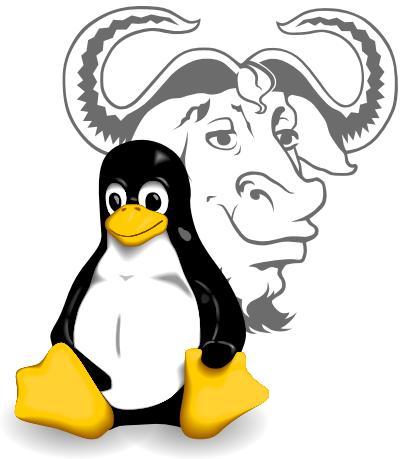
UD06.- Herramientas para la administración

Aprendemos herramientas para la gestión de redes y recursos compartidos.



Larry Ewing (Free Art License) <https://commons.wikimedia.org/wiki/File:Gnulinux.svg>

0.- Temas pendientes

Durante el transcurso del módulo no hemos tenido tiempo para explicar dos conceptos importantes y muy relacionados con los recursos compartidos. Por lo tanto vamos a parar en estos conceptos para verlos en profundidad antes de explicar la gestión de redes y servicios.

0.1.- Montaje de unidades

El montaje de unidades se refiere al proceso de incorporar un sistema de archivos almacenado en un dispositivo de almacenamiento (como un disco duro, una partición, o un dispositivo de almacenamiento extraíble) en el árbol de directorios del sistema.

En linux no se montan las unidades automáticamente al conectarlas al sistema. Para ello, se deben crear **puntos de montaje**, subdirectorios que representan cada una de las unidades. Estos subdirectorios se crean en el directorio /mnt/

Caso práctico

1. Antes de montar una unidad, hay que identificar el sistema de archivos que quieres montar. El comando lsblk nos muestra información sobre todos los dispositivos conectados a nuestro ordenador, disco duro, cd... Identificamos el disco que vamos a querer montar
2. Creamos una carpeta en /mnt/ donde montaremos nuestra unidad. mkdir /mnt/usb32
3. Utilizamos el comando mount. Este comando nos permite montar la unidad en la carpeta que le especifiquemos. Utilizaremos la información que hemos extraído de lsblk. mount /dev/sda1 /mnt/usb32

Ya tenemos accesible en nuestra carpeta /mnt/usb32 la unidad montada.

0.2.- Quotas

Las cuotas de disco en Linux son una funcionalidad crucial para gestionar y controlar el consumo de espacio de almacenamiento por parte de usuarios y grupos. Este mecanismo permite establecer límites predefinidos en la cantidad de espacio que cada usuario o grupo puede utilizar en el sistema de archivos.

Establecer quotas a nivel de usuario y grupo

Establecemos quotas.

Para establecer quotas lo primero hay que editar el fichero /etc/fstab

En la línea del sistema de archivos sobre el que queramos establecer la quota, especificamos el tipo de quota que queramos establecer en el sistema de archivos específico, si de grupo o de usuario: grpquota, usrquota.

Instalamos el paquete quota.

Ahora creamos los archivos aquota.user y aquota.group ejecutando:

```
quotacheck -cug /home
```

c indica que se deben crear nuevos archivos de cuota.
u indica que se cree un archivo para la cuota de usuario.
g indica que se cree un archivo para la cuota de grupos.

Necesitamos generar la tabla de uso del disco actual:

```
quotacheck -avug
```

Y ya podemos asignar cuotas de uso por usuario:

```
edquota username
```

Y por grupo:

```
edquota -g groupname
```

1.- Redes en las VM

Vamos a ver las distintas configuraciones del adaptador de red de los dos hipervisores que utilizamos en clase.

1.1.- VirtualBox

Modos de configuración del adaptador de red:

- NAT (Network Address Translation):
  - En este modo, la VM comparte la dirección IP del host y se conecta a la red a través de él. La VM puede acceder a Internet y a recursos externos, pero no es directamente visible desde la red externa. Es útil para configuraciones donde solo se necesita conectividad saliente.
- Adaptador puente (Bridge):
  - En modo puente, la VM se conecta directamente a la red física como si fuera un dispositivo independiente. Esto permite que la VM sea visible y accesible desde otras máquinas en la red externa. Es útil cuando se requiere una integración completa en la red física.
- Adaptador sólo-anfitrión (Host-Only):
  - Este modo crea una red aislada entre la VM y el host. Las VMs en modo host-only pueden comunicarse entre sí y con el host, pero no tienen acceso a la red externa. Es útil para configuraciones de desarrollo y pruebas.
- Red Interna (Internal Network):
  - Similar al modo host-only, pero permite que varias VMs se comuniquen entre sí **sin acceso externo**. Es útil para entornos de laboratorio o configuraciones donde solo se requiere comunicación interna.
- Adaptador de Red Genérico (Generic Driver):
  - Este modo permite instalar un controlador de red específico en la VM y es útil cuando se requiere un controlador de red personalizado. Es más avanzado y generalmente se utiliza en situaciones específicas.
- Red Nat:
  - Permite que varias máquinas se conecten entre sí y tengan conectividad externa a la vez. Requiere crear una red desde el administrador de redes de VirtualBox para utilizarlo.
- Desconectado (Not Attached):
  - Este modo desactiva completamente el adaptador de red en la VM. La VM no tiene conectividad de red.

1.2.- VMWare

Modos de configuración del adaptador de red:

- Bridged:
  - El adaptador de red de la máquina virtual se conecta directamente a la red física, permitiendo que la VM sea visible y accesible desde otras máquinas en la red externa.
- NAT:
  - La VM comparte la dirección IP del host y se conecta a la red a través de él. La VM tiene acceso a Internet, pero no es directamente accesible desde la red externa.
- Host-only:
  - Crea una red privada que se comparte entre la máquina y el host.
- Custom:
  - Permite configurar adaptadores de red personalizados, proporcionando opciones avanzadas de configuración según las necesidades específicas del usuario.
- LAN Segment:
  - Este modo crea una red privada entre máquinas virtuales, permitiendo la comunicación interna sin acceso a la red externa.

2.- Gestión de redes

Linux es conocido por su eficiente gestión de redes y ofrece un robusto conjunto de herramientas y características para administrar y configurar conexiones de red. Aprenderemos las más importantes.

2.1.- Tarjeta de red

Son dispositivos de hardware que permiten a un ordenador conectarse a una red y comunicarse con otros dispositivos en esa red.

Debes conocer

Las tarjetas de red en Linux suelen tener una nomenclatura específica que las identifica de manera única en el sistema. La nomenclatura puede variar según la versión del kernel y la distribución de Linux que estés utilizando. De manera general, es la siguiente:

Cable:

Tradicionalmente, las interfaces Ethernet por cable suelen seguir la nomenclatura ethX, donde X es un número que indica la secuencia de la interfaz. Por ejemplo, eth0, eth1, etc.

Inalámbrica (Wi-Fi):

Las interfaces inalámbricas pueden tener nombres como wlanX o wifiX, donde X es un número que indica la secuencia de la interfaz. Por ejemplo, wlan0, wlan1, etc.

2.2.- Comandos de gestión de redes

En este apartado, vamos a ver el funcionamiento de los comandos, que se explican brevemente de nuevo a continuación:

- El comando **`ip`** permite mostrar y manipular las interfaces de red.
- El comando **`ping`** permite verificar la conectividad entre varios dispositivos.
- El comando **`traceroute`** se utiliza para rastrear la ruta de los paquetes que se envían.
- El comando **`netstat`** muestra información detallada sobre las conexiones activas.
- Permiten extraer información sobre dominios: **`whois, dig, host`**
- Descarga directa de archivos: **`wget`**

Stevertigo (CC BY-SA) <https://commons.wikimedia.org/wiki/File:Wikipedesketch.png>

2.2.1.- Comando ip

El comando *ip* en Linux es una herramienta integral para la configuración y gestión de la red en el sistema operativo. A diferencia de *ifconfig* y *route*, que son herramientas más antiguas y están siendo gradualmente reemplazadas, *ip* proporciona una interfaz única para realizar una variedad de operaciones relacionadas con la red. Forma parte del paquete *iproute2* . Proporciona funcionalidades avanzadas para la configuración y gestión de la red, incluyendo la manipulación de interfaces de red, la configuración de direcciones IP, la gestión de enrutamiento...

Antiguamente se utilizaba el comando *ifconfig* para la configuración de interfaces de red y el comando *route* para la gestión de tablas de enrutamiento. Ambos han sido sustituidos por el comando ***ip***.

Opciones de IP

El comando ip admite una amplia variedad de subcomandos y opciones. Algunos de los subcomandos más comunes incluyen:

`ip link`: Configuración y visualización de interfaces de red.
`ip addr`: Configuración y visualización de direcciones IP y propiedades de las interfaces de red.
`ip route`: Configuración y visualización de la tabla de enrutamiento.
`ip neigh`: Visualización y manipulación de la tabla ARP (Resolución de Direcciones de Protocolo).

Uso de IP

Uso de ip para realizar tareas comunes de configuración de red:

1. Mostrar Información de Interfaces de Red:

```
ip link show
```

Este comando muestra información sobre todas las interfaces de red en tu sistema, incluyendo su nombre, estado y tipo.

2. Configurar una Dirección IP en una Interfaz:

```
sudo ip addr add 192.168.1.2/24 dev eth0
```

Este comando asigna la dirección IP 192.168.1.2 a la interfaz Ethernet eth0 con una máscara de red de 24 bits.

3. Desconectar una Interfaz de Red:

```
sudo ip link set eth0 down
```

Este comando desconecta la interfaz Ethernet eth0.

4. Conectar una Interfaz de Red:

```
sudo ip link set eth0 up
```

Este comando conecta la interfaz Ethernet eth0.

5. Cambiar la Dirección MAC de una Interfaz:

```
sudo ip link set dev eth0 address 00:11:22:33:44:55
```

Este comando cambia la dirección MAC de la interfaz Ethernet eth0.

6. Eliminar una Dirección IP de una Interfaz:

```
sudo ip addr del 192.168.1.2/24 dev eth0
```

Este comando elimina la dirección IP 192.168.1.2 de la interfaz Ethernet eth0.

2.2.2.- Comando ping

El comando ping en Linux y otros sistemas operativos se utiliza para verificar la conectividad de red entre tu máquina y otra máquina en la red. Ping envía paquetes de solicitud ICMP Echo a la dirección especificada y espera respuestas.

Sintaxis:
`ping [opciones] [dirección o nombre del host]`

1. Petición sencilla:
`ping iescomercio.com`

Este comando envía paquetes de solicitud ICMP Echo al servidor de iescomercio.com y muestra el tiempo de respuesta.

2. Petición sencilla a IP:

`ping 192.168.0.1`

Este comando envía paquetes de solicitud ICMP Echo a la IP 192.168.0.1 y muestra el tiempo de respuesta.

Opciones

-c: Especifica el número de paquetes a enviar antes de detenerse automáticamente.

-i: Establece el intervalo entre envíos de paquetes en segundos.

-t: Muestra la marca de tiempo en cada línea de salida.

-c: envía paquetes ilimitados hasta que se interrumpa de manera manual.

2.2.3.- Comando traceroute

El comando traceroute en Linux (y en otros sistemas operativos) se utiliza para rastrear la ruta que sigue un paquete desde tu máquina hasta una máquina de destino en una red. Este comando muestra la serie de saltos que realiza el paquete a través de los routers en la red, y también muestra el tiempo de ida y vuelta para cada salto.

Sintaxis:
`traceroute [opciones] [nombre del host o dirección IP]`

Uso común:
`traceroute www.iescomercio.com`

Este comando rastrea la ruta desde tu máquina hasta el servidor de iescomercio.com y muestra la lista de saltos.

Debes conocer

Traceroute también se puede usar con direcciones IP en lugar de nombres de host.

Otras opciones

Especificando Número de Saltos:
`traceroute -m 20 www.iescomercio.com`

Este comando limita el número de saltos a 20.

Usar ICMP en lugar de UDP:
`traceroute -I www.iescomercio.com`

Este comando utiliza paquetes ICMP en lugar de UDP para realizar la traza.

Mostrar Nombres de Host en lugar de Direcciones IP:
`traceroute -n www.iescomercio.com`

Este comando muestra las rutas con direcciones IP en lugar de resolver los nombres de host.

Mostrar tiempo de retardo para Cada Salto:

```
traceroute -q 3 www.iescomercio.com
```

Este comando envía 3 paquetes de traza para cada salto y muestra los tiempos de retardo.

2.2.4.- Comando netstat

El comando netstat en Linux proporciona información sobre diversas estadísticas de red, conexiones de red, tablas de enrutamiento y más. Aunque netstat está siendo reemplazado gradualmente por `ss` y por el comando `ip`, aún es una herramienta ampliamente utilizada.

Sintaxis

```
netstat [opciones]
```

Opciones comunes

Mostrar Todas las Conexiones:

```
netstat -a
```

Este comando muestra todas las conexiones y escuchas de red, tanto IPv4 como IPv6.

Mostrar Estadísticas por interfaz:

```
netstat -i
```

Este comando muestra estadísticas de las interfaces de red, incluyendo paquetes transmitidos y recibidos.

Mostrar Estadísticas por protocolo:

```
netstat -s
```

Este comando muestra estadísticas detalladas para diversos protocolos, como TCP, UDP, ICMP, etc.

Mostrar Procesos que utilizan Puertos:

```
netstat -tulpn
```

Este comando muestra los procesos que están utilizando puertos, junto con sus identificadores de proceso (PID) y nombres.

Mostrar Solo Conexiones IPv4:

```
netstat -4
```

Mostrar servicios escuchando la red:

```
netstat -ltp
```

Este comando muestra los programas que están escuchando en puertos, junto con sus

PID y nombres.

Mostrar Información de Estado de Conexiones:

```
netstat -an
```

Este comando muestra información del estado de las conexiones, como establecido, esperando, etc.

Debes conocer

La opción -n en netstat muestra direcciones y puertos en formato numérico en lugar de dominios.
Netstat también puede ser utilizado para mostrar información detallada sobre conexiones o interfaces específicas.

2.2.5.- Comando whois, dig, host

Estos tres comandos son esenciales para obtener la información sobre los dominios. Un dominio es una dirección única en Internet que se utiliza para identificar un sitio web. Está compuesto por un nombre de dominio y una extensión, como por ejemplo "iescomercio.com". Cuando un usuario escribe un nombre de dominio en su navegador de internet, se envía una solicitud al servidor de nombres de dominio para traducir el nombre de dominio que es fácil de recordar en una dirección IP. Esta dirección corresponde al servidor donde está alojado el sitio web. Luego, el navegador se conecta al servidor web utilizando esa dirección IP y carga el contenido del sitio web asociado al nombre de dominio.

El comando **whois** se utiliza para obtener información sobre el registro de un nombre de dominio. Proporciona detalles sobre quién lo tiene registrado, el propietario, los servidores de nombres y otra información relevante.

Sintaxis:

```
whois [nombre del dominio]
```

Ejemplo:

```
whois example.com
```

El comando **dig** permite obtener más información sobre el dominio. Nos puede dar información de cómo está configurado el dominio y cómo realiza la redirección al hosting.

Sintaxis:

```
dig [nombre del dominio] [tipo de registro]
```

Ejemplo:

```
dig example.com A
```

Obtiene información sobre el registro A del dominio.

El comando **host** se utiliza para realizar consultas DNS básicas y obtener información sobre la resolución de nombres. Muestra las direcciones IP asociadas con un nombre de dominio.

Sintaxis:

```
host [nombre del dominio]
```

Ejemplo:

```
host example.com
```

2.2.6.- Comando wget

El comando wget es una herramienta de la terminal que se utiliza para descargar archivos desde la web.

Sintaxis:
```
wget [opciones] [URL]
```

Ejemplo:
```
wget https://iescomercio.com/informatica/download/PD/FP/SMR/23-24_SMR1_SOM.pdf
```

Descargar en un Directorio Específico:
```
wget -P /ruta/del/directorio https://iescomercio.com/informatica/download/PD/FP/SMR/23-24_SMR1_SOM.pdf
```

Descargar con un Nombre de Archivo Específico:
```
wget -O nombre_personalizado.pdf https://iescomercio.com/informatica/download/PD/FP/SMR/23-24_SMR1_SOM.pdf
```

Limitar la Velocidad de Descarga:
```
wget --limit-rate=100k https://iescomercio.com/informatica/download/PD/FP/SMR/23-24_SMR1_SOM.pdf
```

Descargar Recursivamente:
```
wget -r https://iescomercio.com/
```

Ignorar Certificado SSL:
```
wget --no-check-certificate https://iescomercio.com/informatica/download/PD/FP/SMR/23-24_SMR1_SOM.pdf
```

Debes conocer

Wget es una herramienta muy versátil y puede ser utilizada para realizar descargas desde servidores HTTP, HTTPS y FTP.
Puedes usar wget para descargar archivos de manera interactiva y automática desde la línea de comandos.
La opción -r se utiliza para descargar recursivamente, lo que significa que descargará todo el contenido vinculado desde la URL proporcionada.

2.3.- Ficheros de configuración de redes (revisar)

/etc/resolv.conf

Es un archivo de configuración que se utiliza para especificar los servidores DNS que el sistema utilizará para resolver nombres de dominio.

El contenido del fichero puede parecerse a lo siguiente:

```
nameserver 8.8.8.8
```

Cada línea contendrá una sola dirección IP correspondiente a un servidor DNS. Si deseamos añadir más servidores DNS, podemos agregar hasta un máximo de 3 líneas. El orden es crucial, ya que las consultas se dirigirán al servidor de la primera línea; si este falla, se dirigirán al servidor de la segunda línea; y si este también falla, se dirigirán al servidor de la tercera línea.

Netplan

Netplan es una utilidad de configuración de red que permite los usuarios de sistemas configurar la red de manera más sencilla y coherente mediante archivos de configuración en formato YAML.

Estos ficheros de configuración se ubican en: `/etc/netplan`

Debería aparecer un fichero llamado `01-network-manager-all.yaml`

Un ejemplo de configuración del fichero puede ser el siguiente:

```
network:
  version: 2
  renderer: networkd
  ethernets:
    ens33:
      addresses: [192.168.1.100/24]
      gateway4: 192.168.1.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
```

Con los ficheros YAML hay que tener mucho cuidado, ya que necesitan una tabulación correcta para su interpretación.

Una vez realizados los cambios en el fichero se puede probar la configuración con `sudo netplan try`

Si la configuración es correcta, podremos aplicar los cambios con `sudo netplan apply`

3.- Servicios en red

Las características de Linux como sistema operativo lo hacen propicio para una configuración y comunicación eficiente en tareas de redes locales. Por ello se instalan servicios en red en las máquinas Linux.

Los servicios en red son procesos que se ejecutan y proporcionan funcionalidades específicas en una red de ordenadores. Estos servicios permiten la comunicación entre dispositivos en una red y pueden ofrecer una amplia gama de funcionalidades, desde compartir recursos hasta proporcionar acceso remoto a equipos.

Vamos a ver cómo gestionar los servicios y algunos de los más importantes.

3.1.- Gestión de servicios

Para gestionar los servicios, en Ubuntu se utiliza el comando systemctl.

Inicio y Parada de Servicios:

`systemctl start nombre_servicio`: Inicia un servicio.
`systemctl stop nombre_servicio`: Detiene un servicio.
`systemctl restart nombre_servicio`: Reinicia un servicio.
`systemctl reload nombre_servicio`: Recarga la configuración de un servicio sin reiniciarlo.

Activación y Desactivación de Servicios:

`systemctl enable nombre_servicio`: Habilita un servicio para que se inicie automáticamente en el arranque del sistema.
`systemctl disable nombre_servicio`: Deshabilita un servicio para que no se inicie automáticamente en el arranque del sistema.

Consultar el Estado de los Servicios:

`systemctl status nombre_servicio`: Muestra el estado actual de un servicio, incluyendo si está en ejecución o detenido.

Listar Todos los Servicios:

`systemctl list-units --type=service`: Lista todos los servicios disponibles en el sistema.

3.2.- SSH

El servicio SSH (Secure Shell) es comúnmente utilizado para acceder y administrar de forma segura sistemas remotos. Este servicio nos permite conectarnos de manera segura a cualquier equipo con linux y trabajar de manera remota.

En la mayoría de los sistemas basados en Unix, incluyendo Ubuntu, el servidor SSH es implementado mediante el paquete OpenSSH. Se instala con el siguiente comando: `sudo apt install openssh-server`

Una vez instalado deberemos comprobar el estado del servicio. Al estar activo podremos utilizar un cliente como Putty para conectarnos desde cualquier equipo Windows a nuestro equipo con Linux.

3.3.- FTP

FTP (File Transfer Protocol) es un protocolo de red estándar utilizado para transferir archivos entre un cliente y un servidor. Este protocolo nos permite subir, descargar, ver o manipular archivos en y del servidor.

Para ello necesitamos un servidor y un cliente.

De manera general en se suele utilizar el servidor `vsftpd` aunque existen otros.

Para instalarlo usaremos sudo `apt install vsftpd`

Este servidor se instala con una configuración genérica. El fichero `/etc/vsftpd.conf` contiene la configuración del servicio y ahí deberemos realizar todos los cambios de configuración. Para poder subir ficheros, deberemos descomentar estas tres líneas:

```
anonymous_enable=NO

write_enable=YES

local_enable=YES
```

Una vez hecho esto, reiniciamos el servicio y ya nos podremos conectar con cualquier usuario al servidor ftp. Para ello podemos utilizar un cliente ftp como Filezilla.

3.4.- Samba

Samba es un servicio en Linux que implementa el protocolo SMB y permite compartir archivos entre windows y linux de manera invisible para el usuario.

Para el funcionamiento correcto de Samba, las máquinas que quieres conectar deben estar conectadas en red.

Instalación y configuración de Samba

Instalación del paquete samba:

```
sudo apt install samba
```

Configuración del servicio:

Tenemos que decidir qué carpeta queremos compartir. Podemos crear una carpeta llamada samba en la raíz del sistema:

```
sudo mkdir /samba
```

Una vez creada la carpeta, tenemos que modificar el fichero de configuración de samba para indicarle la carpeta y otros datos:

```
sudo nano /etc/samba/smb.conf
```

Aquí hacemos los siguientes cambios:

```
[samba-share]
path = /samba #indicamos la ruta
read only = no #queremos poder realizar cambios desde todos los equipos
browsable = yes #podemos acceder a todas las carpetas comparidas
```

Creamos un nuevo usuario con adduser y establecemos su contraseña y lo añadimos al sistema samba con :

```
smbpasswd -a [nombre-usuario]
```

Este usuario lo usaremos para autenticarse al sistema de samba.

Ya solo nos queda preparar en la carpeta samba que hemos creado antes una carpeta public y establecer los siguientes permisos:

```
sudo chown -R nobody:nogroup /samba/public
sudo chmod -R 0777 /samba/public
sudo chgrp sambashare /samba/public
```

Reiniciamos el servicio y ya nos podemos conectar desde cualquier otro equipo de la red.
