

LINUX

Primer contacto con Linux

Objetivos

- Podrán entrar y salir del sistema
- Tendrán conocimiento para utilizar los comandos básicos
- Diferenciarán los tipos de archivos
- Inicializar el sistema en diferentes runlevels

Como introducir comandos

La forma de introducir comandos en Linux es similar a la del DOS y los sistemas operativos orientados a línea de comando.

Al igual que UNIX, Linux distingue entre mayúsculas y minúsculas.

Debido a esto si al introducir un comando este no es reconocido, deberemos asegurarnos de haberlo escrito en forma correcta, con las mayúsculas y minúsculas apropiadas. La ejecución del comando se realiza luego de que se presiona la tecla <Enter>.

Como entrar y salir del sistema

Una sesión es el período de tiempo que transcurre desde que nos identificamos en el sistema hasta que salimos de él.

Cabe distinguir dos operaciones básicas:

-El inicio y la terminación de sesiones

-El correcto apagado del equipo.

Para iniciar sesión en el sistema debemos loguearnos con usuario y password. Una vez que seamos autenticados se nos presentará el prompt con dos posibles valores

Para el caso de superuser

\$ Para el caso de usuario normal

Cuando terminamos de trabajar con el ordenador podemos apagarlo de forma que no podemos trabajar con él sin encenderlo de nuevo, o bien podemos cerrar la sesión en curso. La última opción nos permite volver a loguearnos en cualquier momento.

Para salir de una sesión sólo basta con escribir exit, logout o mantener apretadas las teclas Ctrl-D.

Para detener el sistema utilizaremos el comando shutdown seguido de -h hora, donde hora es la hora en que se apagará el sistema. Si ingresamos como hora el número 0 o escribimos now el sistema comenzará a detenerse en el instante.

Otros métodos para apagar el sistema inmediatamente son utilizando el comando poweroff o halt.

Para reiniciar el sistema utilizaremos el comando shutdown -r. También podemos reiniciar el sistema en el instante usando el comando reboot o apretando las teclas Ctrl-Alt-Supr.

Como agregar usuarios

Administración de usuarios

Cada usuario recibe una cuenta de usuario, un password (el cual puede cambiar), un directorio personal y el acceso a ciertas partes del sistema. Esta administración se lleva a cabo de forma centralizada por un usuario especial llamado «root» y solo este puede crear o borrar a un usuario.

Cada usuario, tiene su propio login y su propia password para acceder al sistema. Esto permite prevenir que otro usuario utilice un login que no le corresponde.

El archivo donde están almacenados los passwords de los usuarios es **/etc/passwd**.

Creación de usuarios

Para dar de alta a un usuario, utilizaremos el comando «useradd».

Elegimos un nombre, por ejemplo «dmenino» y ejecutamos:

useradd -m dmenino

La opción «-m» obliga al comando a crear un directorio personal del usuario en /home, por otra parte en el directorio personal del usuario son agregados los archivos de configuración que se encuentren en el directorio /etc/skel (este contiene una serie de archivos de configuración del sistema y de algunos programas).

Los sistemas operativos multiusuarios mantienen una administración de usuarios para poder controlar lo que estos puedan hacer al sistema o a otros usuarios.

Cada usuario, tiene su propio login y su propia password para acceder al sistema. Esto permite prevenir que otro usuario utilice un login que no le corresponde.

El comando useradd crea al usuario con un numero de usuario (UID o User ID) mayor a 100 e incluye a este usuario en el grupo 100 (GID o Group ID). También le asigna al usuario un shell, que por default es /bin/bash. Todas estas opciones se encuentran por defecto en el archivo /etc/default/useradd.

Cada vez que se crea un usuario se agrega una línea en la base de datos de usuario.

Esta base de datos de usuarios es el archivo /etc/passwd.

Significado de la línea dentro del archivo /etc/passwd

1. El nombre de usuario.
2. El password. En este campo se guarda el password que asignamos al usuario de forma encriptada. Si en vez de un password aparece una «x» es que el sistema guarda ese password en otra base de datos que es /etc/shadow, la cual tiene una encriptación mayor (usa el sistema de encriptación md5) y un control de expiración de password mayor que el del /etc/passwd.
3. El UID o user ID. Este número identifica al usuario en el sistema. El sistema mantiene un control de usuarios por el UID, no por el nombre de usuario. Por eso es importante controlar que no haya UID repetidos ni que un usuario tenga como UID el numero 0, ya que este UID es el perteneciente a root.
4. El GID o Group ID. Este numero identifica al grupo al que esta incluido en primera

instancia el usuario. Por defecto, en Linux, todos los usuarios comunes están incluidos en el grupo 100 o users. El GID 0 es el grupo perteneciente a root.

5. Comentario. Este campo es simplemente un comentario, en donde normalmente se pone el nombre completo del usuario, o algo referido al usuario.

6. El directorio personal. Indica cual es la carpeta personal del usuario.

7. El shell. Cuando un usuario ingresa al sistema, ejecuta un shell para poder usar el sistema, este shell esta definido en este campo y por defecto es /bin/bash. En el caso de que el sistema utilice password shadow en él /etc/passwd observaremos el campo del password remplazado por una x como se ve a continuación:

Esto nos indica que el password encriptado se encuentra en el archivo /etc/shadow como se ve a continuación:

Los campos en esta base de datos se describen a continuación:

1. Nombre del usuario
2. Password encriptado con el sistema md5.
3. Este número es la fecha en que fue cambiado por ultima vez el password.
4. Días que tiene el usuario para cambiar el password desde que fue cambiado.
5. Días que tiene antes de cambiar el password.
6. Días que el usuario tiene para cambiar el password, luego de que fue avisado de que tiene que asignar un nuevo password.
7. Días que tiene antes de que el password vencido pase a ser una cuenta deshabilitada.
8. Día exacto que hace que esa cuenta sea deshabilitada.
9. Este campo es reservado para el sistema.

Asignando atributos a un usuario

Uno de los atributos más importantes es el password. Para cambiar este último, existe el comando passwd. Este comando se encarga de cambiar la clave de un usuario, inclusive la clave del root.

Cuando utilice el comando passwd debiera que tener en claro que al ejecutarlo no hay forma de volver atrás. Solamente el superuser puede cambiar passwords de otros usuarios.

Para ingresar un password a dmenino debe ejecutar el comando «passwd». Seguidamente ingresará la password, por ejemplo xGp1234, presione <Enter> y luego reingrese la clave:

Eliminando a un usuario

La forma de eliminar un usuario del sistema es con el comando «userdel». Este elimina al usuario del sistema pero no borra el home del usuario. Para borrar el usuario y el directorio de usuario se le debe

indicar al comando `userdel` que borre el usuario de forma recursiva con la opción «-r».

Si se desea deshabilitar temporalmente a un usuario, basta con poner un asterisco «*» en el campo de password de la base de datos de usuarios o de password, según corresponda.

Grupos de Usuarios

Cada usuario pertenece como mínimo a un grupo. El uso de grupos tiene como fin mantener un sistema de permisos sobre los archivos y poder definir de qué forma pueden los usuarios de ese grupo acceder al archivo. Un grupo puede tener permisos de lectura, escritura y ejecución. El sistema crea algunos grupos, como son el grupo root, bin, sys, mail, etc. Los usuarios no deben pertenecer a ninguno de estos grupos, sino al grupo users.

Creación de grupos

La forma más simple de crear un grupo es agregando una línea en la base de datos de los grupos, que es `/etc/group`. Esta línea cuenta con los siguientes campos separados por dos puntos:

1. Nombre del grupo. Es lo que identifica al grupo.
2. Password del grupo. Este campo quedó en desuso y tiene el mismo formato que el de `/etc/passwd`, el cual si tiene una x es que se guarda en `/etc/gshadow`.
3. El group ID o GID. Sirve para que el sistema pueda identificar el grupo.
4. En este campo se agregan los usuarios que van a estar incluidos en este grupo, separados por una coma « , ».

Un ejemplo de una entrada en el `/etc/group`, sería:

```
Marketing:x:103:lcarlos,mperez,mperalta
```

Es decir: se creó el grupo Marketing que tiene el password almacenado en `/etc/gshadow`, que tiene un GID 103 y que están incluidos en este grupo lcarlos, mperez y mperalta.

La entrada del `/etc/gshadow`, sería:

```
Marketing*:root:root
```

Donde Marketing es el nombre del grupo, el «*» significa que no tiene password y que el único que puede cambiar ese password es root.

Otra forma de crear un grupo es con el comando «`groupadd`».

Eliminar un grupo

Para eliminar un grupo solo basta con borrar la línea correspondiente en `/etc/group` y `/etc/gshadow` o usar el comando `groupdel`.

Cambio de contraseñas

Uno de los atributos más importantes es el password. Para cambiar este último, existe el comando `passwd`. Este comando se encarga de cambiar la clave de un usuario, inclusive la clave del root.

Para ingresar un password a dmenino debe ejecutar el comando «`passwd`». Seguidamente ingresará la password, por ejemplo `xGp1234`, presione <Enter> y luego reingrese la clave:

Cuando se utilice este comando tendrá que tener en claro

que al ejecutarlo no hay forma de volver atrás. Solamente el root puede cambiar passwords de otros usuarios.

Comandos básicos

Comandos para el manejo de archivos:

cd = cambia el directorio
compress = comprime archivos en formato .Z
cp = copia un archivo
chmod = cambia los permisos de un archivo o directorio
chown = cambia el propietario del archivo o directorio
df = muestra el espacio libre en disco
du = muestra el espacio en disco utilizado
fdformat = formatea un diskette
fdisk = para particionar unidades
file = determina el tipo de archivo a través del análisis parcial de su contenido
find = busca un archivo
fsck = chequea el sistema de archivos
gzip = descomprime un archivo en formato GZip
ln = crea un enlace simbólico o físico depende la opción
ls = sirve para listar el contenido de un directorio
mkdir = crea un directorio
mkfs = crea un nuevo sistema de archivos
mkswap = crea un espacio de intercambio
more = exhibe el contenido de un archivo
mount = monta una unidad o partición en el sistema de archivos
mv = mueve un archivo y se utiliza para renombrar
pwd = devuelve la cadena correspondiente al directorio actual
rm = borra un archivo
rmdir = borra un directorio
swapon = activa el espacio de intercambio
swapoff = desactiva el espacio de intercambio
tar = empaqueta o desempaqueta un archivo en formato .tar
type = muestra la ubicación de un archivo señalando su "path"
umount = desmonta una unidad montada

Comandos para el manejo de procesos:

free = muestra la memoria libre y utilizada
halt = apaga la maquina
kill = mata un proceso señalando el numero de este
ldd = muestra las librerías necesarias para ejecutar un proceso
ps = muestra todos los procesos q se están ejecutando en el sistema
pstree = lo mismo q ps pero en forma de árbol
reboot = reinicia el sistema
shutdown = cierra el sistema
top = monitorea procesos y el estado del sistema
uname = muestra información del sistema

Comandos para el manejo de usuarios:

adduser = crea una cuenta de usuario
chsh = cambia la shell de un usuario

groups = muestra el listado de grupos de usuarios del sistema
id = muestra la información de usuario y grupo de un determinado usuario
logout = para salir del sistema y permitir el ingreso a otro usuario
passwd = cambia el password de un determinado usuario
su = le da privilegios de root a un usuario
talk = permite chatear con otros usuarios
users = lista los usuarios conectados al sistema
who = muestra información de los usuarios conectados al sistema
whoami = muestra información nuestra

Otros comandos:

cal = muestra el calendario
date = muestra el día y la hora
info = muestra la ayuda de un comando
man = muestra las paginas del manual de un comando
startx = para iniciar XWindow

Comandos de red

ping = el tan famoso ping que manda paquetes esperando una respuesta
ifconfig = para ver las placas de red
netstat = testeo de red
host "destino" = muestra ip de "destino"
nmap "ip de destino" = pequeño programa para analizar ip's o rangos de ellas

Comandos para el manejo de paquetes

rpm - "opción" "paquete" = instala o desinstala el "paquete" depende la opción
dpkg - "opción" "paquete" = instala o desinstala el "paquete" depende la opción (solo debian)
tar - "opción" "paquete" = comprime o descomprime el "paquete" de formato tar.gz .tgz o tar.bz2

Comandos para el manejo de colas de impresión

lpr archivo = Añade un documento a la cola de impresión
lpr -#3 archivo = Realiza 3 copias del fichero.
lpq = Muestra los documentos en cola de impresión
lprm = Cancela la impresión del documento activo.
lprm 3 = Cancela la impresión del trabajo nº 3.

Comandos Debian

apt-get update = actualiza la base de datos de los paquetes .deb
apt-get upgrade = actualiza los paquetes a su ultima versión
apt-get install "paquete" = instala el paquete deseado
apt-get remove "paquete" = desinstala el paquete deseado
apt-get check = actualiza la cache de paquetes
apt-get clean = borra los paquetes .deb descargados
apt-get dist-upgrade = hace un upgrade del SO
apt-get source "paquete" = descarga fuentes del "paquete"
apt-cache showpkg "paquete" = muestra todas las versiones disponibles de "paquete"
modconf = pequeño programa para sacar o poner módulos del kernel
update-rc.d "opción" "programa o script" "opción" = Remueve o Agrega el script o programa a los niveles de corrida que se le asigne

aptitude = es una versión mejorada de apt

Comandos Red Hat

rpm -q "programa" = para saber si "programa" esta instalado
rpm -qs "programa" = estado de todos los archivos de "programa"
rpm -qd "programa" = documentación de "programa" instalada
rpm -qc "programa" = archivos de configuración de "programa"
rpm -qa "programa" = muestra todos los rpm de "programa"
rpm -qa | grep "programa" = busca el nombre de paquete del "programa"
rpm -i "programa" = instala "programa"
rpm -u "programa" = actualiza "programa"
rpm -e "programa" = elimina "programa"
rpm -ivh "programa" = instala el "programa" en pasos y muestra el progreso de la instalacion

Diferentes tipos de archivos

Comprensión de los nombre de archivos

En Linux, al igual que en otros sistemas operativos como el DOS, se debe distinguir entre un nombre de archivo y un nombre de ruta de acceso. Un nombre de archivo es una serie simple de letras contiguas, números y algunos signos de puntuación. Los nombres de archivos no pueden incluir espacios ni caracteres especiales. Por ejemplo, el nombre de archivos "carlos.dir" es valido, pero "carlos dir" no lo es. Un nombre de ruta de acceso puede tener cualquier número de caracteres. En Linux, los archivos no existen en el vacío, sino en un directorio. El directorio ubicado en la parte más alta se llama directorio raíz y su símbolo es una diagonal «/». Si en el directorio raíz hay un archivo llamado agenda, su nombre de ruta de acceso absoluto es /agenda.

Cuando un usuario se agrega al sistema con el comando adduser, se le asigna un directorio base. Por convención, este directorio base se encuentra bajo el directorio home. Si a un usuario llamado mfernandez se le asigna el directorio /home/mfernandez, todos los archivos que cree mfernandez se vinculan con el directorio /home/mfernandez. Un nombre de ruta de acceso absoluto para uno de los archivos de mfernandez podría ser /home/mfernandez/agenda. Un nombre de ruta de acceso absoluto especifica con exactitud en que parte del sistema de archivos puede encontrar un archivo.

Hay otro tipo de nombre de ruta: el nombre de ruta de acceso relativo, que señala sin ambigüedades un archivo relativo al directorio actual. Si mfernandez esta en su directorio base, el nombre de archivo "agenda" también es el nombre del archivo relativo, esto es, relativo al directorio actual. Para saber cual es el directorio actual, debe usar el comando pwd (print working directory). También puede revisar el contenido de la variable de ambiente \$PWD para ver cual es el directorio de trabajo actual; para ello se debe ejecutar el comando echo \$PWD.

Un nombre de archivo no debe incluir caracteres que tengan un significado especial para el shell. Dichos caracteres son:

| @ # \$ % & / \ () * [] { } ' " : ; < >

Tipos de Archivos

En Linux existen cuatro tipos básicos de archivos: los archivos ordinarios, los directorios, las vinculaciones o links y los archivos especiales.

Archivos Ordinarios

Los archivos ordinarios son los que más se manejan y pueden contener texto, códigos fuente en lenguaje C, scripts de shell, programas binarios ejecutables.

Archivos de directorios

Los directorios son archivos que contienen los nombres de archivos y de subdirectorios, así como los indicadores de estos archivos y subdirectorios. Al listar el contenido de un directorio con el comando ls, en realidad se lista el contenido del archivo de directorios.

Links o vinculaciones

Las vinculaciones o link en realidad no son archivos; son simples entradas de directorio que apuntan a un archivo o directorio

Es posible saber si un archivo está vinculado a otro mediante el comando ls -l, ya que en la respuesta se presenta el nombre del archivo local y enseguida una indicación del archivo vinculado, como se muestra a continuación:

ls -l Info

lrwxrwxrwx 1 root root 6 Sep 17 22:50 Info -> /usr/Info

Archivos especiales

Cada dispositivo físico relacionado con el sistema Linux, incluidos discos rígidos, terminales, impresoras, etc., está representado en el sistema de archivos.

La mayoría de los dispositivos se localiza en el directorio /dev. Por ejemplo, si trabaja en una terminal estándar, su nombre de dispositivo puede ser /dev/tty01.

Directorios en Linux

Directorio Contenido

/bin Comandos ejecutables, compilados en forma dinámica.

/boot Núcleo, información de booteo.

/dev Dispositivos.

/etc Casi todos los archivos de configuración.

/etc/opt/ Archivos de configuración utilizados por programas alojados dentro de /opt/

/etc/X11/ Archivos de configuración para el sistema X Window

/etc/sgml/ Archivos de configuración para SGML

/etc/xml/ Archivos de configuración para XML

/home Directorios de cada usuario.

/lib Mayoría de las bibliotecas de funciones.

/mnt/ Sistemas de archivos montados temporalmente.

/media/ Puntos de montaje para dispositivos de medios como CD

/opt/ Paquetes de aplicaciones estáticas.

/proc Archivos especiales con información del estado del equipo.

/root Directorio del superusuario.

/sbin Binarios compilados.

/tmp Archivos temporarios.

/srv/ Datos específicos de sitio servidos por el sistema.

/usr Comandos, documentación, archivos del entorno gráfico, fuentes del núcleo, etc.

/usr/bin/ Programas binarios.
/usr/include/ Ficheros de inclusión estándar (cabeceras de cabecera utilizados para desarrollo).
/usr/lib/ Bibliotecas compartidas.
/usr/share/ Datos compartidos independientes de la arquitectura del sistema. Imágenes, ficheros de texto, etc.
/usr/src/ Códigos fuente
/usr/X11R6/ Sistema X Window, versión 11, lanzamiento 6
/usr/local/ Datos compartidos de solo lectura específicos del ordenador que los comparte.
/var Archivos de logs del equipo, spooler de impresoras, mail, etc.
/var/cache/ Cache de datos de aplicaciones.
/var/crash/ Depósito de información referente a caídas del sistema
/var/games/ Datos variables de aplicaciones para juegos
/var/lib/ Información de estado variable. Algunos servidores como MySQL y PostgreSQL almacenan sus bases de datos en directorios subordinados de éste.
/var/lock/ Archivos de bloqueo.
/var/log/ Archivos y directorios de registro del sistemas (logs).
/var/mail/ Buzones de correo de usuarios (Opcional)
/var/opt/ Datos variables de /opt/.
/var/spool/ Colas de datos de aplicaciones.
/var/tmp/ Ficheros temporales preservados entre reinicios.

Inicialización y runlevels

Cuando se enciende la máquina, Linux empieza a inicializarse. Lo primero que ejecutara es el propio núcleo del sistema o kernel (en inglés). El núcleo verifica todo el hardware que se encuentra en la máquina y cargara los drivers de cada dispositivo. Una vez que el núcleo terminó de verificar todo, le pasa el control al comando «init», este es el primer proceso de Linux; «padre» de todos los procesos que se ejecutaran en el sistema operativo.

Proceso «init» y run-levels

Linux, como casi todos los Unix, posee 7 diferentes modos de ejecución. Un modo de ejecución, o run-level, definirá que tipo de procesos se ejecutaran en el sistema operativo.

Dependiendo los procesos que se ejecuten, Linux, trabajará en mono-usuario, en red, en modo gráfico, etc. o simplemente se apagará.

Entonces, cada modo de ejecución tendrá un listado de todos los procesos que deba correr. El siguiente, es el detalle de cada modo de ejecución:

Run-Level Descripción

- 0 Apagado del sistema
- 1 Mono usuario
- 2 Multi usuario (s/NFS)
- 3 Multi usuario completo
- 4 No se utiliza
- 5 Modo gráfico
- 6 Reinicio de sistema

Como se puede observar, cada run-level generará que Linux se comporte de una forma o de otra.

Es posible cambiar de run-level aunque Linux se haya inicializado en un run-level diferente.

Por ejemplo, necesitamos hacer tareas de mantenimiento en el equipo y decidimos inicializar Linux en modo mono usuario (run-level: 1). Una vez que terminamos con las tareas de mantenimiento podemos dejar que todos los usuarios trabajen normalmente, por lo tanto, en vez de reinicializar el equipo nuevamente, simplemente, utilizamos el comando «telinit» para entrar en modo multi usuario completo (run-level: 3). También, podrá volver atrás (suponiendo que olvido realizar algo) a modo mono usuario (run-level: 1) utilizando el mismo comando (telinit).

Como se arranca Linux en otro run-level que no sea el default

Para realizar esto tendrá varias alternativas, la más común es hacerlo a través del LILO o GRUB (administradores de booteo de Linux). Cuando vemos el prompt del lilo «LILO: », en vez de presionar <Enter>, deberá escribir «linux <nro-run-level>». Por ejemplo, para inicializar Linux en modo mono usuario, escribirá «linux 1» en el prompt del lilo.

En GRUB al momento de iniciar la maquina presionaremos la tecla «e» para editar. Seleccionaremos la linea que indica el kernel (en nuestro ejemplo la segunda linea) y presionaremos nuevamente la tecla «e». Al final de la misma agregamos el numero de runlevel en el que queremos iniciar el sistema presionamos «enter» y luego la tecla «b» para bootear el sistema.

```
title Debian, kernel 2.6.17-11-386
root (hd0,0)
kernel /boot/vmlinuz-2.6.17-11-386 root=/dev/sda1 ro quiet splash 3
initrd /boot/initrd.img-2.6.17-11-386
savedefault
boot
```

En el caso de que no tengamos instalado el LILO, podemos hacer este cambio en el archivo «/etc/inittab». Existe una línea en este archivo que indica el run-level que se utiliza por defecto. Cuando se instala Linux, el run-level por defecto es multi usuario completo (run-level: 3).

Editando el archivo «/etc/inittab» encontraremos, entre muchas otras cosas, esta línea:

id:3:initdefault:

Para cambiar el run-level por defecto, simplemente, cambiaremos el «3» por otro número,

por ejemplo modo gráfico (run-level: 5) poniendole «5». Entonces, la línea del archivo «/etc/inittab» quedará de esta forma:

id:5:initdefault:

Como cambiamos de un run-level a otro

Para cambiar de un run-level a otro, mientras esta corriendo Linux, utilize el comando «telinit». Este comando recibe como argumento el número de run-level deseado. Por ejemplo, si quisiera pasar de modo multi usuario completo (run-level: 3) a modo gráfico (run-level: 5), ejecutaremos «telinit 5».

Terminales y consolas

¿Qué son las consolas virtuales?

Las consolas virtuales aparecieron por primera vez en minix y en FreeBSD, proveen acceso a multiple login sessions, lo que significa que varios usuarios pueden loguearse en el sistema y cada uno de ellos pueden ejecutar diversas tareas a la vez.

Con Alt + alguna de las teclas F1, F2, F3, etc. nos cambiamos a otra virtual terminal. Normalmente hay un máximo de 6 terminales virtuales disponibles. Linux permite trabajar al mismo tiempo en varias consolas de texto.

Puede configurar el número de terminales editando el archivo /etc/inittab.

Si está usando X-Windows, la parte gráfica, con la combinación de teclas: Ctrl + Alt + Fn vuelve a la consola de texto. Para regresar a X-Windows use Alt+F7, accediendo así a la consola empleada (la primera libre, es decir no manejada por un getty). Por ejemplo, si tiene 6 consolas virtuales (de Alt+F1 hasta Alt+F6), con Alt+F7, volveremos al modo gráfico.

Para pasar de una a otra, pulse Alt + Fn donde n es el número de la ventana virtual y Fn una tecla de función (por ejemplo F1, F2, etc.).

Bibliografía de referencia

Libros:

Ubuntu® Linux® Toolbox: 1000+ Commands for Ubuntu and Debian Power Users

Christopher Negus y François Caen
Wiley Publishing, Inc.

Debian GNU/Linux 3.1 Bible

David B. Harris y Jaldhar Vyas

Wiley Publishing, Inc.

Learning Debian GNU/Linux

Bill McCarty

CUESTIONARIO

Leer y completar los siguientes puntos.

- 1 Como introducir comandos.
- 2 Como entrar y salir del sistema.
- 3 Como agregar usuarios.
- 4 Cambio de contraseñas.
- 5 Comandos básicos.
- 6 Diferentes tipos de archivos.
- 7 Directorios en Linux.
- 8 Inicialización y runlevels.
- 9 Terminales y consolas.