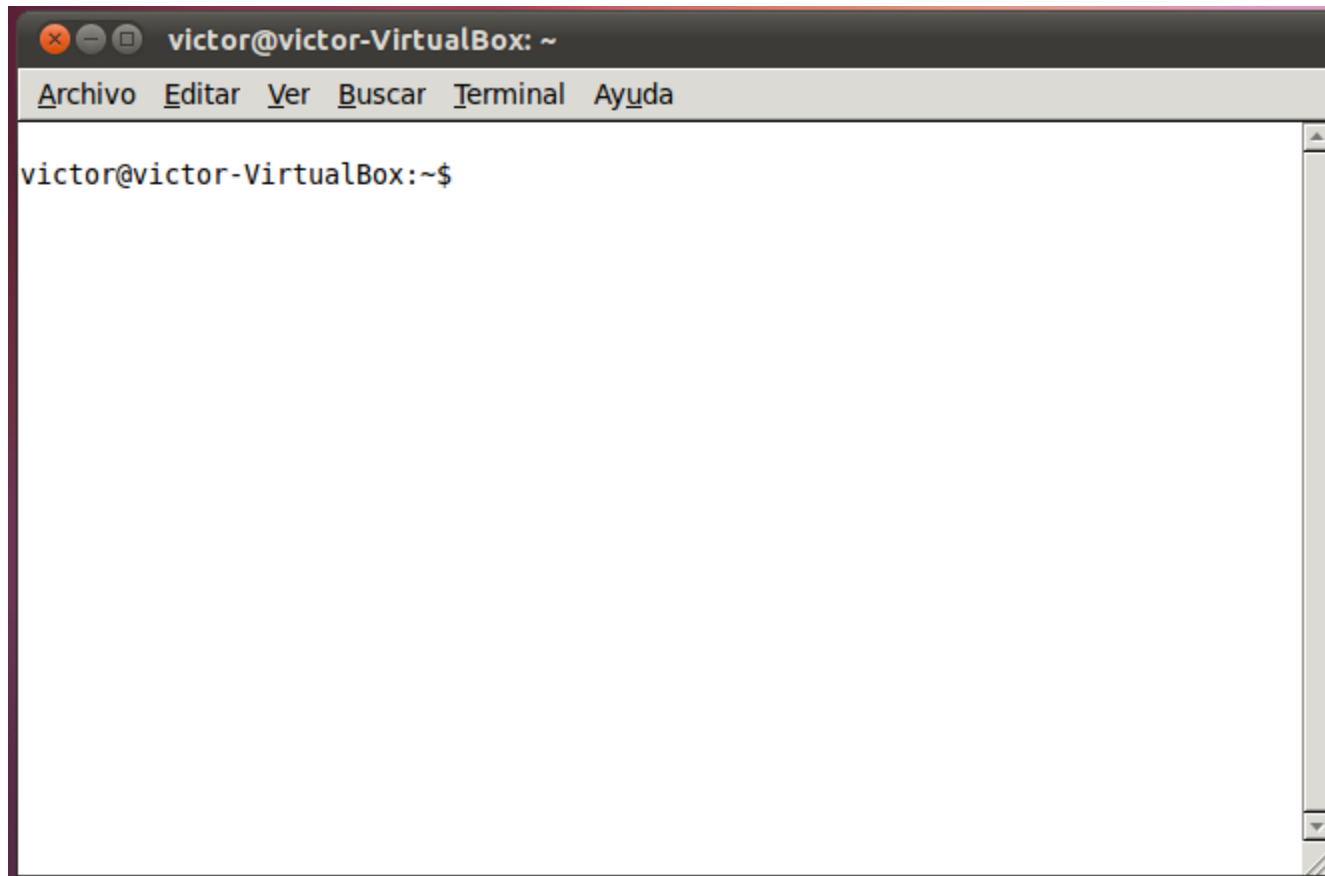


Comandos en Linux

Profesor: Victor Valenzuela

Consola



Autocompletado y revisión de historial

- ▶ TAB
- ▶ Flecha arriba y abajo



Información

- ▶ **man [comando]:** entrega información sobre el comando particular a usar. Para salir del programa, pulse “q”.
- ▶ **pwd:** visualiza el directorio actual (esto les permite saber su ubicación exacta).
- ▶ **history:** muestra el listado de comandos usados por el usuario.
- ▶ **lsmod:** lista los módulos cargados (controladores de un dispositivo o servicio).
- ▶ **lspci:** imprime información detallada sobre todos los puertos y dispositivos PCI en el sistema. (en slackware, si no está configurado fuera de root, este comando se tiene que ejecutar desde su directorio (/sbin/lspci))
- ▶ **date:** fecha y hora del sistema.
- ▶ **env:** se usa para imprimir una lista de las variables de entorno del sistema.
- ▶ **clear:** limpia la pantalla (en bash se puede limpiar la pantalla con ctrl+l)

Mostrar el contenido de una directorio

- ▶ **ls [modificadores][ruta de directorio]**: muestra el contenido del directorio especificado. Si no se especifica una ruta, muestra el directorio actual.
- ▶ **Variantes:**
 - **-l** : muestra además, propiedades y atributos de archivos.
 - **-la** : lista también los archivos ocultos de sistema (+propiedades y atributos).
 - **-lh** : lista también los tamaños de archivo en formato b, Kb, Mb, etc (+propiedades y atributos).
 - **| more** : muestra el directorio de forma paginada.

Rutas

- ▶ Dependiendo de la ubicación en la que se encuentren en la máquina, se puede indicar un archivo o directorio de diferentes maneras:
- ▶ Directo: **examples.desktop**
- ▶ Dirección Completa:
/home/victor/examples.desktop
- ▶ Dirección Relativa **./examples.desktop**
- ▶ **..** : un directorio más arriba
../examples.desktop (desde un directorio más abajo)

Movimiento dentro de directorios

- ▶ **cd [ruta de directorio]:** permite acceder a una directorio (tomando en cuenta la ubicación específica de la consola dentro del sistema).
- ▶ **cd .. :** sube a la directorio que contiene a la directorio actual.



Manipulación de archivos

- ▶ **touch [nombre de archivo]:** sirve para crear un archivo vacío.
- ▶ **mkdir [nombre de directorio]:** crea una directorio vacío.
- ▶ **cp [-dpR] [ruta de directorio/archivo] [ruta de destino][nuevo nombre]:** Copia archivo o directorio a la ruta especificada con un nuevo nombre. Si no se le pone un nuevo nombre, se mantiene el mismo.
 - **-R:** Copia un directorio recursivamente, salvo los ficheros especiales.
 - **-p:** Copia preservando permisos, propietario, grupos y fechas.
 - **-d:** Conserva los enlaces simbólicos como tales y preserva las relaciones de los duros.
 - **-a:** Lo mismo que -dpR .

Manipulación de archivos

- ▶ **rmdir [ruta de directorio]:** elimina un directorio (vacío).
- ▶ **rm [modificadores][ruta de directorio/archivo]:** permite eliminar un directorio o archivo.
 - **-r** : borra los ficheros de un directorio recursivamente (incluyendo el directorio).
 - **rm *.*[extensión]** : elimina todos los archivos con la extensión indicada

Comodines

<i>Comodines</i>	<i>coincidencias</i>
*	Cero o más caracteres
?	Exactamente un carácter
[abcde]	Exactamente un carácter de los listados
[a-e]	Exáctamente un carácter del rango dado
[!abcde]	Cualquier caracter que no esté listado
[!a-e]	Cualquier caracter que no esté en el rango dado
{debian,linux}	Exactamente una palabra entera en las opciones dadas

Manipulación de archivos

- ▶ **Mv [archivo o directorio a mover o renombrar] [ruta de directorio de destino][nuevo nombre]:** Mueve o renombra un archivo o directorio.

Manipulación de contenido en archivos

- ▶ `cat [archivo 1] [archivo 2]...[archivo n] > [archivo combinado]`: permite observar o combinar el contenido de un archivo, para luego redirigir lo mostrado por pantalla a un archivo (finalmente no muestra por pantalla)
- ▶ `>`: redirecciona la salida standard (salida por pantalla) hacia un archivo. Si el archivo existe y tiene información adentro, la reemplaza. Si el archivo no existe, lo crea.
- ▶ `>>`: redirecciona la salida standard a un archivo, pero la información se escribe al final del archivo.
- ▶ `<` : se usa para utilizar la información de un archivo como entrada de la orden especificada
 - “sort”, “more” (aunque es realmente requerido en muy pocos casos)

Procesos

- ▶ **ps -aux**: revisión de todos los procesos de la máquina, sin importar quien los controle (es una instantánea del momento en que se realiza el comando).
- ▶ **Top**: revisión de procesos en vivo. Para salir, usar “q”.
- ▶ **grep [expresión regular] [archivo]**: muestra las líneas en el archivo que contengan esa expresión regular.
- ▶ **ps -aux | grep [nombre del proceso o pid]** : muestra solo los procesos que coincidan con el criterio dado.
- ▶ **Kill [pid]**: sigterm -> termina el proceso, pero permite que guarde su estado antes de hacerlo.
- ▶ **Kill -9 [pid]**: sigkill -> termina el proceso inmediatamente.

Permisos de archivo y directorio

- Linux dispone de 3 comandos que permite cambiar los permisos, el propietario y el grupo de un archivo y/o directorio respectivamente:
- **Comando chmod** : se utiliza para cambiar los permisos del archivo o directorio
`$ chmod [permisos] [archivo/directorio] [opciones]`
- **Comando chown** : se utiliza para cambiar el propietario del archivo o directorio
`# chown [nuevo usuario propietario] [archivo/directorio] [opciones]`
- **Comando chgrp** : utilizado para cambiar el grupo del archivo o directorio
`# chgrp [nuevo grupo] [archivo/directorio] [opciones]`

Parámetros (caracteres)

	Símbolo	Descripción
Identidades	u	Es el usuario propietario del archivo o directorio
	g	Es el grupo al que pertenece el archivo o directorio
	o	Otros usuarios, el resto del mundo, ni el propietario ni su grupo
	a	Todo el mundo – propietario, grupo y otros
Permisos	r	Acceso de lectura
	w	Acceso de escritura
	x	Acceso de ejecución
Acciones	+	Añade los permisos
	-	Elimina los permisos
	=	el único permiso

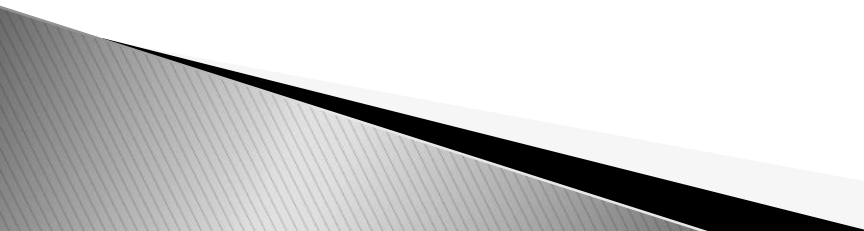
Parámetros (números)

Valor	Permisos	Descripción
0	---	El valor cero significa que no se han asignado permisos
1	--x	sólo se ha asignado el de ejecución
2	-w-	sólo permiso de escritura
3	-wx	permisos de escritura y ejecución
4	r--	sólo permiso de lectura
5	r-x	permisos de lectura y ejecución
6	rw-	permisos de lectura y escritura
7	rwX	permisos: lectura, escritura y ejecución

Ejemplos

- ▶ añadimos permiso de lectura y ejecución al grupo y eliminamos todos los permisos de ejecución a otros:
- ▶ **chmod g+rx,o-rwx foto1.png**
- ▶ **chmod 550 foto1.png**

Ejercicios

- Diríjase a su carpeta de usuario (desde este momento, no puede moverse de esta carpeta)
 - Crear un archivo (ej01.txt)
 1. Darle permisos de lectura y escritura para usuario
 2. Quitarle los permisos de lectura a otros y darle permiso de escritura al grupo
 3. Darle todos los permisos a otros
 4. Quitarle el permiso de ejecución a otros
 5. Quitarle todos los permisos al grupo
 6. Darle sólo permiso de ejecución al grupo
 7. Quitarle todos los permisos al usuario
- 

Ejercicios

1. Crear otro archivo (ej02.txt)
2. Crear una copia de el llamada "ej03.txt"
3. Crear una copia de el llamada "ej04.txt"
4. Renombrar "ej04.txt" a "ejx.txt"
5. Crear la carpeta "cosas"
6. Mover "ejx.txt" a "cosas"
7. Eliminar, dentro de la carpeta "cosas", todos los archivos que comiencen con la letra "e" y que tengan extensión "txt" o "doc"
8. Guardar el resultado de un ls en la carpeta de usuario en "ej02.txt"
9. Guardar el resultado de un ls de la carpeta cosas en "ej03.txt"
10. Crear un archivo llamado "total.txt" dentro de la carpeta cosas
11. Poner el contenido de "ej02.txt" y "ej03.txt" en "total.txt", usando "cat"
12. Leer el contenido de "total.txt"
13. Borrar cada uno de los archivos y carpetas creadas

Ejercicios

- ▶ Alt+f2 => cambio a otra consola
- ▶ Abrir vi o top
- ▶ Volver a consola principal (Alt+f1) Encontrar su pid (del programa que se abrió)
- ▶ Matar el proceso y revisar qué pasó en consola inicial
- ▶ Probar “killall -u [su nombre de usuario]”
- ▶ Compare el resultado al usar “killall -s KILL -u [su nombre de usuario]”