Contenido

[1. Comandos más comunes 2](#_Toc129510802)

[1.1 mkdir 2](#_Toc129510803)

[2. Usuarios, grupos 2](#_Toc129510804)

# Procedo de instalación en Windows.

Para instalar una imagen de Ubuntu 22 LTS en VirtualBox, sigue los siguientes pasos:

1. Descarga la imagen ISO de Ubuntu 22 LTS desde el sitio web oficial de Ubuntu: https://releases.ubuntu.com/22.04/
2. Abre VirtualBox y haz clic en el botón "Nuevo" para crear una nueva máquina virtual.
3. Ingresa un nombre para la máquina virtual y selecciona "Linux" como el tipo de sistema operativo y "Ubuntu (64-bit)" como la versión.
4. Asigna la cantidad de memoria RAM y de almacenamiento que deseas utilizar para la máquina virtual.
5. En la pantalla "Disco duro", selecciona "Usar un archivo de disco duro existente" y haz clic en el botón "Seleccionar archivo". Navega hasta la ubicación donde descargaste la imagen ISO de Ubuntu 22 LTS y selecciónala.
6. Haz clic en "Crear" para crear la máquina virtual.
7. Inicia la máquina virtual y selecciona la opción "Instalar Ubuntu" en el menú de inicio.
8. Sigue las instrucciones en pantalla para instalar Ubuntu en la máquina virtual. Cuando se te solicite, selecciona la opción de "Instalación normal" y sigue los pasos para configurar el idioma, la ubicación y el usuario.
9. Una vez que se complete la instalación, reinicia la máquina virtual y ya podrás usar Ubuntu 22 LTS en VirtualBox.

# Comandos más comunes

## mkdir

El comando mkdir permite crear un nuevo directorio o directorios, es decir, lo que en Windows o en general se entiende clásicamente como una carpeta o lugar donde se almacenan otros archivos o carpetas.

**mkdir [depart1] [depart2] … [depart\_n]**

Además, tenemos otras opciones que podemos añadir al comando mkdir:

|  |  |
| --- | --- |
| **Opción** | **Descripción** |
| -p, --parents | Crea directorios padres de forma recursiva. Por ejemplo, mkdir -p /ruta/a/b/c creará los directorios /ruta, /ruta/a, /ruta/a/b y /ruta/a/b/c si no existen. |
| -m, --mode=modo | Establece los permisos del nuevo directorio en modo octal. Por ejemplo, mkdir -m 755 /ruta/nuevo\_directorio establece los permisos del nuevo directorio en rwxr-xr-x. |
| -v, --verbose | Muestra un mensaje por cada directorio creado. Por ejemplo, mkdir -v /ruta/nuevo\_directorio mostrará el mensaje creando directorio '/ruta/nuevo\_directorio' en la terminal. |
| -h, --help | Muestra la ayuda y la documentación del comando mkdir. |
| -version, --version | Muestra la versión del comando mkdir. |

Nota: dependiendo de la distribución de Unix podremos tener más o menos opciones.

# Permisos, Usuarios, grupos

## Permisos

Muchas veces queremos asegurarnos de que el permiso que se le da a un usuario para leer, escribir o ejecutar elementos (archivos, directorios, programas, etc) de nuestro sistema operativo este limitado según el tipo de usuario. Por ejemplo, bajo el siguiente caso:

* Tenemos tres usuarios: Juan, María y Pablo.
* Tenemos un archivo que ha creado Juan denominado como “confidencial.txt”.
* Juan solo quiere que acceda a su archivo María, dado que tiene información sensible que no quiere que conozca Pablo.

¿Cómo podemos hacer o controlar los permisos? ¿Qué le dice al sistema operativo que si alguien ha iniciado sesión como

La respuesta a esto es la mascará de permisos, cada archivo, directorio, programa, etc de Linux tiene asociada una, tal y como se muestra en la siguiente imagen:

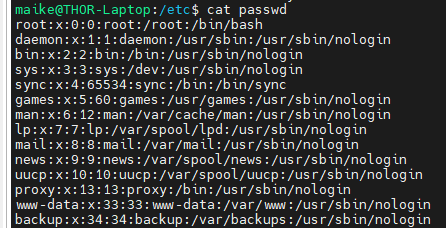
## Usuarios

### useradd

El comando para agregar un usuario sería

**useradd nombre\_usuario**

Una vez que hemos creado el usuario podemos comprobar su existencia visitando el archivo que contiene todos los usuarios del sistema operativo, esta archivo se denomina ‘passwd’ y se encuentra dentro del directorio raíz /etc. Al abrirlo veremos un archivo con múltiples líneas, donde cada una de ellas representa a un usuario del sistema operativo:



Si fragmentamos cada línea en diferentes componentes divididos por el carácter ‘:’ podremos apreciar, que de izquierda a derecha cada uno significa lo siguiente:

* **Nombre de usuario:** El primer campo es el nombre de usuario, que en este caso es root.
* **Contraseña:** El segundo campo es la contraseña del usuario. En su lugar, en distribuciones modernas de Linux y Unix, se utiliza una cadena de caracteres cifrada o un asterisco (\*) para indicar que la contraseña se almacena en otro archivo (como /etc/shadow) o que no se ha establecido una contraseña para el usuario.
* **UID:** El tercer campo es el ID de usuario (UID), que es un número único que identifica al usuario. En este caso, el UID es 0, que es el UID del usuario root y es el UID predeterminado para el usuario administrador en muchos sistemas.
* **GID:** El cuarto campo es el ID de grupo (GID) del usuario. En este caso, el GID es también 0, que es el GID del grupo root. Todos los usuarios con el mismo GID pertenecen al mismo grupo.
* **Información del usuario:** El quinto campo es la información del usuario, que generalmente incluye el nombre completo del usuario. En este caso, la información del usuario es simplemente root.
* **Carpeta de inicio:** El sexto campo es la carpeta de inicio del usuario, que es el directorio donde el usuario comienza cuando inicia sesión en el sistema. En este caso, la carpeta de inicio del usuario root es /root.
* **Shell predeterminado:** El séptimo campo es el shell predeterminado del usuario, que es el programa que se ejecuta cuando el usuario inicia sesión. En este caso, el shell predeterminado para el usuario root es /bin/bash.

Además, cuando el usuario inicia sesión en el sistema operativo, por defecto, si abre el terminal, se le iniciará un Shell determinado (una consola para ejecutar comandos), hay muchos tipos de consolas, cada una con sus ventajas y sus desventajas y con diversas sintaxis.

### Opciones de useradd.

#### -d (establecer directorio raíz)

La opción -d establece el directorio por defecto que se carga al crear un usuario

**sudo useradd -d /home/misusuarios/user\_name user\_name**

#### -s(establecer Shell por defecto)

La opción -s especifica el bash por defecto que se va a usar en el usuario

**sudo useradd -s /bin/bash user\_name**

### Opciones de usermod.

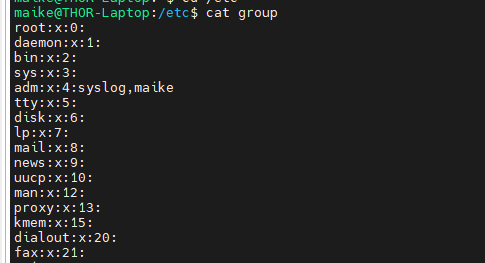
## Grupos

Los grupos existen ante la necesidad de que varios usuarios tengan los mismos permisos sobre un archivo, por ejemplo, queremos que Juan María tenga permisos de escritura en el archivo “confidencial.txt”, sin embargo, no queremos que Pablo tenga ningún permiso.

Para crear un grupo tenemos el comando

**groupadd [depart1]  
groupadd [depart2]**

Una vez hemos creado un grupo, normalmente, podemos visualizar el archivo /etc/group y ver un texto con el con el formato representado en la imagen de la derecha de abajo, cada línea representa a un grupo y cada grupo esta compuesta por varios elementos, separados por el carácter ‘:’, si lo estudiamos de izquierda a derecha tendremos que cada uno de estos elementos son:

* **Nombre del grupo**: es el nombre del grupo de usuarios en el sistema.
* **Contraseña del grupo**: es una contraseña cifrada para el grupo. Este campo se suele dejar en blanco en las distribuciones modernas de Linux, y se utiliza una x para indicar que la contraseña está cifrada y se almacena en otro archivo.
* **ID del grupo**: es el ID numérico del grupo en el sistema.
* **Miembros del grupo**: es una lista separada por comas de los nombres de usuario que pertenecen al grupo. Si no hay miembros en el grupo, este campo se deja en blanco.

# Archivos.

## Chown

El directorio chown puede interpretarse como “change owner” y nos permite cambiar la propiedad de un objeto del sistema (enlaces, archivos, ejecutables, directorios, etc).