y no sobre los *Unix domain sockets*)
- -p
Indica el programa a quien pertenece el socket
Si lo ejecuta un usuario ordinario, solo muestra algunos nombres
Si lo ejecuta root, muestra todos los nombres
- -a
Muestra todos los sockets, no solamente las conexiones establecidas sino también los sockets que están *escuchando*
- -n
Muestra direcciones IP y no nombres de máquina.
Muestra números de puerto, no nombre de servicio asociado (en los *well know ports*). No intenta resolver nombres de máquina.

```
koji@afrodita:~$ sudo netstat -tupan
```

Conexiones activas de Internet (servidores y establecidos)

Prot	Recv-Q	Send-Q	Dirección_Local	Dirección_Externa	Estado	PID/ Program name
tcp	0	0	0.0.0.0:22	0.0.0.0:*	ESCUCHAR	27488/sshd
tcp	0	0	127.0.0.1:631	0.0.0.0:*	ESCUCHAR	1320/cupsd
tcp	0	0	193.147.71.120:22	193.147.71.62:56881	ESTABLECIDO	26653/sshd: koji
tcp6	0	0	:::79	:::*	ESCUCHAR	19514/xinetd
tcp6	0	0	:::22	:::*	ESCUCHAR	27488/sshd
tcp6	0	0	:::1:631	:::*	ESCUCHAR	1320/cupsd
udp	0	0	0.0.0.0:49573	0.0.0.0:*		32398/avahi-daemon:
udp	0	0	0.0.0.0:5353	0.0.0.0:*		32398/avahi-daemon:

```
koji@afrodita:~$ sudo netstat -tupa
```

Conexiones activas de Internet (servidores y establecidos)

Prot	Recv-Q	Send-Q	Dirección_Local	Dirección_Externa	Estado	PID/ Program name
tcp	0	0	*:ssh	*:*	ESCUCHAR	27488/sshd
tcp	0	0	localhost:ipp	*:*	ESCUCHAR	1320/cupsd
tcp	0	0	afrodita:ssh	doublas:56881	ESTABLECIDO	26653/sshd: koji
tcp6	0	0	:::finger	:::*	ESCUCHAR	19514/xinetd
tcp6	0	0	:::ssh	:::*	ESCUCHAR	27488/sshd
tcp6	0	0	ip6-localhost:ipp	:::*	ESCUCHAR	1320/cupsd
udp	0	0	*:49573	*:*		32398/avahi-daemon:
udp	0	0	*:mdns	*:*		32398/avahi-daemon:

Problema:

Netstat no comprueba que el demonio que está atado a un puerto sea el demonio *habitual* en ese puerto.

Ejemplo: si atamos un servicio cualquiera al puerto 80, netstat lo tomará por un servidor web

- Un demonio puede escuchar en un puerto (p.e. el 22) de cualquier dirección de la máquina, por tanto, en cualquiera de los interfaces de la máquina

tcp	0	0	0.0.0.0:22	0.0.0.0:*	ESCUCHAR	27488/sshd
-----	---	---	------------	-----------	----------	------------

- O bien puede escuchar en un puerto (p.e. el 631), pero no de todas las direcciones/todos los interfaces, sino de una en concreto

tcp	0	0	127.0.0.1:631	0.0.0.0:*	ESCUCHAR	1320/cupsd
-----	---	---	---------------	-----------	----------	------------

Este demonio está en la máquina afrodita, pero no atiende peticiones hechas a la dirección pública de afrodita, solamente a *localhost* /127.0.0.1

- Podemos usar grep para filtrar la salida de netstat

```
koji@afrodita:~$ netstat -tupan |grep 22
```

tcp	0	0 0.0.0.0:22	0.0.0.0:*	ESCUCHAR	-
tcp	0	0 193.147.71.120:22	193.147.71.62:34285	ESTABLECIDO	-
tcp6	0	0 :::22	:::*	ESCUCHAR	-

Netsat en macOS

Uso típico:

```
netstat -f inet -an
```

- Para ver solo las conexiones tcp y udp, la opción no es -tu sino -f inet
- -a y -n se comportan como en Linux
- Para saber qué proceso escucha en cada puerto, no hay un equivalente a -p
Pero podemos usar
`sudo lsof -iTCP:12345 -sTCP:LISTEN`
(Donde 12345 sería el puerto a consultar)
o bien
`sudo lsof -i -n -P |grep UDP`