

Linux: BBK - The Bridge

Comandos Linux

1. `ls` : Lista archivos y directorios en el directorio actual.
2. `cd` : Cambia de directorio.
3. `pwd` : Muestra el directorio de trabajo actual.
4. `mkdir` : Crea un nuevo directorio.
5. `touch` : Crea un nuevo archivo vacío o actualiza la fecha de modificación de uno existente.
6. `cp` : Copia archivos y directorios.
7. `mv` : Mueve o renombra archivos y directorios.
8. `rm` : Elimina archivos y directorios.
9. `rmdir` : Elimina directorios vacíos.
10. `find` : Busca archivos y directorios.
11. `grep` : Busca patrones dentro de archivos.
12. `cat` : Muestra el contenido de archivos.
13. `less` / `more` : Muestra contenido de archivos de forma paginada.
14. `head` : Muestra las primeras líneas de un archivo.
15. `tail` : Muestra las últimas líneas de un archivo.
16. `chmod` : Cambia los permisos de archivos y directorios.
17. `chown` : Cambia el propietario y grupo de archivos y directorios.
18. `ln` : Crea enlaces (enlaces simbólicos o duros) a archivos.

19. `du` : Muestra el uso del disco por archivo o directorio.
20. `df` : Muestra el espacio libre y utilizado en el sistema de archivos.

Instalación de paquetes:

1. `apt-get` : Herramienta de gestión de paquetes para sistemas basados en Debian (Ubuntu, por ejemplo).
2. `yum` : Herramienta de gestión de paquetes para sistemas basados en Red Hat (CentOS, Fedora, etc.).
3. `dnf` : Herramienta de gestión de paquetes utilizada en distribuciones Fedora a partir de Fedora 22.
4. `pacman` : Gestor de paquetes de Arch Linux y distribuciones relacionadas.
5. `zypper` : Herramienta de gestión de paquetes utilizada en distribuciones openSUSE.
6. `dpkg` : Herramienta de bajo nivel para gestionar paquetes en sistemas Debian.
7. `rpm` : Herramienta de bajo nivel para gestionar paquetes en sistemas basados en Red Hat.
8. `snap` : Sistema de gestión de paquetes universal para distribuciones Linux.
9. `flatpak` : Plataforma de distribución de software independiente del sistema operativo.
10. `appimage` : Formato de paquete de software que funciona en varias distribuciones Linux.

Otros comandos útiles:

1. `sudo` : Ejecuta comandos con privilegios de superusuario.
2. `su` : Cambia de usuario.
3. `exit` : Sale de la sesión actual.
4. `tar` : Crea, descomprime y manipula archivos tar.
5. `zip` / `unzip` : Comprime y descomprime archivos en formato ZIP.
6. `ssh` : Accede a una máquina remota de forma segura.

7. `scp` : Copia archivos de forma segura entre hosts a través de SSH.
8. `wget` : Descarga archivos desde la web.
9. `curl` : Herramienta para transferencia de datos con sintaxis de URL.
10. `history` : Muestra el historial de comandos ejecutados.

Creación y gestión de grupos en Linux

1. Crear grupo en Linux:

Para crear un grupo en Linux, puedes utilizar el comando `groupadd` . Abre una terminal y ejecuta el siguiente comando, reemplazando `<nombre_grupo>` con el nombre que desees para el grupo:

```
bashCopy code
sudo groupadd <nombre_grupo>
```

2. Crear usuario en Linux y asignarle contraseña:

Para crear un usuario en Linux y asignarle una contraseña, utiliza el comando `adduser` . Ejecuta el siguiente comando en la terminal, reemplazando `<nombre_usuario>` con el nombre que desees para el usuario:

```
bashCopy code
sudo adduser <nombre_usuario>
```

Luego, seguirás las instrucciones para establecer una contraseña para el usuario.

3. Asignar usuarios a un grupo:

Para asignar un usuario a un grupo existente, utiliza el comando `usermod` . Ejecuta el siguiente comando en la terminal, reemplazando `<nombre_usuario>` con el nombre del usuario y `<nombre_grupo>` con el nombre del grupo:

```
bashCopy code
sudo usermod -aG <nombre_grupo> <nombre_usuario>
```

4. Eliminar grupo:

Para eliminar un grupo en Linux, utiliza el comando `groupdel`. Ejecuta el siguiente comando en la terminal, reemplazando `<nombre_grupo>` con el nombre del grupo que deseas eliminar:

```
bashCopy code
sudo groupdel <nombre_grupo>
```

5. Eliminar usuario de un grupo:

Para eliminar un usuario de un grupo en Linux, utiliza el comando `gpasswd`. Ejecuta el siguiente comando en la terminal, reemplazando `<nombre_usuario>` con el nombre del usuario y `<nombre_grupo>` con el nombre del grupo:

```
bashCopy code
sudo gpasswd -d <nombre_usuario> <nombre_grupo>
```

6. Hacer que un directorio solo esté disponible para un grupo:

Para hacer que un directorio esté disponible solo para un grupo específico, puedes establecer los permisos adecuados en el directorio y luego asignar el grupo al directorio. Aquí hay una serie de comandos para hacerlo:

```
bashCopy code
# Crear el directorio sudo mkdir /ruta/al/directorio # Establ
ecer permisos para el grupo sudo chown :<nombre_grupo> /ruta/
al/directorio sudo chmod 770 /ruta/al/directorio
```

Esto establece el propietario del directorio al grupo especificado y establece los permisos para que el propietario y el grupo tengan acceso completo al directorio.

7. Cambiar de usuario:

Para cambiar de usuario en Linux, utiliza el comando `su`. Ejecuta el siguiente comando en la terminal y proporciona la contraseña del usuario al que deseas cambiar:

```
bashCopy code
su <nombre_usuario>
```

Si deseas cambiar al usuario root, simplemente ejecuta `su` sin ningún argumento:

```
bashCopy code
su
```

Para ver los grupos y usuarios en un sistema Linux, puedes utilizar una combinación de comandos en la terminal. Aquí te muestro cómo hacerlo:

Ver grupos:

Puedes utilizar el comando `cat` para ver el contenido del archivo `/etc/group`, que contiene la información sobre los grupos en el sistema. Abre una terminal y ejecuta el siguiente comando:

```
bashCopy code
cat /etc/group
```

Este comando mostrará una lista de todos los grupos del sistema, cada línea representa un grupo y su información asociada.

Ver usuarios:

Para ver la lista de usuarios en el sistema, puedes utilizar el comando `cat` con el archivo `/etc/passwd`. Ejecuta el siguiente comando en la terminal:

```
bashCopy code
cat /etc/passwd
```

Este comando mostrará una lista de todos los usuarios del sistema, cada línea representa un usuario y su información asociada.

Alternativamente, puedes usar el comando `getent passwd` para obtener una lista de usuarios del sistema:

```
bashCopy code
getent passwd
```

Este comando mostrará una lista similar a `cat /etc/passwd`, pero también incluirá información sobre usuarios que pueden no tener una entrada en el archivo `/etc/passwd` (por ejemplo, usuarios de sistemas de autenticación externos como LDAP).

Recuerda que algunos sistemas pueden tener un gran número de usuarios y grupos, por lo que la salida de estos comandos puede ser extensa. Puedes usar herramientas de filtrado como `grep` para buscar usuarios o grupos específicos si es necesario:

```
bashCopy code
cat /etc/passwd | grep <nombre_usuario>
```

o

```
bashCopy code
cat /etc/group | grep <nombre_grupo>
```

Agregar usuario al grupo "sudo" (Ubuntu y sistemas basados en Debian)

1. Abre una terminal.
2. Ejecuta el siguiente comando para agregar el usuario al grupo "sudo", reemplazando `<nombre_usuario>` con el nombre del usuario al que deseas otorgar acceso de root:

```
bashCopy code
```

```
sudo usermod -aG sudo <nombre_usuario>
```

Asignando permisos a un supuesto usuario llamado david

Para dar permisos de escritura al usuario `david` en el directorio `DavidFiles` y su archivo `hola_mundo.txt`, puedes utilizar los comandos `chown` y `chmod`.

Primero, cambia el propietario del directorio y del archivo para que sean `david`:

```
bash
```

```
sudo chown -R david:DavidFiles DavidFiles
```

El `-R` hace que el cambio de propietario sea recursivo, afectando también a los archivos y subdirectorios dentro de `DavidFiles`.

Luego, otorga permisos de escritura al usuario `david` en el directorio y el archivo:

```
bash
```

```
chmod +w DavidFiles hola_mundo.txt
```

Con esto, el usuario `david` debería tener ahora permisos de escritura en el directorio `DavidFiles` y en su archivo `hola_mundo.txt`. Se usa `chgrp` para cambiar el grupo