

Sesión 1

¿Qué es Linux?

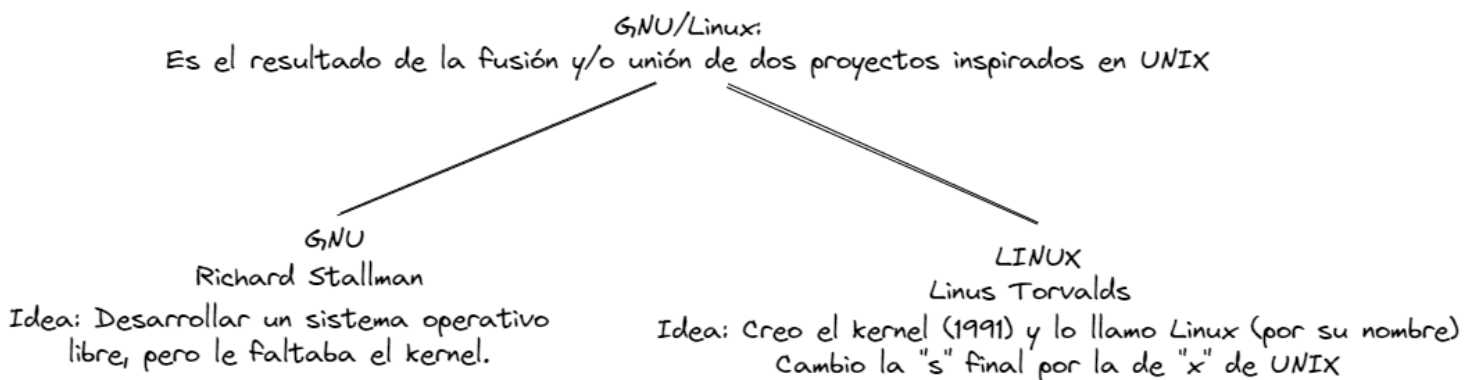
Linux es un **Kernel** libre y abierto, de manera que cualquier desarrollador puede “usarlo” para crear un sistema operativo. Linux nació en 1991, y le debe su nombre a su creador: Linus Torvalds. Es abierto y utiliza la Licencia Pública General de GNU (GNU General Public License). Esto quiere decir que **cualquiera puede utilizarlo** en sus propios sistemas operativos

¿Qué es un kernel?

El Kernel o Núcleo es una pieza fundamental en cualquier sistema operativo, ya sea que tengamos Windows, macOS o **GNU/Linux**, todos ellos tienen su propio núcleo que se encarga de que el software y el hardware de cualquier ordenador trabajen juntos. El Kernel administra la memoria que utilizan los procesos y las aplicaciones que se ejecutan en su ordenador. También se encarga de administrar y utilizar los drivers que tiene, de forma que las aplicaciones y programas puedan utilizar correctamente los componentes físicos del equipo como el procesador o la tarjeta gráfica. El Kernel de por sí no puede funcionar como sistema operativo si no se le añaden herramientas y aplicaciones.

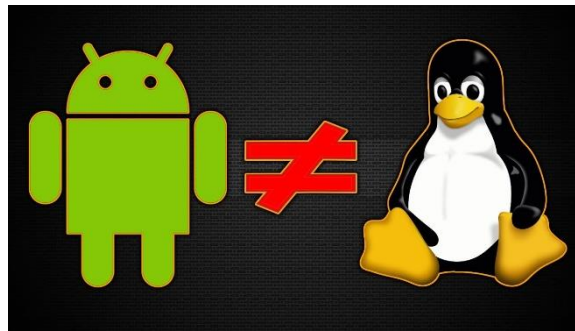
¿Qué es GNU/Linux?

GNU buscaba ser un sistema operativo libre completo. En este contexto Linux le vino como anillo al dedo, ya que al sistema operativo GNU le faltaba un Kernel. Como Linux es un kernel libre y abierto que utiliza la Licencia Pública General de GNU, esto significaba que podría utilizarse en el sistema operativo GNU, y así como nace **GNU/Linux** que **es básicamente un sistema operativo que utiliza como kernel a Linux y tiene añadidos programas y herramientas que estaban escritos por o para el proyecto GNU desarrollado por Richard Stallman**. Por lo tanto, podemos decir que GNU/Linux es el nombre de un sistema operativo que utiliza el Kernel Linux con componentes heredados de GNU. Este sistema a su vez tiene una familia de distribuciones o versiones del sistema operativo.



¿Por qué Android no es GNU/Linux?

Recordemos que Linux es un Kernel libre y abierto, de manera que cualquier desarrollador puede tomarlo para crear su sistema operativo, el cual **no tiene por qué tener herramientas de GNU**. **Android** es un ejemplo de ello, Google decidió utilizar el Kernel (Linux) e incorporarle sus herramientas y bibliotecas. Esto hace que contenga Linux (kernel) sin ser GNU/Linux. Por lo tanto, aunque Android y GNU/Linux comparten una pequeña parte de su código que es el kernel o núcleo, **son sistemas operativos totalmente diferentes**. Sus librerías son diferentes, lo que hace que las aplicaciones de ambos sistemas se desarrollen de forma totalmente distinta, y que por lo tanto las de uno no sean compatibles con el otro.

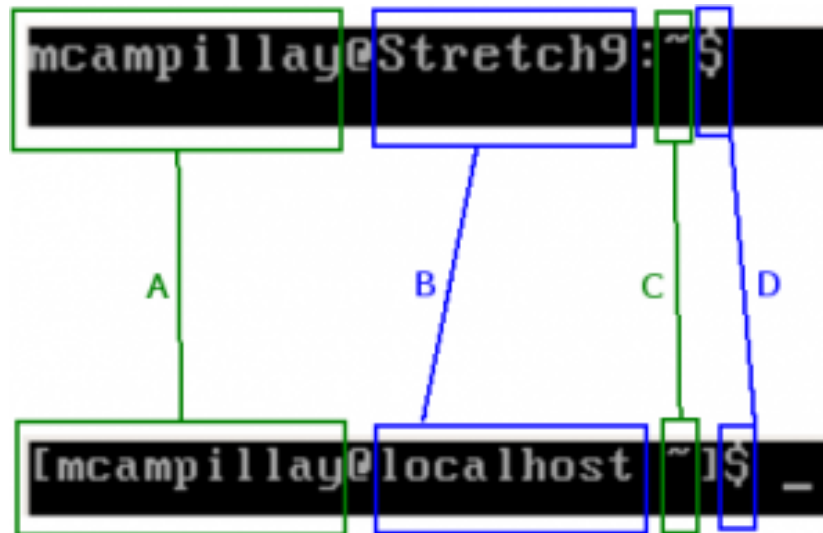


¿Qué es el Shell o interprete de comandos?

El Shell es un fichero u archivo ejecutable (programa) que actúa como un intermediario entre el sistema operativo y el usuario que introduce comandos en modo texto. **Su función es la de leer la línea de comandos, interpretar su significado, ejecutar el comando y después arrojar el resultado por medio de las salidas.** Existen varios shells, entre los que destacamos los siguientes: sh (llamada "*Bourne shell*"), bash ("*Bourne again shell*"), csh ("*C Shell*"), tcsh ("*Tenex C shell*"), ksh ("*Korn shell*") y zsh ("*Zero shell*"). Generalmente, sus nombres coinciden con el nombre del ejecutable. Cada usuario tiene un shell predeterminado, el cual se activará cuando el usuario entre en el sistema o cuando lo ejecute explícitamente. El shell predeterminado se especifica en el archivo de configuración **/etc/passwd**, en el último campo de la línea que corresponde al usuario. Es posible cambiar de shell durante una sesión, para lo cual solo se debe ejecutar el fichero ejecutable correspondiente, por ejemplo: `/bin/bash`.

De manera predeterminada, para la mayoría de los shells, **el prompt en la shell de GNU/Linux** es la información o el símbolo que se encuentra antes del cursor, es decir donde empezamos a escribir los comandos que ingresamos a la shell.

El shell nos presenta el prompt (que es cuando el sistema operativo queda esperando que se ingresen los comandos), en el cual dependiendo el tipo de usuario nos muestra una serie de datos en referencia al equipo en el que se está trabajando, a continuación se explicará que significa lo que se está visualizando:



Para que se vea de forma gráfica y sencilla, se realizaron las capturas de pantalla en las dos grandes ramas de distribuciones de GNU/Linux ; siendo la captura superior correspondiente a «*Debian*» y la inferior la de «*CentOS*» ; marcándose los distintos componentes del prompt y lo que representan:

A: Representa el nombre del usuario que entro al sistema.

B: Representa el nombre lógico del equipo sobre el que se está trabajando (es el hostname)

C: Representa el directorio dentro de la estructura del sistema de archivos donde se encuentra posicionado en este momento, el carácter que se ve en la imagen capturada está representando que se está posicionado en el directorio personal del usuario (home directory), en caso que se estuviera en otro lugar el sistema operativo lo indicaría en esa posición.

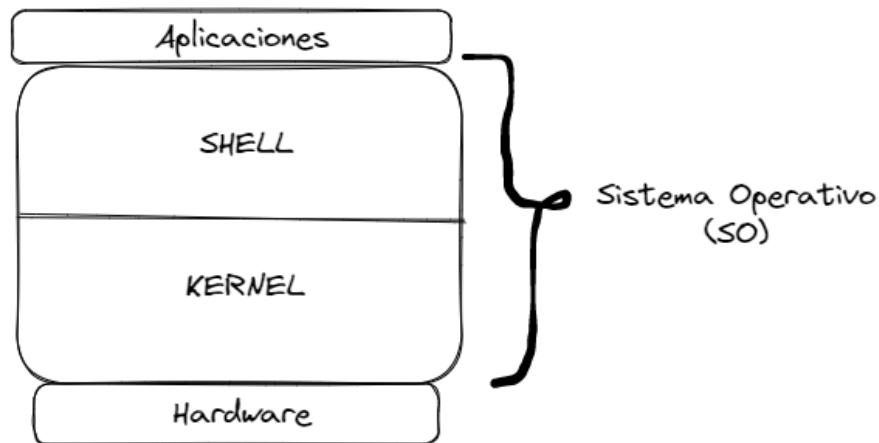
D: Representa el tipo de usuario con el que se está conectado, el símbolo “\$” está denotando que es un **usuario normal o común**, si por el contrario el símbolo fuera el “# (numeral)”, sería indicativo que se trata del **usuario root o super usuario**.

¿Qué es un Sistema Operativo?

Un sistema operativo (SO) es el programa que, después de ser cargado inicialmente en la computadora por un programa de arranque (programa boot), administra todos los demás programas de aplicación en una computadora. Los programas de aplicación hacen uso del sistema operativo realizando solicitudes de servicios a través de una interfaz de programa de aplicación (API) definida.

Además, los usuarios pueden interactuar directamente con el sistema operativo a través de una interfaz de usuario, como una interfaz de línea de comandos (CLI) o una IU gráfica (GUI).

Un SO consta de un Kernel y un Shell:



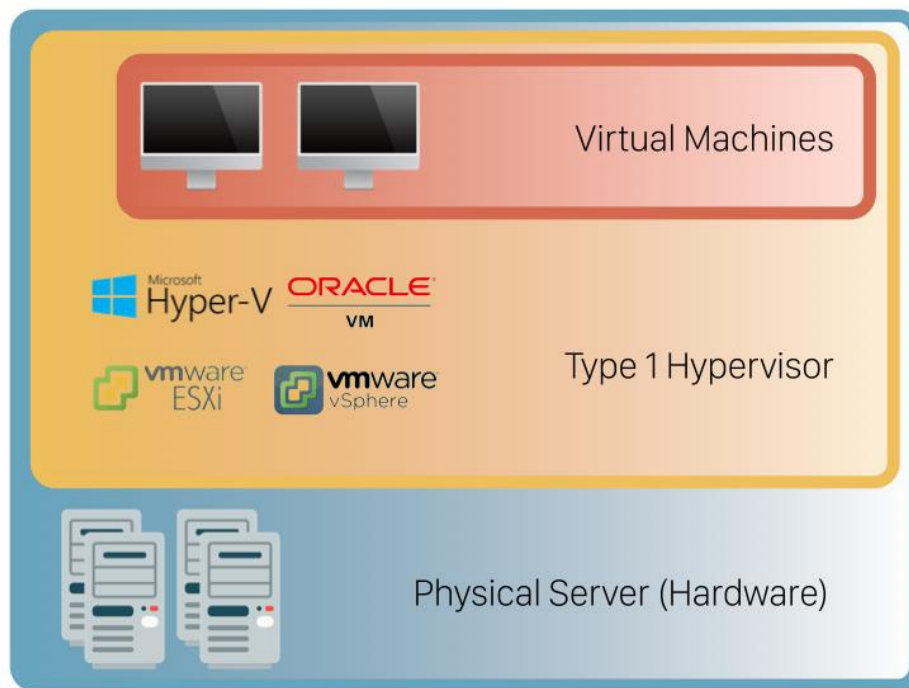
¿Qué es la virtualización?

La virtualización es una tecnología que se puede usar para crear representaciones virtuales de servidores, almacenamiento, redes y otras máquinas físicas. El software virtual imita las funciones del hardware físico para ejecutar varias máquinas virtuales a la vez en una única máquina física. Las empresas recurren a la virtualización para utilizar sus recursos de hardware de manera eficiente y obtener retornos mayores de sus inversiones. También potencia los servicios de computación en la nube que ayudan a las organizaciones a administrar la infraestructura de manera más eficaz. Los dos conceptos más importantes para entender qué es la virtualización son los de anfitrión e invitado. Ambos conceptos se refieren a nuestros sistemas operativos, y por lo tanto deberíamos hablar de sistema operativo anfitrión y sistema operativo invitado. El anfitrión es el sistema operativo del ordenador en el cual instalamos nuestro programa de virtualización y que asignará o prestará determinados recursos de hardware a la máquina virtual que creemos. El invitado es el sistema operativo que instalamos en el ordenador virtual que hemos creado, mediante nuestro programa de virtualización y al cual hemos asignado determinados recursos para funcionar.

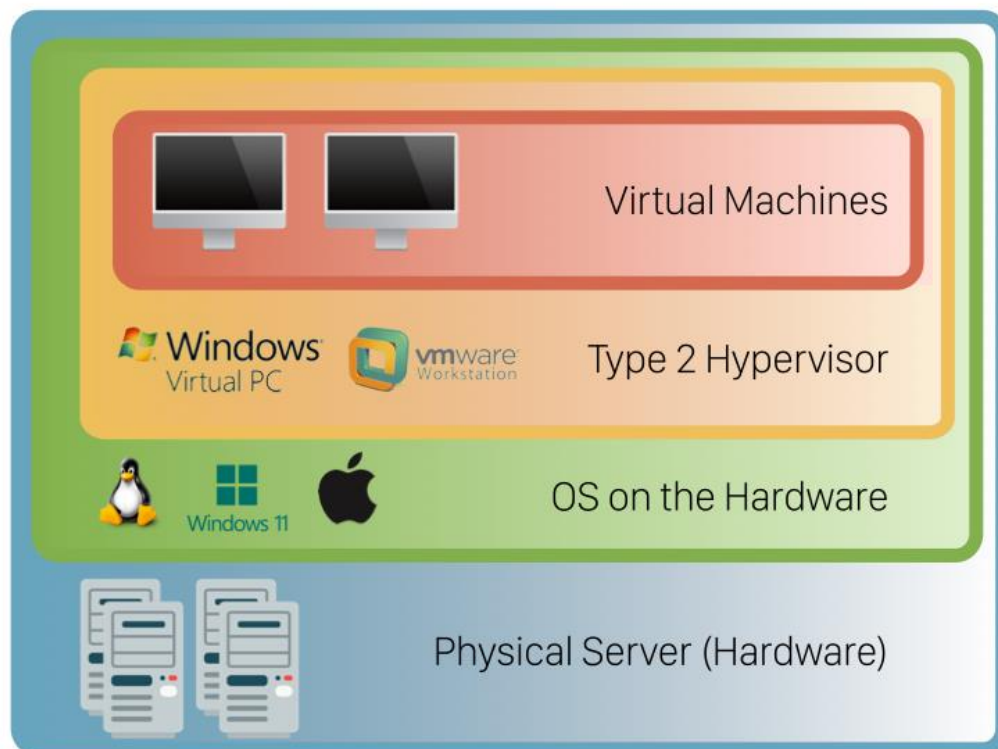
¿Qué es un hipervisor?

Un **hipervisor**, conocido también como monitor de máquinas virtuales, es un proceso que crea y ejecuta máquinas virtuales. Un hipervisor permite que un ordenador host preste soporte a varias máquinas virtuales invitadas mediante el uso compartido virtual de sus recursos, como la memoria y el procesamiento. En general, hay dos tipos de hipervisores. Los **hipervisores de tipo 1**, denominados «hipervisores bare metal», se **ejecutan directamente en el hardware del host**. Los **hipervisores de tipo 2**, denominados «alojados», **se ejecutan como un software sobre un sistema operativo**, como otros programas informáticos. **Básicamente un hipervisor es un programa que nos servirá para utilizar la virtualización o para virtualizar.**

Ejemplo de hipervisor tipo 1: VMware vSphere con ESX / ESXi como parte del paquete

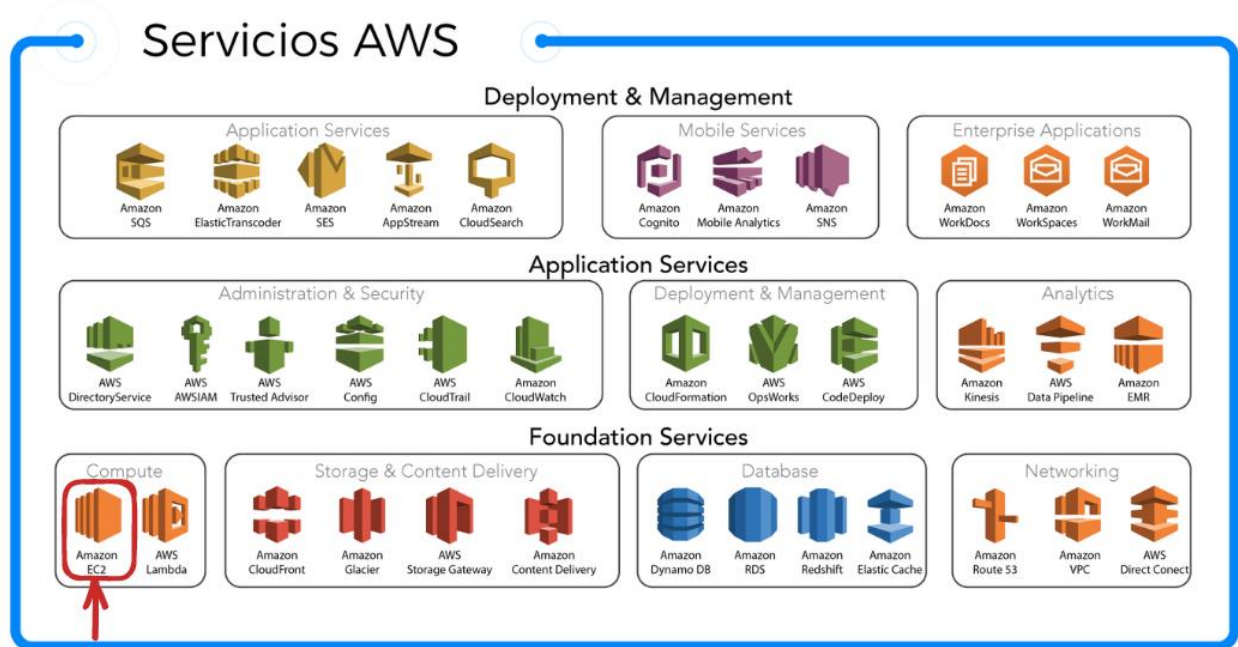


Ejemplos de hipervisor tipo 2: **Oracle VM VirtualBox**, VMware Workstation Pro



¿Qué es Amazon EC2?

Amazon Elastic Compute Cloud (Amazon EC2) proporciona capacidad de computación escalable en la nube de Amazon Web Services (AWS). El uso de Amazon EC2 elimina la necesidad de invertir inicialmente en hardware, de manera que puede desarrollar e implementar aplicaciones en menos tiempo. **Puede usar Amazon EC2 para lanzar tantos servidores virtuales como necesite, configurar la seguridad y las redes, y administrar el almacenamiento.**



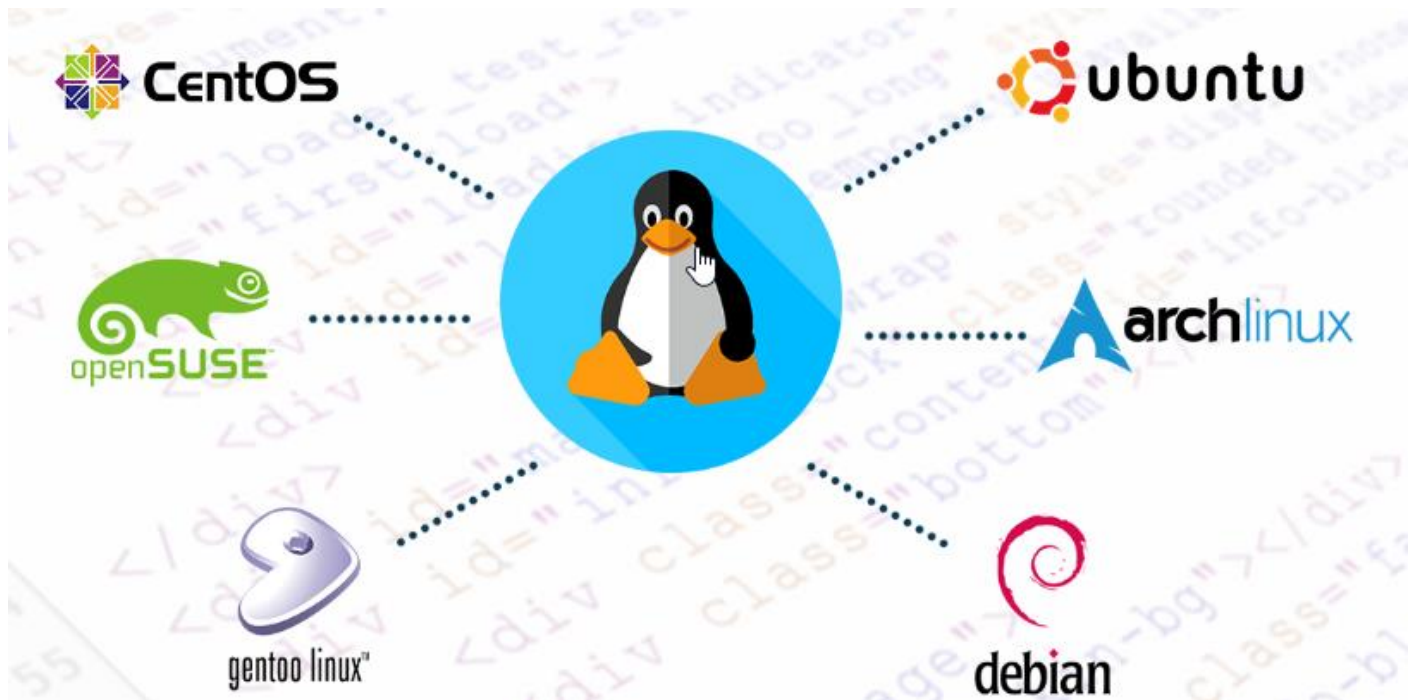
¿Qué es una distribución de Linux?

Básicamente una distribución de Linux es tomar el kernel Linux y añadir **programas y herramientas para satisfacer las necesidades de un grupo específico de usuarios**

Por ejemplo, si al kernel Linux le añadimos programas y herramientas del proyecto GNU, diremos que es una distribución GNU/Linux.

Ejemplos de Distribuciones GNU/Linux:

- Debian
- Ubuntu
- Linux Mint
- Red Hat Enterprise Linux
- openSUSE
- CentOS
- Fedora
- Kali Linux



Formas de instalar o utilizar Ubuntu Server 22.04

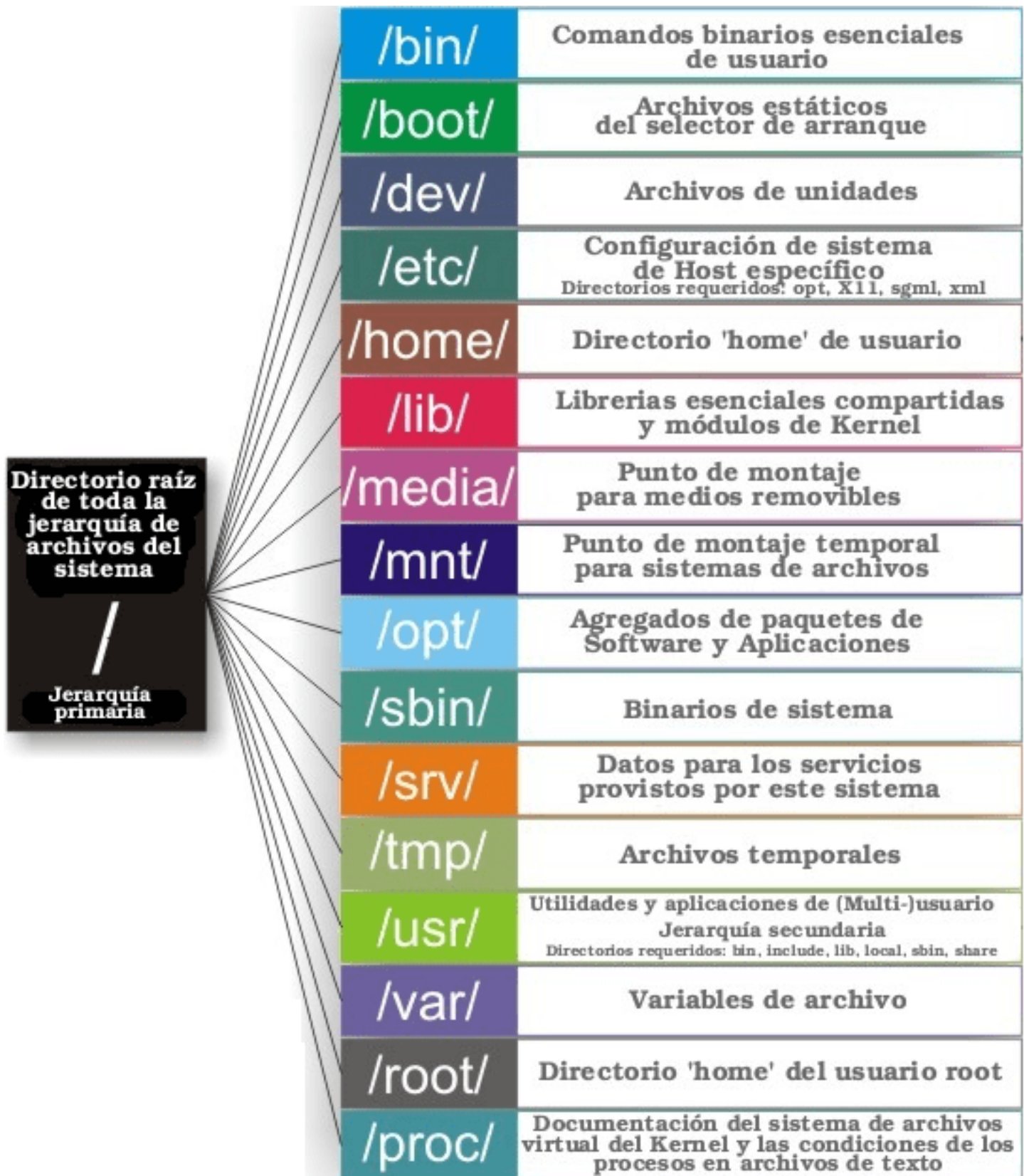
1. Virtualizando la distribución de GNU/Linux en un hipervisor tipo 1 o 2
2. Utilizando servicios en la Nube (ejm: EC2 de AWS)

La forma que utilizaremos nosotros para desarrollar el curso será la forma 1, es decir utilizaremos un hipervisor tipo 2 (VirtualBox) para instalar Ubuntu Server 22.04 en una máquina virtual, este proceso ya fue detallado y explicado en la guía “**Preparación del Entorno de Laboratorio Linux**”.

Estructura de Directorios en Linux:

Linux al igual que Unix, organiza la información del sistema en una estructura de árbol jerárquico de directorios compuesta de ficheros. Esta estructura se forma mediante un sistema de ficheros raíz (file system root) y un conjunto de sistemas de ficheros montables. Un sistema de ficheros, o file system, es una estructura de directorios completa. **Para poder utilizar un sistema de ficheros hay que montarlo;** o sea, enlazarlo a la estructura de directorios ya existente. Los sistemas de ficheros se montan automáticamente cada vez que se inicia el sistema operativo. Cuando un usuario se conecta al sistema, se encuentra un único árbol de directorios formado por los distintos sistemas de ficheros que se encuentran montados en ese instante.

Nota: Fichero y Archivo son lo mismo en el contexto de la informática.



¿Qué es UNIX?

UNIX es un sistema operativo que fue desarrollado durante la década de los 70s por la compañía AT&T. Este sistema operativo fue diseñado para cumplir ciertos requisitos que no comparte con otros sistemas de la época, **tenía características como la portabilidad, multi usuario y multi tarea**, motivos por los que fue escrito en el lenguaje de programación C. Dado el creciente mercado informático de la época, la creación de UNIX con esas bases fue todo un éxito, ya que contaba con proyección de futuro y a unas grandes empresas detrás de su desarrollo. Esto implicó que en UNIX se fueran unificando diferentes tecnologías y creando estándares, que luego serían usados para el lanzamiento y desarrollo de sistemas operativos (SO) basados en UNIX. **UNIX es el punto de partida para numerosos sistemas operativos.** Es la parte que tienen en común el sistema original con las distintas distribuciones que se han ido lanzando basadas en Linux, por ejemplo. De esta manera, GNU/Linux es un SO basado en UNIX, y desde Linux se derivan numerosas distribuciones que seguramente conozca (Ubuntu, Fedora, Mint, Debian, etc). Así pues, UNIX es el sistema operativo original del que se ha tomado una base para crear GNU/Linux (entre otros), y a partir de Linux se personalizan diferentes sistemas operativos para crear las distribuciones, que ya son versiones específicas que indagan en ciertos aspectos. Por ello, UNIX es una base de código propietario, es decir, no libre, de la cual se extrae y utiliza parte de él para crear código libre. Las distintas distribuciones basadas en Linux (y, por lo tanto, en UNIX), se solidarizan con el usuario sin conocimientos en programación, ya que ofrecen interfaces visuales (GUI) para que el usuario medio pueda realizar todas o casi todas las instrucciones mediante interfaz y no mediante líneas de comandos.

¿Qué es un repositorio en Linux?

Un repositorio es una lista de programas, generalmente siempre actualizada, que nos permite buscar y descargar fácilmente todo tipo de programas y herramientas en nuestra distribución. Los repositorios de software son servidores que indican al gestor de paquetes (por ejemplo, APT) dónde buscar los programas. Cuando instalamos **Ubuntu**, la distro instala una serie de repositorios pertenecientes a distintos canales:

- **Main:** el repositorio instalado por defecto. Consiste en software libre FOSS que puede ser distribuido libremente y sin restricciones.
- **Universe:** incluye software gratis y de código abierto, pero no puede garantizar las actualizaciones regulares de seguridad.
- **Multiverse:** incluye software que es de la FOSS. Debe ser el usuario quien analice las licencias y decida si puede, o no, usar estos programas de forma legal.
- **Restricted:** dentro de este canal podemos encontrar software de código cerrado. Es utilizado, por ejemplo, para distribuir los drivers de NVIDIA, entre otros muchos.
- **Partner:** incluye software propietario que ha sido empaquetado por Canonical.

Y además de estos repositorios, también podemos añadir repositorios de terceros a nuestra distro Linux para instalar otro tipo de software de forma segura.

Los repositorios de Linux son contenedores de paquetes y aplicaciones, estos están montados sobre servidores que se encuentran en internet. Dependiendo del tipo de repositorio y de la importancia de la organización que lo mantiene, disponen de mayores o menores prestaciones a la hora de interactuar con ellos.

Un repositorio es un servidor accesible mediante internet que almacena paquetes y programas para que nosotros los podamos descargar e instalar en nuestra distribución GNU/Linux. Cada una de las distribuciones GNU/Linux dispone de sus propios repositorios en los que se hallan los programas que nosotros podemos instalar en nuestro equipo. Aparte de los repositorios propios de cada una de las distribuciones, también podemos añadir y usar repositorios de terceros que contendrán versiones más actuales del software que tenemos instalado o programas que no han incluido los creadores de la distro que usamos.

Existen dos tipos de repositorios:

Repositorios oficiales

Contienen las aplicaciones que su distribución soporta, estos servidores de repositorios son mantenidos por la distribución que desarrolla esas aplicaciones almacenadas y gestionadas desde dicho contenedor. Se realizan revisiones de forma periódica para comprobar que los paquetes gestionados se encuentran en estado óptimo y no representan un riesgo en la seguridad del equipo cliente que lo instale.

Repositorios no oficiales

Estos servidores de repositorios son mantenidos por comunidades de usuarios que desarrollan aplicaciones no oficiales. Esto no quiere decir que estas sean perjudiciales para tu equipo, pero sí que pueden serlo en algunos casos. Estos servidores no suelen disponer de protocolos que revisen el correcto estado de los paquetes almacenados y si no sabes muy bien lo que estás haciendo, puedes descargar una versión errónea o una aplicación que afecte al correcto funcionamiento de tu equipo.

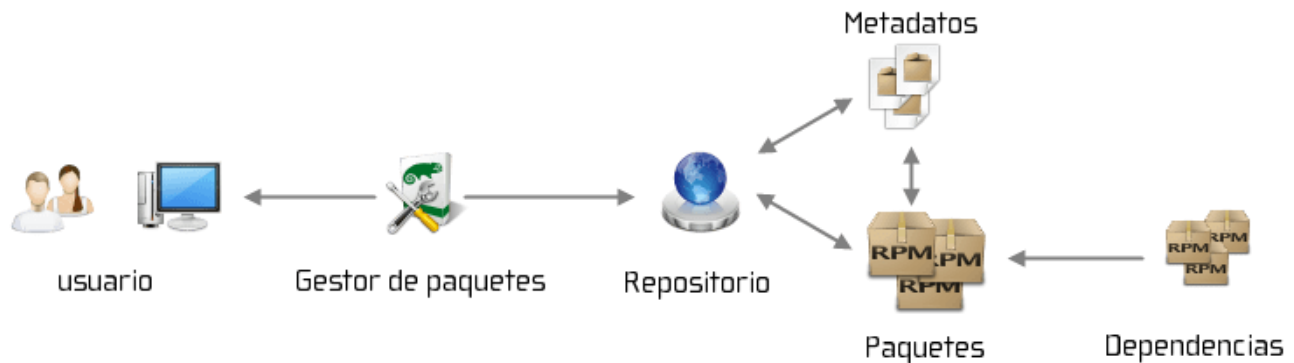
Ejemplo de instalación de software a través de un repositorio:

Si en nuestra distribución Linux queremos instalar el reproductor de vídeo VLC ejecutaremos el siguiente comando en la terminal:

```
sudo apt-get install vlc
```

En el momento de ejecutar el comando sucederá lo siguiente:

1. Con el gestor de paquetes apt nos conectaremos al repositorio de internet que contiene los paquetes que queremos descargar. Antes de empezar la descarga, mediante un par de claves asimétricas, un sistema de firmas y una función hash se comprobará que los paquetes a descargar provienen de un repositorio seguro y no han sido modificados por nadie.
2. Una vez realizada la comprobación se descargarán los paquetes y dependencias necesarias para instalar VLC.
3. Una vez descargados los paquetes se procederá a la instalación de los mismos.



Quien realiza el mantenimiento de los repositorios

Los repositorios de cada una de las distribuciones Linux son mantenidos, gestionados y actualizados por los siguientes actores:

1. El personal que creo y gestiona la distribución Linux
2. La comunidad del software Libre.

En el caso que usemos repositorios de terceros ajenos a nuestra distribución Linux, la gestión y el mantenimiento es realizado por:

1. El personal que gestiona el repositorio.

Nota: En el caso que usemos repositorios de terceros hay que asegurar que estén gestionados por personas confiables. En caso contrario podríamos estar instalando malware en nuestro sistema operativo.

Ventajas que proporcionan los repositorios

El uso de repositorios para instalar software proporciona las siguientes ventajas:

1. Instalar software a nuestro equipo de forma mucho más sencilla que en otros sistemas operativos. En Linux nunca tendremos la necesidad de buscar y descargar programas de Internet. Cabe recordar que muchas de las infecciones en Windows se producen al instalar software proveniente de sitios de dudosa reputación.
2. Tendremos la seguridad que el software instalado proviene de una fuente segura y esta libre de malware. Los repositorios de donde nos descargamos los programas disponen de las medidas de seguridad necesarias para asegurar que los programas descargados están libres de Virus y Malware.
3. El proceso de actualización del sistema operativo es mucho más sencillo. En el momento que se actualizan los repositorios podremos actualizar fácilmente nuestro sistema operativo mediante nuestro gestor de paquetes.

Desventajas que proporcionan los repositorios:

El uso de repositorios únicamente presenta la siguiente desventaja frente otros sistemas operativos.

1. En el caso que queramos instalar un programa o actualizar el sistema operativo precisamos de una conexión de internet. Esto es así porque descargar la totalidad de dependencias de un programa de forma manual exige demasiado tiempo.

En lo particular no parece ser ningún tipo de desventaja porque hoy en día todo el mundo dispone de conexión a Internet, pero se coloca a modo de mención.

¿Qué es un paquete en Linux?

Un paquete en Linux consiste en una colección de archivos que permiten la instalación de un programa y sus tareas relacionadas, como la búsqueda de dependencias, instalaciones previas, etc. Por ejemplo, podemos pensar en un programa popular como es GIMP. Este software de edición de imágenes se distribuye a través de un paquete. Dentro de él, todos los archivos requeridos para ser instalado y posterior ejecución. Además, se incluye un pequeño archivo que proporciona metadatos importantes sobre el programa.



La razón de distribuir software en base a paquetes es simple. Cuando hablamos de GNU/Linux nos referimos a un ecosistema muy amplio de distribuciones que cuentan con notables diferencias entre sí. Esto hace que no sea posible “garantizar” que un software funcione correctamente en una computadora determinada. El uso de paquetes resuelve este problema de interoperabilidad gracias al pequeño archivo que antes mencionamos con metadatos que actúa como un manifiesto de dependencias que deben cumplirse para que el software empaquetado se ejecute correctamente en un ordenador determinado.

¿Qué es un gestor de paquetes en Linux?

El uso de paquetes nos ahorra muchos dolores de cabeza a la hora de instalar o mantener nuevo software, pero en ocasiones el número de pasos que debemos dar puede consumirnos demasiado tiempo, sobre todo cuando tenemos que instalar varios programas. Para esto tenemos los gestores de paquetes. Se tratan de utilidades presentes en cada distribución que se encarga de automatizar

el proceso, también listan otros paquetes disponibles en el repositorio y muestran información sobre sus dependencias. Existen muchos, pero en general realizan la misma función básica de instalar y administrar nuevos programas, pero cada uno usa una arquitectura interna ligeramente diferente y diferentes interfaces de usuario para realizar las tareas centrales del administrador de paquetes.

Los sistemas comunes de administración de paquetes incluyen:

- **DPKG:** el administrador de paquetes base para distribuciones basadas en Debian.
- **Apt-get:** una interfaz que hace más amigable y añade funciones para el sistema DPKG, que se encuentra en las distribuciones basadas en Debian.
- **Aptitude:** Aptitude es una interfaz para APT para distribuciones basadas en Debian. Muestra una lista de paquetes de software y permite al usuario elegir de modo interactivo cuáles desea instalar o eliminar
- **Synaptic:** Synaptic es instalado por defecto en Debian en la versión de escritorio, es una herramienta gráfica para la gestión de paquetes basada en GTK+ y APT.
- **RPM:** el administrador de paquetes base que se encuentra en las distribuciones basadas en Red Hat, como Red Hat Enterprise Linux, CentOS y Fedora.
- **Yum:** un front-end para el sistema RPM, que se encuentra en las distribuciones basadas en Red Hat.
- **Dnf:** una interfaz más rica en funciones para el sistema RPM.
- **ZYpp:** se encuentra en SUSE y OpenSUSE.
- **Pacman:** el administrador de paquetes para distribuciones basadas en Arch Linux.

Independientemente del administrador de paquetes específico, el proceso de mantenimiento del software en una computadora basada en Linux es generalmente el mismo. Lanzas un catálogo de software que lee de uno o más repositorios (archivos de software optimizados para una plataforma determinada). Elija y elija qué software instalar o desinstalar a través del catálogo gráfico, o utilice una sesión de shell para ejecutar los comandos manualmente.

Comandos básicos de Linux

Antes, recordemos los tipos de PROMPT:

#	PROMPT DEL SUPERUSUARIO o ROOT (ADMINISTRADOR)
\$	PROMPT DEL CUALQUIER USUARIO COMUN (jeanleonardo)

Luego de ingresar con el usuario root (a través de **sudo su**) o un usuario simple los comandos básicos son:

Comando	Explicación y/o comentario
# who am i	QUIEN SOY YO
# who	QUIENES ESTAN CONECTADOS EN MI SERVIDOR
# w	w es un programa de Unix que muestra información sobre los usuarios que han iniciado sesión en el sistema y qué están haciendo
# hostname	NOMBRE DEL SERVIDOR QUE LE PUSISTE AL MOMENTO DE INSTALAR
# clear	Limpia la pantalla en cualquier momento

Comando	Explicación y/o comentario
# pwd	Donde estoy, en que directorio me encuentro
# date	Fecha y hora actual del sistema
# cal	Calendario del mes
# cal 7 1990	Mes=7=julio Año=1990
# hostnamectl	Para saber la versión de Ubuntu que tenemos y más información.
# ifconfig	Muestra la interface de red, el IP. Mascara, Broadcast Output: ens33 → 1era Tarjeta de Red ens34 → 2da Tarjeta de Red
# free	Muestra la memoria y sus características en Bytes
# free -h	Muestra la memoria y sus características en MB
# fdisk -l	Muestra las particiones y sus Sistemas de Archivos
# df	Muestra las particiones Linux con sus tamaños en Bytes
# df -h	Muestra las particiones Linux con sus tamaños en MBytes
# top	Muestra los procesos del sistema en tiempo real. Es similar al administrador de Tareas en Windows, pero en Linux.
# cd /proc	IR AL DIRECTORIO PROC
# ls	Ver el listado de los archivos y directorios (solo visibles). Ubique los siguientes directorios: cpuinfo meminfo version partitions
# more cpuinfo	Vemos el contenido del archivo cpuinfo (información del CPU) . Puede usar el comando “cat” para lo mismo.
# cat cpuinfo	Vemos el contenido del archivo cpuinfo (información del CPU)
# cat meminfo	Vemos el contenido del archivo meminfo (información de la memoria)
# cat version	Vemos el contenido del archivo version
# cat partitions	Vemos el contenido del archivo partitions hda hda1 hda2 → controladora IDE del disco (antiguo) sda sda1 sda2 → controladora SATA o SCASI del disco (actual)

Creación de Directorios:

Comando	Explicación y/o comentario
# cd /root	Vamos al directorio hogar del usuario root, utilizamos cd para desplazarnos entre directorios
# mkdir compras	Esto crea el directorio con el nombre “compras”
# ls	Listamos el contenido y vemos que efectivamente se creó el directorio “compras”
# cd compras	Ingresamos al directorio
# pwd	Verificamos la ruta actual (/root/compras)
# ls	Listamos el contenido y vemos que el directorio esta vacío

Comando	Explicación y/o comentario
# mkdir compra ventas mermas inventarios	Dentro del directorio compras creamos los directorios compra ventas mermas inventarios
# ls	Listamos el contenido y vemos que se crearon los directorios dentro del directorio compras
# ls -l	Listado del contenido con propiedades o con detalles

Creación de Directorios Compuestos:

Esto se da cuando queremos crear directorios cuyo nombre tenga un espacio.

Comando	Explicación y/o comentario
# mkdir "Archivos de Programas" "Mis Documentos"	Dentro del directorio compras creamos los directorios " Archivos de Programas " " Mis Documentos "
# ls -l	Listamos el contenido con detalle
# mkdir .presupuestos .planilla	Creación de directorios ocultos
# ls -la	Ver el listado de los archivos y directorios (visibles y no visibles (ocultos))

Creación de Archivos:

Comando	Explicación y/o comentario
# touch archivo.txt	Creación de un archivo con el nombre "archivo.txt"
# cat > archivo.txt	Creación de un archivo con el nombre "archivo.txt", la extensión ".txt" es opcional. Pulse <CTRL> <D> para grabar y salir al prompt
# ls -l	Listamos el contenido con detalle
# cat archivo.txt	Vemos el contenido del archivo de texto

Nota: Si desea corregir el archivo deberá usar un editor de programas se recomienda el editor vi.

Eliminar Archivos o Directorios:

Comando	Explicación y/o comentario
# rm -rf compra	Elimina un archivo o directorio sin confirmación, borra todo así tuviera algún contenido dentro en el caso de los directorios.
# ls	Listamos el contenido sin detalle
# rmdir ventas	A través de este comando solo borramos el directorio, solo si se encuentra vacío internamente, de lo contrario nos dará error.

Comando	Explicación y/o comentario
# rm -rf *	Elimina todo (menos los archivos ocultos), mucho cuidado al realizar esto en el directorio raíz / Elimina todo lo que está dentro de la ruta en la que te encuentras (pwd), por ejemplo, si estoy en /root/compras borraría todo lo que está dentro que es de /root/compras
# ls -a	Vemos que después de haber aplicado “rm -rf *” se borraron todos los visibles pero no los ocultos
# rm -rf .*	Elimina todo (incluyendo los ocultos excepto . y ..)

Nota: En Linux no existe “Papelera de Reciclaje” como en Windows, por lo tanto, tenga cuidado cuando elimina archivos y/o directorios.

Caracteres Especiales a utilizar en Linux:

Carácter Especiales	Combinacion de Teclas	Representacion
Comillas Invertidas	Alt + 96	`
Corchete Izquierdo	Alt + 91	[
Corchete Derecho	Alt + 93]
Pipeline (tuberia)	Alt + 179	
	Alt + 126	~
Circunflejo	Alt + 94	^
Barra Invertida	Alt + 92	\
Ampersand	Alt + 38	&

Desplazamiento entre Directorios:

El carácter punto (.) tiene un significado especial para el sistema de archivos e indica el directorio actual en el que te encuentras (el que se obtiene con pwd)

El carácter dos puntos seguidos (..) tiene un significado especial para el sistema de archivos e indica el directorio anterior con referencia al donde estas actualmente (pwd)

Comando	Explicación y/o comentario
# cd ..	Subir un nivel de directorio
# cd ../..	Subir 2 niveles de directorios
# cd ../../..	Subir 3 niveles de directorios
# cd /	Ir al directorio RAIZ o ROOT
# cd /root	Ir al directorio hogar del usuario ROOT
# cd /root/compras	Ir a una ruta específica

Copiar 1 Archivo:

Comando	Explicación y/o comentario
# mkdir /root/Escritorio	Creamos el directorio Escritorio dentro de /root
# cat > file3.txt	Creamos el archivo file3.txt Introduzca el texto: este es un archivo de prueba Pulse <CTRL> <D> para grabar y salir al prompt
# ls	Comprobamos que existe el archivo file3.txt
# cp file3.txt /root/Escritorio	Con el comando cp podemos copiar un archivo o directorio a un destino. En este caso copiamos el archivo file3.txt dentro del directorio "Escritorio" con ruta /root/Escritorio.
# cd /root/Escritorio	Vamos al destino o ruta /root/Escritorio
# cat Escritorio	Vemos el contenido del archivo
# ls -l	Listamos el contenido del directorio con detalle

Mover 1 Archivo:

Comando	Explicación y/o comentario
# mkdir /root/Descargas	Creamos el directorio Descargas dentro de /root
# mv file3.txt /root/Descargas	Con el comando mv podemos mover un archivo o directorio a un destino. En este caso copiamos el archivo file3.txt dentro del directorio "Escritorio" con ruta /root/Descargas
# ls	Comprobamos que el archivo se movió y ya no aparece.

Renombrar o cambiar el nombre de un directorio o archivo:

Comando	Explicación y/o comentario
# cd /root/Escritorio	Vamos al destino o ruta /root/Escritorio
# mkdir "Mis Documentos"	Creamos el directorio "Mis Documentos"
# ls	Listamos
# mv "Mis Documentos" "Tus Documentos"	El nombre actual es "Mis Documentos" y el nombre que quiero que tome será "Tus Documentos"
# ls	Listamos
# cd /home	/home es el directorio de los Usuarios, dentro de él aparecen los directorios hogar de cada usuario (común) que hayamos creado
# ls	

El editor vi:

Características:

- EL editor vi no es un procesador de textos
- El editor vi es un editor de programas en UNIX/LINUX
- No acepta caracteres especiales, ni vocales acentuadas.
- El editor vi está presente en todos los UNIX y todas las distribuciones LINUX, hasta en la versión mínima.
- VIM es la versión mejorada de vi y esta presente en las versiones actuales

Hay 2 modos de operación en el editor vi:

- Modo de entrada (cuando presionas la **tecla i**)
- Modo de comando (cuando presionas **"Esc"** generalmente)

Utilicé el modo de entrada para introducir texto en un archivo, mientras que el modo de comando se utiliza para introducir comandos que llevan a cabo funciones específicas de vi específicas. **El modo de comando es el modo estándar para vi.**

Creación de un archivo con vi:

Desde el prompt, ingrese:

Comando o instrucción	Explicación y/o comentario
# cd /root	Vamos al destino o ruta /root
# pwd	Ver la ruta actual /root → Confirmamos que estamos en el directorio hogar del root
# vi file Pulse la tecla INSERT <PARA INSERTAR TEXTO> e ingrese lo siguiente: <ul style="list-style-type: none"> ~ Nombres = Jean Leonardo ~ Apellidos = Estrada Roque ~ Distrito = Los Olivos ~ Ciudad = Lima ~ Provincia = Lima ~ Departamento = Lima ~ País = Perú ~ Equipo = UNI ~ Ocupación = Ingeniero 	"file" será el nombre de nuestro archivo y al entrar con vi el cursor se ubica en la 1ª Fila. Crea un archivo nuevo vacío Para entrar al modo inserción se presionan las teclas <i> o <a> Para salir del modo inserción e ir al modo instrucción o comando se presiona tecla <Esc>.
Presione la tecla ESC Tipee: :set nu	Para activar la numeración de las líneas, escriba :set nu y presione Retorno (o Enter)
Presione la tecla ESC Tipee: :set nonu	Para desactivar la numeración de las líneas, escriba :set nonu y presione Retorno (o Enter)

Guardar o Salir de un Archivo:

Comando o instrucción	Explicación y/o comentario
Pulse <ESC> <SHIFT>: q! Le debería quedar :q!	Salir sin guardar
Pulse <ESC> <SHIFT>: w Le debería quedar :w	Guardar sin salir
Pulse <ESC> <SHIFT>: w file.bk Le debería quedar :w file.bk	Guarda el archivo como .bk sin salir, guarda una copia del archivo original con el nuevo nombre especificado.
Pulse <ESC> <SHIFT>: wq! Le debería quedar :wq!	Guarda el archivo y salimos al prompt
# ls	Listamos y deberíamos tener los siguientes archivos: file file.bk

Iniciar con el cursor en una posición especificada dentro de vi:

Comando o instrucción	Explicación y/o comentario
# vi file	Ingresa y el cursor se coloca en la primera fila
# vi +6 file	Ingresa y el cursor se ubica en la fila 6
# vi + file	Ingresa y el cursor se ubica en la última Fila

Modo desplazamiento dentro del editor vi:

Dentro de vi puedo probar lo siguiente:

Combinación de teclas	Explicación y/o comentario
Esc 1 G	Ir a la Fila 1
Esc 10 G	Ir a la Fila 10
Esc j	Baja una fila
Esc k	Sube una fila
Esc h	Desplaza a la Izquierda
Esc l	Desplaza a la Derecha
Ctrl F	Avanza una pantalla
Ctrl B	Retrocede una pantalla

Modo inserción dentro del editor vi:

Combinación de teclas	Explicación y/o comentario
Esc i	Inserta texto
Esc a	Inserta texto
Esc o (letra o minúscula)	Inserta una fila bajo el cursor
Esc O (letra o mayúscula)	Inserta una fila sobre el cursor

Deshacer los cambios realizados dentro del editor vi:

Combinación de teclas	Explicación y/o comentario
Esc u	Deshacer

Modo de eliminación dentro del editor vi:

Combinación de teclas	Explicación y/o comentario
Esc x	Elimina un carácter (de derecha a izquierda)
Esc 5x	Elimina 5 caracteres (de derecha a izquierda)
Esc dd	Elimina una fila (las elimina hacia abajo no hacia arriba)
Esc 4dd	Elimina 4 filas (las elimina hacia abajo no hacia arriba)
Esc dw	Elimina 1 palabra hacia la derecha desde donde este el cursor y encuentre el espacio (considera palabra a algo separado por espacio)
Esc 3dw	Elimina 3 palabras hacia la derecha desde donde este el cursor y encuentre el espacio (considera palabra a algo separado por espacio)
Esc d\$	Elimina desde la posición inicial hasta el fin de la fila

Búsqueda de un carácter o palabra en un texto:

Comando o instrucción	Explicación y/o comentario
Esc 1 G (mayúscula)	Vamos a la fila 1 del archivo file
Esc /Lima y presione "Enter"	Búsqueda del texto "Lima" Acá buscaremos en todo el texto la palabra Lima, con n y N ya nos desplazamos hacia abajo o hacia arriba, respectivamente
Pulse n (next) para buscar coincidencias hacia abajo Pulse N (next) para buscar coincidencias hacia arriba	El cursor se ubica en el texto a buscar, si hay más coincidencias puede buscarlas o desplazarse hacia ellas con las teclas "n" y "N"

Búsqueda de un carácter o palabra en un texto y reemplazar su contenido:

Comando o instrucción	Explicación y/o comentario
Esc 1 G (mayúscula)	Vamos a la fila 1 del archivo file
Esc /Lima y presione "Enter"	Búsqueda del texto "Lima" Acá buscaremos en todo el texto la palabra Lima, con n y N ya nos desplazamos hacia abajo o hacia arriba, respectivamente
:s/Lima/Arequipa Lima: texto actual Arequipa: nuevo texto	Reemplazar Lima por Arequipa solo 1 vez
%s/Lima/Arequipa Lima: texto actual Arequipa: nuevo texto	Reemplazar Lima por Arequipa en todas las coincidencias que encuentre en todo el texto.

Nota: Él reemplaza viendo si hay alguna coincidencia con el texto actual desde la posición del cursor "hacia abajo", no mira "hacia arriba", si no lo encuentra hacia abajo no lo reemplaza así la coincidencia este hacia arriba del cursor

Copiar y pegar texto en el editor vi:

Comando o instrucción	Explicación y/o comentario
Esc 1 G (mayúscula)	Vamos a la fila 1 del archivo file
Esc 4 Y (YANK)	Copiar las primeras 4 filas
Ir al final del archivo y para pegar, pulse: p → Pega bajo el cursor (p minúscula) P → Pega sobre el cursor (p mayúscula)	

Copiar y pegar texto en el editor vi:

Si desea ver la ayuda de un comando, es decir su sintaxis, parámetros, argumentos etc. Introduzca el siguiente comando:

man Nombredelcomando

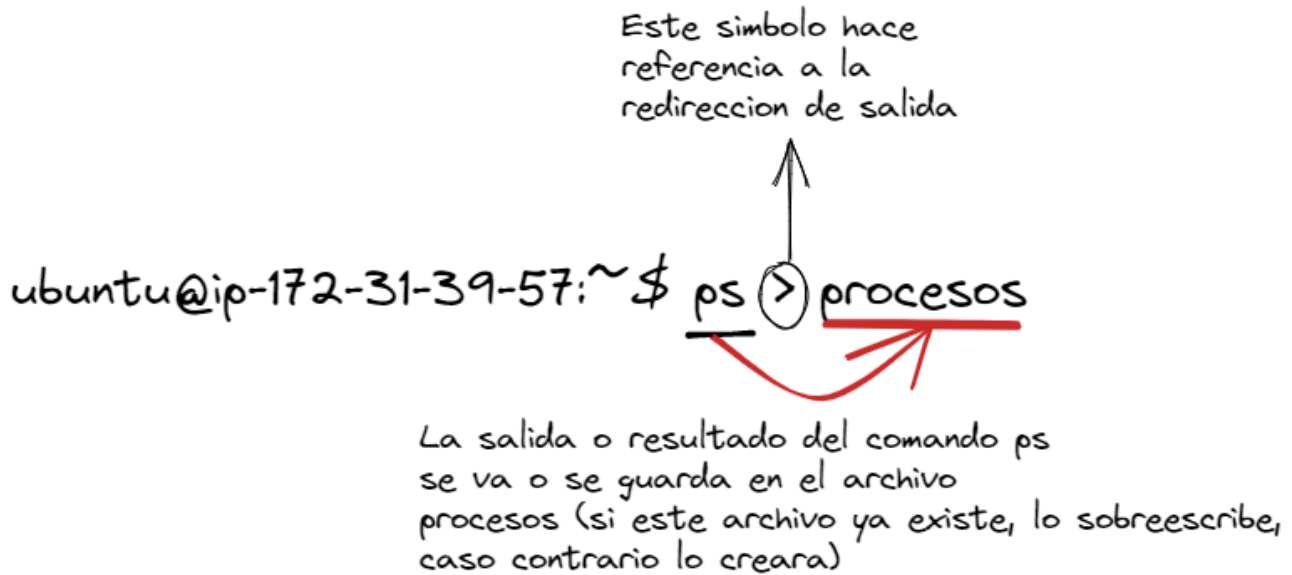
Ejemplos:

- man ls
- man pwd
- man vi
- man clear
- man cal

Nota: Para salir del man, pulse **q** <QUIT>

Redireccionamiento de Salida:

Si desea almacenar el resultado de un comando, puede redireccionarlo a un archivo.



Si queremos que la salida de otro comando también se redirija a un mismo archivo pero que se agregue al archivo y no sobrescriba el archivo borrando el contenido que ya estaba ahí presente, no usamos > sino >>.

Resumen:



Ejemplos:

Comando o instrucción	Explicación y/o comentario
# who > asistencia	Guarda la salida del comando who en el archivo asistencia, de no existir el archivo lo creará.
# ls	Listamos
# cat asistencia	Vemos el contenido del archivo asistencia, y será la salida del comando who

Apagar el equipo:

Desde una consola, ingrese cualquiera de los siguientes

- `init 0`
- `shutdown now`

Reiniciar el equipo:

- `init 6`
- `reboot`
- `shutdown -r now`

Cambiar la IP en su servidor Ubuntu Server 22.04:

Se sugiere realizar este cambio dentro de la misma virtual o desde MobaXterm, pero sabiendo que cuando cambie su IP y aplique los cambios, su sesión se cerrará.

Comando o instrucción	Explicación y/o comentario
<code># cd /etc/netplan/</code>	Vamos al destino /etc/netplan/
<code># ls -l</code>	Listamos con detalles
<code># cat 00-installer-config.yaml</code>	Vemos que en este archivo de texto podemos configurar nuestros parámetros de red, entonces para esto editaremos este archivo con el editor vi
<code># vi 00-installer-config.yaml</code>	Presione la tecla "i" o INSERTAR para poder ingresar texto y efectuar cambios. Edite según los parámetros de red de su LAN.
<code>:wq!</code>	Para salir de vi y guardar el archivo modificado)
<code># netplan apply</code>	Para que los cambios que realizamos en el paso anterior tengan efecto, tenemos que ejecutar este comando.
<code># ip a s</code>	Vemos nuestra nueva IP asociada a nuestra interface o adaptador de red.
<code># ip route show</code>	Para ver la IP de nuestro gateway ejecutemos el comando <code>ip route show</code> .

Nota: Realizando esto, ya puede ir a MobaXterm y probar una conexión remota con SSH hacia su servidor con la nueva IP.