

Capítulo 1

Tutorial de Linux

1.1. Ingresando al sistema

Antes de comenzar, si lo que busca es documentación, le recomiendo que visite el sitio *The Linux Documentation Project* en <http://www.tldp.org/>.

Para poder trabajar en un sistema Linux, usted debe proveer un usuario y contraseña. Hay dos modos básicos en los que corre un sistema: la rápida y sobria consola de texto o bien el modo gráfico que se ve mejor pero que consume más recursos del sistema. En estos días el modo por defecto es el gráfico, una vez que usted ingresa su usuario y contraseña se tomará unos segundos en entrar al ambiente gráfico. Si usted navega los menús se dará cuenta que muchas cosas se pueden hacer sin entrar a la consola. Para la mayoría de los usuarios el viejo método para operar con la computadora de “apuntar y clickear” le sirve, pero esta guía es para futuros administradores de red, y como tales, necesitarán una herramienta más poderosa que un ratón para enfrentarse a los problemas que se le presentarán en el día a día. Esta herramienta es la consola de texto. Cuando estamos en modo gráfico también podemos activar esta consola. Personalmente llevo años administrando sistemas operativos UNIX y creame, se lo digo por experiencia, 105 teclas nos dan mucho más opciones y flexibilidad para trabajar que tan sólo 2.

Al momento de presentarse en el sistema verá la siguiente línea de comandos en la pantalla, por favor introduzca nombre de usuario y pulse Enter. Luego su contraseña y pulse Enter nuevamente.

```
lab2 login: alumno
Password:
```

Si introduce un nombre de usuario o una contraseña incorrecta, se mostrará el siguiente mensaje:

```
Login incorrect
```

Una vez introducida correctamente la contraseña usted estará oficialmente en el sistema y listo para comenzar a trabajar.

1.2. Crear una cuenta de usuario

Para crear una cuenta de usuario usted puede usar alguna herramienta GUI (Grafic User Interface) como “User manager” ó bien, la línea de comandos:

1. Abra una consola como root. Si no es root escriba `su -` e ingrese la contraseña.
2. Escriba `useradd` seguido por un espacio en blanco y el nombre de la cuenta de del usuario; por ejemplo `useradd jhendrix` para Jimi Hendrix. Presione [Enter].

3. Escriba `passwd` seguido por el nombre de usuario nuevamente. Por ejemplo `passwd jhendrix` y presione [Enter].
4. Ingrese la contraseña cuando aparezca `New password:` y presione [Enter].
5. Confirme la contraseña en `Retype new password:` y presione [Enter].

Usted debe ser precavido al momento de elegir su contraseña pues es la llave a su cuenta, por lo que debe ser único y fácil de recordar. Debería ser como mínimo 6 caracteres, puede combinar mayúsculas y minúsculas, así como también caracteres especiales y números. Trate de no elegir contraseñas simples, y si le es simple de recordar trate de variarlas tal que sean de alguna manera únicas, considere variaciones de una palabra como ser *G35710nD3R3d* para GestionDeRed, o mejor aún *g3\$nd7103Rd€*.

1.3. Documentación y ayuda

Hay muchos recursos disponibles para obtener la información que usted necesita. Están los manuales, documentos que detallan como se usan aplicaciones importantes y archivos; las páginas INFO que nos brindan la información de acuerdo al contenido pudiendo elegir distintas opciones en menús; y los archivos “help” que puede encontrar en la interfaz gráfica. Todo esto se instala por defecto y de no estar, puede ser muy fácilmente instalado.

1.3.1. Páginas Man

Recuerdo que a mis inicios mi buen amigo Maximiliano Eschoyez escribió grande en el pizarrón la siguiente frase previo a comenzar su curso de \LaTeX :

MAN IS YOUR FRIEND

De él he tomado esto y como verá en clase es lo primero que escribo cada vez que explico este tema. Cada comando de Linux tiene asociado un manual bien estructurado y documentación. Tome por costumbre consultarlo antes de preguntar.

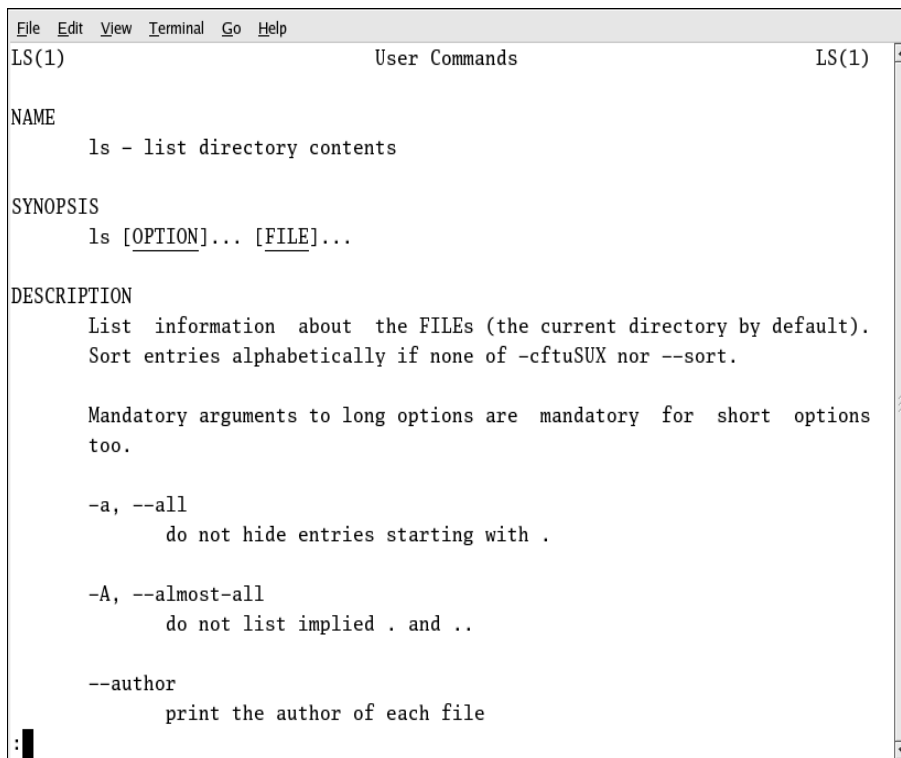
1.3.2. El comando man

Puede acceder vía la línea de comandos usando el comando `man` y el nombre del ejecutable; por ejemplo para acceder al manual del comando `ls` usted debe ingresar `man ls`.

El campo NAME muestra el nombre del ejecutable y una breve descripción. El campo SYNOPSIS muestra el modo más común de usar el ejecutable, como qué opciones están declaradas y qué tipo de entradas (archivos o valores) soporta el ejecutable. El campo DESCRIPTION describe las opciones disponibles y los valores asociados. El campo See Also nos refiere a archivos, términos y programas asociados.

Para navegar en el manual use las teclas \uparrow y \downarrow ó la barra espaciadora para pasar una página o [B] para retroceder una página. Para salir [q]. Busque alguna dentro del man con / y luego la palabra o frase seguida por [Enter] y todas las instancias de la palabra o frase serán resaltadas.

Si le quedan dudas consulte el manual del man, simplemente escribiendo `man man`.



```
File Edit View Terminal Go Help
LS(1) User Commands LS(1)
NAME
    ls - list directory contents
SYNOPSIS
    ls [OPTION]... [FILE]...
DESCRIPTION
    List information about the FILES (the current directory by default).
    Sort entries alphabetically if none of -cftuSUX nor --sort.

    Mandatory arguments to long options are mandatory for short options
    too.

    -a, --all
        do not hide entries starting with .

    -A, --almost-all
        do not list implied . and ..

    --author
        print the author of each file
:
```

Figura 1.1: Página de manual

1.4. Saliendo de la sesión

Si está en modo gráfico, diríjase a Main Menu \Rightarrow Logout. Si está en la consola escriba `logout`, `exit` o bien `[Ctrl]-[d]`.

1.5. Apagando el sistema

Si está en modo gráfico, diríjase a Main Menu \Rightarrow Shutdown. Si está en la consola escriba `halt`, `shutdown -hP` o bien `init 0`.

1.6. Determinar el directorio actual

Una vez que uno se empieza a mover en el árbol de directorios es fácil olvidarse en donde uno está parado. Para determinar la localización exacta del directorio escriba el comando `pwd`.

En el ejemplo el usuario `sam` está parado en su directorio `home`. Una regla fácil para recordar los comandos es saber de donde vienen, `pwd` viene de “print work directory”.

Cuando usted ejecutó el comando `pwd`, Linux le devolvió el “path” completo del directorio en el que usted está trabajando en el prompt, esta respuesta se denomina *standard output* y puede ser redireccionada como veremos más adelante.



```
File Edit View Terminal Go Help
[sam@Halloween sam]$ pwd
/home/sam
[sam@Halloween sam]$
```

Figura 1.2: Comando pwd

1.7. Cambiando de directorio

Cambiar de directorios es fácil siempre y cuando usted sepa donde está (su directorio actual) y como se relaciona al cual quiere dirigirse.

Para cambiar de directorios el comando es `cd`. Si usted ejecuta el comando `cd` a secas, siempre lo llevará a su directorio home. Moverse a cualquier otro directorio requiere el nombre del path.

```
[alumno@universidad ~]$ pwd
/home/alumno
[alumno@universidad ~]$ cd Scripts/
[alumno@universidad Scripts]$ pwd
/home/alumno/Scripts
[alumno@universidad Scripts]$ cd Perl/
[alumno@universidad Perl]$ pwd
/home/alumno/Scripts/Perl
```

El comando `cd ..` le dice a su sistema que suba un directorio. Para subir dos directorios debe usar `cd ../../`

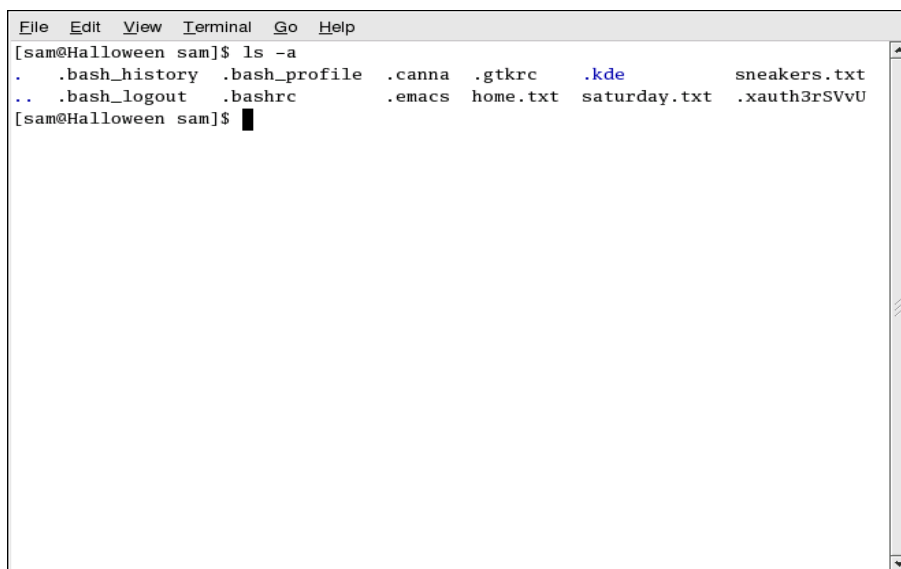
```
[alumno@universidad Perl]$ cd ..
[alumno@universidad Scripts]$ pwd
/home/alumno/Scripts
[alumno@universidad Scripts]$ cd ../../
[alumno@universidad home]$ pwd
/home
[alumno@universidad home]$
```

1.8. Ver el contenido del directorio

Ahora que ya sabe como cambiar de directorio es tiempo de que aprenda a ver el contenido de los mismos. Para ello usaremos el comando `ls`.

Hay muchas opciones que se le pueden agregar a este comando, ejecutado a secas no mostrará todos los archivos, pues existen archivos ocultos que solo podrá ver con opciones adicionales.

Los archivos ocultos son la mayoría archivos de configuración de programas, manejadores de ventanas, shells y más. La razón de ocultarlos es prevenir su manipulación accidental por parte del usuario, además normalmente cuando busca un archivo no busca justamente estos, así que mantenerlos ocultos aclara la pantalla cuando usamos la línea de comandos.



```
File Edit View Terminal Go Help
[sam@Halloween sam]$ ls -a
. .bash_history .bash_profile .canna .gtkrc .kde sneakers.txt
.. .bash_logout .bashrc .emacs home.txt saturday.txt .xauth3rSVvU
[sam@Halloween sam]$
```

Figura 1.3: Comando ls -a

Esta es una lista de las opciones comúnmente usadas del comando `ls`:

- `-a` – all. Lista todos los archivos en un directorio incluyendo los ocultos.
- `-l` – long. Lista detalles de los archivos, incluye permisos, dueños, grupos, tamaño, fecha de creación, si el archivo es un link simbólico y donde apunta.
- `-F` – file type. Agrega un símbolo al final de cada listado. Un directorio es `/`; un link simbólico se indica con `@`; y `*` indica ejecutable.
- `-r` – reverse. Lista los contenidos de atrás para adelante.
- `-R` – recursive. Lista los directorios de forma recursiva.
- `-S` – size. Ordena por tamaño.

1.9. Manipulando archivos

Linux tiene una utilidad que nos permite concatenar archivos y puede ser usada para mostrar información de sistema, es el comando `cat`. Este comando nos muestra en el standard output el contenido de un archivo.

```
[alumno@universidad ~]$ cat file.txt
hello world
[alumno@universidad ~]$
```

1.9.1. Redireccionar la salida estandar

Una redirección es hacer que el shell cambie lo que considera standard input o bien decidir donde el standard output debiese ir. Para redireccionar el standard output se usa el símbolo `>`. Si colocamos `>` luego del comando `cat` (o cualquier aplicación que escriba en el standard output) direccionará la salida al archivo que sigue al símbolo.

Por ejemplo, `cat` por si sólo da como salida todo lo que usted escriba, como si estuviese repitiendo.

```
[alumno@universidad ~]$ cat
Me repito a mi mismo.
Me repito a mi mismo.
```

Para redireccionar entonces la salida del `cat` a un archivo:

```
[alumno@universidad ~]$ cat > aforismo0.txt
El loro plagia las palabras, pero el que está preso es el canario.
```

Presione [Enter] y luego [Ctrl]-[d] para salir del `cat`.

Notará que esta vez no se repitió el texto. Eso es por que el standard output se redirigió al archivo y esa redirección creó un nuevo archivo llamado *aforismo.txt*. Tenga mucho cuidado al usar las redirecciones, fácilmente podría usted sobre escribir un archivo existente.

Ponga atención en el siguiente ejemplo:

```
[alumno@universidad ~]$ cat > aforismo1.txt
Una palabra puede herir. Pero un martillazo es feroz.
```

[Ctrl]-[d] nuevamente para salir del `cat`. Y ahora usaremos `cat` para concatenar los dos archivos en un nuevo archivo llamado *aforismos.txt*:

```
[alumno@universidad ~]$ cat aforismo0.txt aforismo1.txt > aforismos.txt
[alumno@universidad ~]$ cat aforismos.txt
El loro plagia las palabras, pero el que está preso es el canario.
Una palabra puede herir. Pero un martillazo es feroz.
```

1.9.2. Anexar la salida estandar

Se puede usar la redirección para agregar más contenido a un archivo existente. Usted usará ahora `>>` para anexar información al final de un archivo.

```
[alumno@universidad ~]$ cat aforismo0.txt aforismo1.txt >> \
aforismos.txt
[alumno@universidad ~]$ cat aforismos.txt
El loro plagia las palabras, pero el que está preso es el canario.
Una palabra puede herir. Pero un martillazo es feroz.
El loro plagia las palabras, pero el que está preso es el canario.
Una palabra puede herir. Pero un martillazo es feroz.
```

1.9.3. Redireccionar la entrada estandar

También se puede redireccionar el standard input, esto significa indicarle al shell que usted quiere que un archivo sea leído como entrada de un comando.

```
[alumno@universidad ~]$ cat < aforismos.txt
El loro plagia las palabras, pero el que está preso es el canario.
Una palabra puede herir. Pero un martillazo es feroz.
El loro plagia las palabras, pero el que está preso es el canario.
Una palabra puede herir. Pero un martillazo es feroz.
```

1.10. Sobre pipes y pagers

El *pipe* (la barra vertical) conecta la salida de un comando a la entrada de otro. Considere el comando `ls` por ejemplo, si el directorio tiene muchos archivos es difícil verlos bien por que saldrán fuera del alcance de la pantalla muy rápido. Una forma de hacerlo es usando conectarlo a un pager o paginador. Una utilidad para paginar es el `less`.

```
ls -al /etc | less
```

1.11. Más comandos para leer archivos de texto

Los archivos de configuración en Linux están en texto plano, así como también la mayoría de los archivos de logs, por lo que es muy importante que usted pueda manipular y leer archivos de texto. Veremos unos cuantos comandos más.

1.11.1. El comando `head`

El comando `head` puede ser muy útil por que se limita en las primeras n líneas de un archivos. Su modo de uso es el siguiente:

```
head <archivo>
```

Para leer las primeras 20 líneas:

```
head -20 <archivo>
```

1.11.2. El comando `tail`

Este comando le mostrará las últimas 10 líneas de un archivo de texto. También puede ser muy útil para leer archivos de log a medida que se van escribiendo con la opción `-f`.

```
tail -f /var/log/messages
```

1.11.3. El comando `grep`

Este comando es útil para encontrar ocurrencias de un string de texto en archivos. Por ejemplo, el siguiente comando sólo imprimirá en el standard output las ocurrencias del string “nameserver”:

```
grep nameserver /etc/resolv.conf
```

1.11.4. Comodines y expresiones regulares

Los comodines son símbolos especiales que puede usar para substituir letras, números y símbolos que hacen mucho más simple la tarea de encontrar directorios o archivos particulares en un largo listado de ellos al momento de examinarlos.

Para leer más de estos comodines lea el manual de `bash` (`man bash`).

1.12. Permisos y propiedades

Parte de los mecanismos de seguridad de Linux es que todos los archivos tienen un dueño, y éste le puede asignar permisos al mismo para permitir que determinado usuario o grupos de usuarios puedan ejecutarlo, leerlo u escribir en él.

```

$ ls -l /
total 104
dr-xr-xr-x   2 root root   4096 2010-04-24 09:24 bin
dr-xr-xr-x   5 root root   4096 2010-04-11 21:37 boot
drwxr-xr-x  20 root root   4180 2010-04-25 15:09 dev
drwxr-xr-x 129 root root  12288 2010-04-25 14:11 etc
drwxr-xr-x   4 root root   4096 2010-04-01 17:45 home
dr-xr-xr-x  12 root root   4096 2010-04-23 23:17 lib
dr-xr-xr-x   9 root root  12288 2010-04-24 09:24 lib64
drwx-----  2 root root  16384 2010-02-27 22:10 lost+found
drwxr-xr-x   2 root root   4096 2010-04-25 14:12 media
drwxr-xr-x   4 root root   4096 2010-04-01 16:01 mnt
drwxr-xr-x   2 root root   4096 2009-08-25 15:06 opt
dr-xr-xr-x 200 root root     0 2010-04-25 14:11 proc
dr-xr-x---  16 root root   4096 2010-04-21 21:42 root
dr-xr-xr-x   2 root root  12288 2010-04-24 09:24 sbin
drwxr-xr-x   2 root root   4096 2010-02-27 22:32 selinux
drwxr-xr-x   2 root root   4096 2009-08-25 15:06 srv
drwxr-xr-x  13 root root     0 2010-04-25 14:11 sys
drwxrwxrwt  31 root root   4096 2010-04-25 15:22 tmp
drwxr-xr-x  14 root root   4096 2010-02-27 22:53 usr
drwxr-xr-x  24 root root   4096 2010-03-31 00:35 var

```

La primera columna muestra los permisos; tiene diez casilleros. El primer casillero representa el tipo de archivo. Los nueve siguientes son en realidad tres grupos de permisos; el primer grupo corresponde al usuario, el segundo al grupo y el tercero al resto de los usuarios del sistema. Por ejemplo:

```
-rw-rw-r--
```

El primer ítem especifica el tipo de archivo, que puede ser uno de los siguientes:

- d - un directorio
- - (guión) - Archivo regular
- l - Link simbólico

Más allá del primer ítem, verá alguno de los siguientes permisos:

- r - El archivo se puede leer
- w - El archivo se puede escribir
- x - El archivo se puede ejecutar

1.12.1. El comando `chmod`

Para cambiar los permisos de un archivo debe utilizar el comando `chmod`. Usarlo con precaución, recuerde que Linux mantiene la seguridad isolando los permisos de los archivos. Usted debe garantizarle los permisos adecuados a los usuarios adecuados, de otra manera usuarios indeseados podrían borrar, editar o leer archivos.

Identidades

- u - usuario que creó el archivo
- g - grupo de usuarios al que pertenece
- o - otros
- a - todos

Permisos

- r - lectura
- w - escritura
- x - ejecución

Acciones

- + - agrega permiso
- - - remueve el permiso
- = - hace del permiso que sea único

Ejemplo `chmod u-rw aforismos.txt`

Opciones comunes

- g+w - da permiso de escritura al grupo
- u+x - permite al usuario ejecutar el archivo
- a+rw - todos pueden leer y escribir en el archivo
- ug+r - usuario y grupo pueden leer el archivo
- g=rx - solo el grupo puede leer y ejecutar el archivo, nadie lo puede leer

Si agrega la opción `-R` hará al comando recursivo y los cambios se aplicarán a todos los archivos en los subdirectorios.

Puede utilizar números para cambiar los permisos también.

- r=4
- w=2
- x=1
- -=0

Por ejemplo si quisiéramos darle permisos de lectura y escritura al usuario y grupo solamente, deberíamos usar el número 6 ($r+w = 2+4 = 6$) y el comando sería `chmod 660 archivo`.

1.12.2. El comando `chown`

Con el comando `chown` cambiamos la pertenencia del archivo. Debe ser root para utilizarlo y la sintaxis es simple:

```
chown user:group <archivo>
```

Con la opción `-R` lo hace recursivo.

1.13. Crear archivos

Con el comando `touch <archivo>` creará archivos de cero bytes de información, es decir archivos vacíos.

```
-rw-rw-r-- 1 alumno alumno 0 2010-04-25 23:59 test
```

1.14. Copiando archivos

Para copiar un archivo se usa el comando: `cp < origen > < destino >`. Reemplace “origen” por el archivo que quiere copiar y “destino” por el directorio donde lo quiere colocar.

Lea el manual del comando `cp` para ver el listado de opciones disponibles. Entre ellas están:

- `-i` - interactive. Pide confirmación en caso de sobre escribir un archivo en el directorio destino. Esta opción es útil por que puede prevenir errores.
- `-r` - recursive. Para copiar todo un árbol de directorios.
- `-v` - verbose. Muestra progreso mientras los archivos son copiados.

1.15. Mover archivos

Para mover archivos se usa el comando `mv`. para más referencias diríjase a su manual (escriba `man mv`). Algunas opciones comúnmente usadas son:

- `-i` - interactive. pregunta antes de pisar otro archivo.
- `-f` - force. Ignora al modo interactivo. Sea cuidadoso con esta opción.
- `-v` - verbose. Muestra el progreso mientras el archivo es movido.

1.16. Eliminar archivos y directorios

Se utiliza el comando `rm`. Consulte su manual por favor. Algunas de las opciones más usadas son:

- `-i` - interactive. Pide confirmación antes de eliminar un archivo.
- `-f` - force. Obvia el modo interactivo. No es buena idea usarlo.
- `-v` - verbose. muestra progreso.
- `-r` - recursive. Borra todos los archivos y subdirectorios.

Una vez que un archivo se ha borrado con el comando `rm`, es borrado de forma permanente y no es posible recuperarlo. Tenga especial cuidado con `rm -rf`, el mismo Linus Torvalds una vez borró completamente el árbol de directorios de su sistema por error.

Para eliminar un directorio vacío use el comando `rmdir`.