-

ceedcv
CENTRE ESPECÍFIC
D´EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

UD 10. LINUX: REDES

Sistemas informáticos
CFGS DAW

Borja Salom
b.salomsantamaria@edu.gva.es
2022/2023

Versión:230210.0943

Licencia

Nomenclatura

A lo largo de esta unidad se utilizarán diferentes símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

Importante

Atención

Interesante

ÍNDICE DE CONTENIDO

UD10. LINUX: REDES

1. LINUX NETWORKING COMMAN DS

Los comandos de red de Linux se utilizan ampliamente para inspeccionar, analizar, mantener y solucionar problemas de la red o redes conectadas al sistema.

Conozcamos primero la lista de los comandos básicos de red utilizados en Linux seguidos de una explicación detallada de cada uno de ellos.

1.1 Comando Ifconfig

Linux ifconfig significa configurador de interfaz. Es uno de los comandos más básicos utilizados en la inspección de redes.

ifconfig se utiliza para inicializar una interfaz, configurarla con una dirección IP y habilitarla o deshabilitarla. También se utiliza para mostrar la ruta y la interfaz de red.

Utilizando `ifconfig`, puede obtener detalles de una interfaz específica. Esto se muestra a continuación.

```
ifconfig
```

Con `ifconfig <interfaz>`, puede obtener los datos de una interfaz específica.

```
ifconfig eth0
ifconfig lo
ifconfig wlan0
```

Utilice `ifconfig <interfaz> <dirección> máscara de red <dirección>` para asignar una dirección IP y una máscara de red a una interfaz. Sin embargo, estos detalles se restablecerán después de reiniciar el sistema.

```
ifconfig eth0 172.29.79,70
ifconfig eth0 172.29.79.70 máscara de red
255.255.0.0 ifconfig eth0 172.29.79.70/16
```

Utilizar `ifconfig <interfaz> <abajo | arriba>` para activar o desactivar una interfaz.

```
ifconfig eth0 up
ifconfig eth0 down
```

Utilice `ifconfig <interfaz> mtu <número>` para establecer el tamaño de MTU (unidad de transmisión máxima). Por defecto, MTU tiene un tamaño de 1500.

```
ifconfig eth0 mtu 1600
```

1.2 Comando ip

El comando ip se utiliza para asignar una dirección a una interfaz de red y/o configurar los parámetros de la interfaz de red en sistemas operativos Linux. Este comando reemplaza al viejo y ahora obsoleto comando ifconfig en las distribuciones modernas de Linux.

Usando `ip a` o `ip addr` list y mostrar todas las direcciones ip asociadas en todas las interfaces de red

```
ip a
dirección ip
```

Puede seleccionar entre IPv4 e IPv6 utilizando la siguiente sintaxis:

```
ip -4 a
ip -6 a
```

También es posible especificar y listar detalles TCP/IP de interfaces particulares:

```
ip a show eth0
ip a list eth0
ip a show dev eth0
```

Para asignar la dirección IP a la interfaz, podemos utilizar el comando

`ip a add <dirección/máscara> dev <interfaz>`

```
ip a add 192.168.1.200/255.255.255.0 dev eth0
ip a add 192.168.1.200/24 dev eth0
```

Para eliminar o borrar la dirección IP de la interfaz podemos utilizar el comando: `ip a del <dirección> dev <interfaz>.`

```
ip a del 192.168.1.200/24 dev eth0
```

Para cambiar el estado del dispositivo a ARRIBA o ABAJO, podemos utilizar este comando:

`ip link set dev <interfaz> <arriba |abajo>`

```
ip link set dev eth0 up
ip link set dev eth0 down
```

Para cambiar la MTU del dispositivo, podemos utilizar este comando:

`ip link set mtu <número> dev <interfaz>`

```
ip link set mtu 9000 dev eth0
```

El Protocolo de Resolución de Direcciones (ARP) es un procedimiento para asignar una dirección IP dinámica a una dirección física permanente de máquina en una red de área local. Para ver la caché de vecinos/arp:

```
ip n show
ip neigh show
```

Para añadir una nueva entrada ARP podemos utilizar:

`ip neigh add <IP> lladdr <MAC> dev <interface> nud <state>`

```
ip neigh add 192.168.1.5 lladdr 00:1a:30:38:a8:00 dev eth0 nud perm
```

neighbour state (nud)	meaning
`permanent`	The neighbour entry is valid forever and can be only be removed administratively
`noarp`	The neighbour entry is valid. No attempts to validate this entry will be made but it can be removed when its lifetime expires.
`stale`	The neighbour entry is valid but suspicious. This option to ip neigh does not change the neighbour state if it was valid and the address is not changed by this command.
`reachable`	The neighbour entry is valid until the reachability timeout expires.

Para invalidar o borrar un ARP:

`ip neigh del <IP> dev <interface> (relincho ip del <IP> dev <interfaz>)`

```
ip neigh del 192.168.1.5 dev eth1
```

O para cambiar de estado

```
ip neigh chg 192.168.1.100 dev eth1 nud reachable
```

Para visualizar el conteto de las tablas de enrutamiento:

```
ip r
lista ip r
lista de rutas ip
ip r list [options] ip route

ip r list 192.168.1.0
```

Para añadir una nueva ruta;

`ip route add <ip/mask> via <gatewayIP>` or

`ip route add <ip/máscara> dev <interfaz>`

```
ip route add 192.168.1.0/24 via 192.168.1.254
ip route ad 192.168.1.0/24 dev eth0
```

Para eliminar la puerta de enlace predeterminada

```
ip route del default
```

para borrar una ruta:

```
ip route del 192.168.1.0/24 dev eth0
```

1.3 Comando ping

Linux ping es uno de los comandos de solución de problemas de red más utilizados. Básicamente comprueba la conectividad de red entre dos nodos.

1. ping son las siglas de Packet INternet Groper.
2. El comando ping envía la petición de eco ICMP para comprobar la conectividad de la red.
3. Sigue ejecutándose hasta que se

interrumpe. Utilice la tecla Ctrl+C para

interrumpir la ejecución. La sintaxis es `ping <destinarion>.`

```
ping google.com
ping 216.239.38.120
```

Puedes limitar el número de paquetes incluyendo "-c" en el comando ping.

`ping -c <número> <destino>`

El comando se utiliza para medir la respuesta media. Si no hay respuesta para el comando ping, puede asumir uno de los siguientes problemas con la red:

- Hay un problema físico que causa la pérdida de red.
- La dirección de destino puede ser disfuncional o incorrecta.
- La solicitud de ping está bloqueada debido a un objetivo.
- Puede haber un problema con la tabla de enrutamiento.

1.4 Antiguos comandos frente a nuevos comandos

Old command (Deprecated)	New command
`ifconfig enp6s0 down`	`ip link set enp6s0 down`
`ifconfig enp6s0 up`	`ip link set enp6s0 up`
`ifconfig enp6s0 192.168.2.24`	`ip addr add 192.168.2.24/24 dev enp6s0`
`ifconfig enp6s0 netmask 255.255.255.0`	`ip addr add 192.168.1.1/24 dev enp6s0`
`ifconfig enp6s0 mtu 9000`	`ip link set enp6s0 mtu 9000`
`ifconfig enp6s0:0 192.168.2.25`	`ip addr add 192.168.2.25/24 dev enp6s0`
`netstat`	`ss`
`netstat -tulpn`	`ss -tulpn`
`netstat -neopa`	`ss -neopa`
`netstat -g`	`ip maddr`
`route`	`ip r`
`route add -net 192.168.2.0 netmask 255.255.255.0 dev enp6s0`	`ip route add 192.168.2.0/24 dev enp6s0`
`route add default gw 192.168.2.254`	`ip route add default via 192.168.2.254`
`arp -a`	`ip neigh`
`arp -v`	`ip -s neigh`
`arp -s 192.168.2.33 1:2:3:4:5:6`	`ip neigh add 192.168.3.33 lladdr 1:2:3:4:5:6 dev enp6s0`
`arp -i enp6s0 -d 192.168.2.254`	`ip neigh del 192.168.2.254 dev wlp7s0`

2. S SH

SSH, Secure Shell, es un protocolo de administración remota a través del cual los usuarios pueden tanto modificar como controlar sus servidores remotos en Internet. Se creó para sustituir a Telnet, un protocolo no cifrado y que, por tanto, no ofrecía ningún tipo de seguridad a los usuarios.

A cambio, SSH hace uso de las técnicas criptográficas más innovadoras con el claro objetivo de que todas las comunicaciones entre los usuarios y los servidores remotos sean seguras. Dispone de una herramienta que permite autenticar al usuario remoto para, posteriormente, transferir las entradas del cliente al host y, finalmente, devolverlas a los usuarios.

Cabe destacar que los usuarios de los sistemas operativos Linux y MacOS pueden implementar el protocolo SSH en su servidor remoto muy fácilmente a través del terminal. Por supuesto, los usuarios de Windows también pueden hacerlo, aunque el procedimiento es diferente.

2.1 Intall Servidor SSH

Para instalar un servidor SSH en Ubuntu lo mejor es utilizar OpenSSH. Un punto a tener en cuenta es que en la gran mayoría de sistemas Linux de este servidor ya está disponible por defecto. Por lo tanto, para instalarlo simplemente tienes que dar la orden a tu gestor de paquetes.

```
sudo apt-get update
sudo apt-get install ssh
```

2.2 Puertos SSH

Actualmente, prácticamente el 100% de los servidores utilizan Linux como sistema operativo gracias al soporte y estabilidad que ofrece. Es por ello que los ciberataques contra estos servidores son cada vez más frecuentes. Así, es necesario reforzar la seguridad en ellos para evitar cualquier tipo de acceso no autorizado.

Una buena forma de mejorar la seguridad en los servidores Linux es cambiar el puerto SSH que el administrador utiliza para autenticarse mediante el protocolo SSH. La verdad es que cambiar el puerto SSH es un proceso muy sencillo. Te lo explicamos paso a paso.

1. Primero necesitas editar el ssh_config. Para ello debe utilizar el siguiente comando: nano /etc/ssh/sshd_config. Para lo que es necesario que tenga instalado el editor de texto de línea de comandos para Linux.
2. Una vez ejecutado el comando, busca la línea que dice "#Puerto 22". Así, lo que debes hacer es cambiar 22 por el número del puerto que quieres configurar. Además, tienes que quitar el #.
3. A continuación, guarda todos los cambios realizados. Para ello, pulsa las teclas Control y X al mismo tiempo.
4. El siguiente paso es reiniciar el servicio SSH con el siguiente comando: /etc/init.d/sshd restart.
5. A partir de ese momento realizarás todas las conexiones con el puerto que hayas elegido.

2.3 Comandos SSH

Este comando ofrece una comunicación muy segura ya que los datos viajan encriptados, a salvo de cualquier tipo de ciberataque. Cuando te conectas a otro ordenador utilizando SSH debes ejecutar el siguiente comando.

```
ssh <usuario>@<IP o dominio>
```

```
ssh user@192.160.1.1
```

Otro comando que vale la pena conocer, porque permite mover y copiar archivos y ficheros entre dos ordenadores. Es importante destacar que utiliza SSH para transmitir la información, de forma que viaja encriptada para ofrecer la máxima seguridad.

```
scp <ruta de fichero> <usuario>@<IP o Dominio>:<ruta de destino>
```

```
scp /tmp/archivo user@192.160.1.1:/tmp
```

si desea copiar un directorio completo debe utilizar el parámetro `-r`

```
scp -r /tmp/carpeta user@192.160.1.1:/tmp
```

3. S FTP

FTP, o "File Transfer Protocol", era un popular método no cifrado de transferencia de archivos entre dos sistemas remotos.

SFTP, siglas de SSH File Transfer Protocol o Secure File Transfer Protocol, es un protocolo independiente empaquetado con SSH que funciona de forma similar pero a través de una conexión segura. La ventaja es la capacidad de aprovechar una conexión segura para transferir archivos y recorrer el sistema de archivos tanto en sistemas locales como remotos.

En casi todos los casos, es preferible utilizar SFTP, en lugar de FTP, debido a sus características de seguridad subyacentes y su capacidad para aprovechar una conexión SSH. FTP es un protocolo no seguro que sólo debería utilizarse en casos limitados o en redes de confianza.

Por defecto, SFTP utiliza el protocolo SSH para autenticarse y establecer una conexión segura. Por lo tanto, están disponibles los mismos métodos de autenticación que en SSH.

Si puedes conectarte al ordenador utilizando SSH, habrás completado todos los requisitos necesarios para utilizar SFTP para gestionar archivos.

3.1 Conéctese a SFTP

Podemos establecer una sesión SFTP ejecutando el siguiente comando:

```
sftp user@192.160.0.1
```

El comando más útil para conocer primero es el comando help. Este comando te da acceso a un resumen de la ayuda sobre SFTP. Puedes invocarlo escribiendo cualquiera de estos en la sentencia:

```
ayuda
?
```

Podemos navegar por la jerarquía de archivos del sistema remoto utilizando varios comandos que funcionan de forma similar a sus homólogos del shell.

3.2 Transfiera archivos con SFTP

Si queremos descargar archivos de nuestro host remoto, podemos hacerlo ejecutando el siguiente comando:

```
obtener remoteFile
```

Como puede ver, por defecto, el comando get descarga un archivo remoto a un archivo con el mismo nombre en el sistema de archivos local.

Podemos copiar el archivo remoto a un nombre diferente especificando el nombre después:

```
get remoteArchivo localArchivo
```

El comando get también acepta algunas opciones. Por ejemplo, podemos copiar un directorio y todo su contenido especificando la opción recursiva:

```
get -r algunDirectorio
```

Podemos decirle a SFTP que mantenga los permisos y tiempos de acceso adecuados utilizando la bandera - P o -p:

```
get -Pr algúnDirectorio
```

Transferencia local de archivos al sistema remoto

Transferir archivos al sistema remoto es tan fácil como utilizar el comando "put":

```
put localArchivo
```

Las mismas opciones que funcionan con `get` se aplican a `put`. Así, para copiar un directorio local completo, puede ejecutar `put -r`:

```
put -r directorioLocal
```

Una herramienta familiar que resulta útil a la hora de descargar y subir archivos es el comando `df`, que funciona de forma similar a la versión de línea de comandos. Con él puedes comprobar que tienes espacio suficiente para completar las transferencias que te interesan:

```
df -h
```

Cualquier otro comando local funcionará como se espera. Para volver a su sesión SFTP, escriba:

```
salida
```

3.3 Manipulaciones sencillas de archivos con SFTP

SFTP te permite realizar algunos tipos de mantenimiento del sistema de archivos. Por ejemplo, puede cambiar el propietario de un archivo en el sistema remoto con:

```
chown usuarioID archivo
```

Fíjate en que, a diferencia del comando `chmod` del sistema, el comando SFTP no acepta nombres de usuario, sino que utiliza UIDs. Desafortunadamente, no existe una forma integrada de conocer el UID apropiado desde la interfaz SFTP.

Como solución alternativa, puede leer desde el archivo `/etc/passwd`, que asocia nombres de usuario con UIDs en la mayoría de los entornos Linux:

```
obtener /etc/passwd
Menos passwd
```

El comando `chmod` SFTP funciona normalmente en el sistema de archivos remoto:

```
chmod 777 publicArchivo
```