TEMA 10

INDICE

Administración básica del sistema (Linux II).

1.- Administración de usuarios.

GNU/Linux es un sistema operativo multiusuario. Esto significa que permite a varios usuarios utilizar el sistema simultáneamente, a través de la línea de comandos o con conexiones remotas. GNU/Linux controla el acceso al equipo y a sus recursos a través de las cuentas de usuarios y grupos.

En los sistemas GNU/Linux existen tres tipos de usuarios:

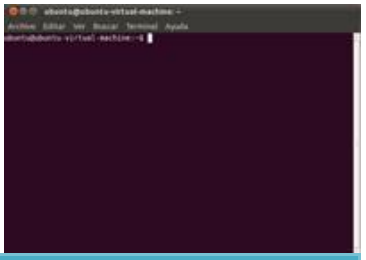
- ✓ **Root.** Es el usuario más importante ya que es el administrador y dueño del sistema. Se aconseja utilizar la cuenta de *root* para las tareas específicas de administración y el resto del tiempo utilizar una cuenta de usuario normal.
- ✓ **Usuarios normales.** Son los usuarios que pueden iniciar sesión en el sistema y tienen una funcionalidad limitada, tanto en los comandos que pueden ejecutar, como a los ficheros a los que tienen acceso.
- ✓ **Usuarios asociados a servicios.** Este tipo de usuarios no pueden iniciar sesión en el sistema. Su utilización es muy útil ya que permiten establecer los privilegios que tiene un determinado servicio. Por ejemplo, el servidor de páginas Web tiene asociado un usuario para poder especificar a qué ficheros tiene acceso; y por lo tanto que ficheros son visibles a través de Internet.

Todos los usuarios del sistema tienen un identificador de usuario (UID) y un identificador de grupo (GID). El administrador del sistema *root* tiene los identificadores de usuario y grupo 0:0 y los demás usuarios tienen un valor mayor que 0.

Existen varias formas de administrar el sistema, que van variando dependiendo de su facilidad o control sobre el sistema. Básicamente, puede administrar el sistema a través de tres formas diferentes:

- ✓ **Interfaces gráficas.** Existen diferentes interfaces gráficas que permiten administrar el sistema de una forma fácil y sencilla. Puede utilizar la interfaz de administración de x-Windows o utilizar la web de administración (webmin). Este método es el más sencillo, pero es el que menos control proporciona sobre el sistema.
- ✓ **Terminal del sistema.** Una de las ventajas de los sistemas GNU/Linux es que puedes administrarlo totalmente a través del intérprete de comandos o terminal del sistema. Una de las grandes ventajas de utilizar el terminal del sistema es que permite una gran flexibilidad a la hora de interactuar con el sistema, pudiendo crear pequeños programas (scripts) para simplificar la administración del sistema.
- ✓ **Ficheros de configuración.** Por último, la modificación directa de los ficheros de configuración es el método que permite tener un mayor control del sistema. Como desventaja hay que destacar que para administrar el sistema de esta forma hay que conocer muy bien el sistema.

No se puede decir que un método sea el mejor, ya que el uso de un método u otro depende siempre de la tarea que desees realizar y de tus conocimientos. Lo mejor, como siempre, es conocer los tres métodos y utilizar el mejor en cada momento.



1.1.- Intérprete de comandos.

La gestión de usuarios y grupos se puede realizar directamente a través del intérprete de comandos. En la siguiente tabla se muestran los comandos más importantes para la gestión de usuarios y grupos.

Comandos más utilizados (usuarios)	
Comando	**Descripción**
	Usuarios
adduser <usuario>	Permite dar de alta a un usuario. Cuando das de alta un usuario el sistema solicita sus datos como nombre completo, dirección, contraseña, etcétera.
addgroup	Permite dar de alta un usuario dentro de un grupo.
chage	Permite establecer los periodos de vigencia de las contraseñas.
id	Muestra el usuario que se está utilizando.
passwd	Permite cambiar la contraseña de un usuario. Si ejecutas passwd cambias la contraseña del usuario actual y si ejecutas passwd nombre_usuario cambia la contraseña del usuario indicado.
su	Permite cambiar de usuario.
sudo	Permite ejecutar un comando como root.
userdel <usuario>	Permite borrar un usuario.
usermod	Permite modificar las propiedades de un usuario.
	Grupos
groups	Muestra los grupos a los que pertenece el usuario.
groupadd	Permite dar de alta a un grupo.
groupdel	Permite borrar un grupo de usuarios.
groupmod	Permite modificar las propiedades de un grupo.
	Manipulación del fichero /etc/shadow
pwconv	Crea y actualiza el fichero /etc/shadow.
pwunconv	Desactiva el fichero /etc/shadow.

1.2.- Ficheros utilizados.

Siempre resulta muy útil conocer el funcionamiento interno del sistema operativo para poder tener un mayor control de las operaciones que realiza. Para conocer el funcionamiento interno debes conocer dos tipos de ficheros: aquellos ficheros que se utilizan para guardar la información de los usuarios y grupos, y los ficheros con los valores predeterminados que utiliza el sistema.

La información de las cuentas de usuario y grupos se encuentran en los siguientes ficheros:

- ✓ `/etc/passwd`. En este fichero se encuentra un listado de las cuentas de usuario que están dados de alta en el sistema.
- ✓ `/etc/shadow`. En este fichero se encuentran cifradas las contraseñas y sus periodos de vigencia.
- ✓ `/etc/group`. Listado de grupos activos en el sistema y usuarios que pertenecen a dichos grupos.

En el fichero `/etc/passwd` se almacenan los datos de las cuentas de los usuarios. A continuación se muestra el fragmento de código de un usuario:

```
javier:x:1000:1000::/home/javier:/bin/bash
```

Como puede ver en el ejemplo anterior, para cada usuario se almacena la siguiente información:

`Login`:`x`:`UID`:`GID`:`Descripción`:`Directorio de trabajo`:`Shell del usuario`

Es recomendable asignar a los servicios del sistema el shell /bin/false para que no puedan iniciar sesión en el sistema.

Por motivos de seguridad, las contraseñas de los usuarios se almacenan en el fichero `/etc/shadow` y no en el fichero `/etc/passwd`. Por ejemplo, para el usuario anterior en el fichero `/etc/passwd` en vez de almacenar la contraseña se guarda el carácter "`x`" y en el fichero `/etc/shadow` se almacena la contraseña cifrada.

El fichero `/etc/group` almacena los datos de los grupos que han sido dados de alta en el sistema. A continuación se muestra un fragmento del fichero:

```
root:x:0:root,javier
javier:x:1000:
```

Para cada grupo el sistema almacena el nombre del grupo, el identificador de grupo (GID) y los usuarios que pertenecen al grupo. En el ejemplo anterior se puede ver como los usuarios `root` y `javier` pertenecen al grupo `root`.

Al dar de alta un usuario si no especifica ningún parámetro el sistema utiliza los valores por defecto. El sistema guarda los valores por defecto en los siguientes ficheros:

- ✓ **`/etc/default/useradd`**. Permite establecer el shell que se va utilizar por defecto, el directorio home que van a tener los usuarios, etcétera.
- ✓ **`/etc/login.defs`**. Entre las opciones más importantes permite establecer los datos de expiración de las contraseñas, longitud mínima de las contraseñas, UID y GID mínimos y máximos, etcétera.

1.3.- Configuración con asistentes.

La administración de los usuarios del sistema se puede realizar gráficamente con la herramienta `Usuarios y grupos` en xWindows o a través de `webmin`.

Inicia la aplicación `Usuarios y grupos` que se encuentra en el submenú `Administración` dentro de `sistema`. Aparece la ventana `Gestor de usuarios` donde puedes realizar la administración de los usuarios del sistema de una forma fácil y sencilla.

Para añadir un nuevo usuario pulsa el botón `Añadir`, introduce el nombre de usuario, pulsa `Aceptar` y posteriormente introduce la contraseña del usuario.

Otra forma de administrar los usuarios del sistema es utilizar `Webmin`. Recuerda que Webmin es una herramienta de configuración de sistemas accesible vía web para GNU/Linux y otros sistemas Unix. Para ello puedes acceder con un navegador a webmin (`https://127.0.0.1:10000`). Una vez dentro en la página principal dentro de menú `System` accedes a `Users and groups.`

2.- Sistema de ficheros.

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.

Existen diferentes formas que permiten administrar el sistema de ficheros y cada una de ellas proporciona diferentes resultados dependiendo de si desea administrar el sistema utilizando particiones, volúmenes o sistemas RAID.

Para identificar los discos duros o particiones se utiliza la siguiente sintaxis `/dev/sda1` donde:

- ✓ ***s*** indica el tipo de disco duro: s – discos duros SATA o SCSI; y h para discos IDE.
- ✓ ***a*** identifica el primer disco duro, b el segundo, etcétera
- ✓ ***1*** indica el número de partición dentro del disco duro.

Así por ejemplo `/dev/sdb3` identifica la tercera partición del segundo disco duro y `/dev/sdb` identifica el segundo disco duro.

Aunque vamos a hacer un repaso de algunas herramientas para trabajar con el sistema de ficheros, te recomendamos el siguiente enlace para conocer algo más del sistema de ficheros de Ubuntu.
http://www.guia-ubuntu.com/index.php/Sistema_de_ficheros

2.1.- Particionado.

El particionado es uno de los procesos más importantes que hay que tener en cuenta, ya que define cómo se van a utilizar los diferentes discos duros del equipo. En el proceso de particionado hay que prestar un especial cuidado para no perder datos del sistema.

La administración de las particiones de los sistemas de ficheros se puede realizar con herramientas gráficas como la `Utilidad de discos`, el `Administrador de volúmenes lógicos` o, manualmente, con el comando `fdisk`.

En los servidores es recomendable utilizar un sistema RAID por hardware para permitir que, en caso de rotura de un disco duro, no se pierda la información del sistema.

2.1.1.- Herramientas gráficas.

Ubuntu Desktop por defecto instala la herramienta `Utilidad de discos` para administrar el sistema de ficheros. Utilizando la

herramienta `Utilidad de discos` puede crear, modificar o eliminar las particiones de los discos duros del sistema.

Para ejecutar la herramienta debes ir al menú `Sistema > Administración` y seleccionar la herramienta `Utilidad de discos`.

Por otra parte, es posible utilizar el `Administrador de volúmenes lógicos`. A diferencia de la herramienta `Utilidad de discos` , con el `Administrador de volúmenes lógicos` es posible crear volúmenes o unidades RAID. Recuerda que un volumen permite agrupar uno o más discos duros para tener un sistema de ficheros de mayor tamaño. Además, puede crear volúmenes en los que se mejore la seguridad de los datos. Por ejemplo, en un volumen reflejado (o RAID 1) los datos se guardan de forma simultánea en dos discos duros.

Para utilizar el administrador de volúmenes lógicos antes debes instalarlo, ejecutando:

```
# apt-get install system-config-lvm
```

y ejecutar la herramienta `Administración de volúmenes lógicos` que se encuentra en el submenú `Herramientas del sistema` dentro del menú `Aplicaciones`.

2.1.2.- fdisk.

La utilidad fdisk, a pesar de que es un poco incomoda de utilizar porque no trabaja bajo una interfaz gráfica, es muy útil y potente. Para aprender mejor lo que hace, se va a utilizar fdisk para crear una partición en uno de los discos duros que tienes libre en el sistema, se formatea y se monta para poder utilizarla.

Los pasos que hay realizar para utilizar un disco son:

- ✓ Crear la partición.
- ✓ Fomatear la partición.
- ✓ Montar la partición.

A continuación vas a utilizar un nuevo disco duro en el sistema.

Documentación del particionado de una unidad de disco.

Crear la partición

El primer paso que debes realizar es conocer los discos duros y particiones que tiene el sistema. Para ello ejecutas: `# fdisk -l`

Tal y como puede ver en la figura 1, el equipo tiene dos discos duros (`/dev/sda` y `/dev/sdb`). El primer disco duro (`/dev/sda`) tiene dos particiones donde está el sistema operativo (`/dev/sda1`) y la partición swap (`/dev/sda2`). Y el segundo disco duro no contiene ninguna tabla de particiones válida.

Por ejemplo, si quieres utilizar `fdisk` en el segundo disco duro entonces hay que ejecutar:

```
# fdisk /dev/sdb
```

Una vez dentro del disco duro el sistema informa que el disco duro no contiene ninguna tabla de particiones válida. Si deseas conocer los comandos disponibles pulsa `m`.

Para crear una partición en el sistema pulsa *n* y realiza los siguientes pasos:

- ✓ Selecciona el tipo de partición que quieres crear: (`p`) primaria y (`e`) extendida. Pulsa *p.*
- ✓ Indica el número de la partición primaria. Como es la primera pulsa `1`.
- ✓ Ahora hay que indicar el tamaño de la partición.
- ✓ A continuación indica el último cilindro. Para especificar el tamaño de la partición puedes indicar el número del último cilindro o indicar el tamaño en Mega Bytes que quieres asignarle a la partición de la forma *+tamañoM (p.e.: 1000M).* Por ejemplo, pulsa `Enter` para utilizar todo el disco duro.

Una vez creada la partición pulsa `p` para ver la tabla de particiones. Tal y como se muestra en la imagen el disco tiene la partición `/dev/sdb1`.

Una vez realizados todos los cambios hay que guardar la configuración y salir de la aplicación, utilizando `w`.

Formateo

Una vez creada la partición, el siguiente paso es formatearla con el comando `mkfs`. Para formatear la partición ejecuta:

```
# mkfs /dev/sdb1
```

Montar la unidad

Una vez lista la partición /dev/sdb1 para poder utilizarla hay que montarla en un directorio existente.

```
# mkdir /datos
```

Existen dos formas diferentes de montar una partición:

- ✓ **Manualmente con el comando mount.** Esta opción es la más sencilla y permite montar un sistema de ficheros de forma puntual ya que si se reinicia el ordenador se pierde el punto de montaje.
- ✓ **Automáticamente editando el fichero /etc/fstab.** Esta opción permite montar de forma permanente un sistema de ficheros. Es la mejor opción en el caso de querer utilizar siempre el sistema de ficheros, o si quieres realizar en él acciones especiales como por ejemplo, utilizar las cuotas de usuarios.

Para montar manualmente nuestra partición ejecutamos:

```
# mount /dev/sdb1 /datos
```

Si deseas montar de forma definitiva el sistema de ficheros entonces hay que editar el fichero `/etc/fstab` y añadir al final la siguiente línea de configuración.

```
/dev/sdb1      /datos ext2      defaults          0 0
```

Una vez modificado el fichero de configuración, la partición se monta automáticamente al reiniciar el equipo o puedes montarla ahora ejecutando `mount /datos`.

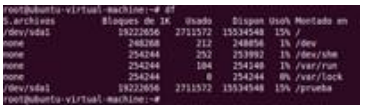
Para finalizar, si quieres ver que la partición está correctamente montada puede ejecutar el comando `mount` o `df`.

Hay que tener mucho cuidado al modificar el fichero /etc/fstab ya que se puede dañar el sistema.

2.2.- Monitorización.

Existen muchas herramientas que permiten monitorizar el sistema de ficheros entre las que destacan:

df. Muestra un resumen sobre el espacio libre que queda en los discos duros del sistema.



du. Muestra la cantidad de espacio que están utilizando los directorios o archivos específicos. Por ejemplo, si quieres ver el espacio que ocupa el directorio `/datos` en Megabytes ejecuta:

```
$ du -ms /datos
```

fsck. Permite comprobar el estado y reparar un sistema de ficheros.

3.- Permisos.

Es muy importante establecer correctamente los permisos en el sistema de ficheros porque así evitaras usos indebidos o pérdidas de datos en el sistema.

Si ejecutas en un directorio el comando `ls -la` puedes ver los permisos del sistema de ficheros. Para cada fichero o directorio se muestran los siguientes datos:

- ✓ **Permisos.** Indica los permisos que tiene el fichero o directorio.
- ✓ **Usuario propietario.**
- ✓ **Grupo propietario.**
- ✓ **Tamañodel fichero o directorio**.
- ✓ **Fecha de creación o de la última modificación.**
- ✓ **Nombre.**

Por ejemplo, los permisos para el directorio `documentos` son `drwxrwx---`. El carácter `d` indica que es un directorio. Luego se muestran tres grupos de caracteres (`rwx`) (`rwx`) (`---`) que permiten indicar los permisos del usuario propietario, del grupo propietario y de los demás usuarios.

El formato para establecer los permisos es (`rwx`) donde `r` indica lectura, `w` escritura y `x` indica ejecución. Si existe el permiso entonces se muestra su correspondiente letra y en el caso de que no exista ese permiso entonces aparece el carácter (`-`).

Por ejemplo, el directorio `documentos` tiene todos los permisos (`rwx`) para el usuario propietario, que es `maria`, el grupo propietario `jefes` también tiene todos los permisos (`rwx`), y el `resto` de los usuarios no tiene ningún permiso (`---`).

El directorio `programas` tiene todos los permisos para el usuario propietario `maria` (`rwx`) y tanto para el grupo propietario `usuarios` como el `resto` de los usuarios tiene permisos de lectura y ejecución (`r-x`).

En un fichero el permiso de ejecución permite ejecutar un programa y en el caso de los directorios el permiso permite indicar que es posible entrar en ese directorio.

3.1.- Establecer los permisos.

Para definir los permisos de un fichero o directorio se emplea el comando chmod. Su sintaxis es:

```
# chmod <modo> fichero
```

donde `<modo>` indica los permisos que le quiere asignar al fichero. Por ejemplo, si quiere establecer los permisos `rw-` para el propietario y `r--` para el resto, el comando que se debe utilizar es:


```
# chmod 644 fichero
```

Con `chmod` se puede establecer los permisos con tres valores numéricos (por ejemplo, `664`): el primer valor corresponde al **usuario propietario**, el segundo al **grupo propietario** y el tercer valor corresponde a **todos los demás usuarios** del sistema.

Cada permiso tiene una equivalencia numérica donde `r` vale **4**, `w` vale **2** y `x` vale **1**. De esta forma si tiene el valor **7** corresponde a (`rwx`), el valor **6** corresponde a (`rw-`), etcétera.
El propietario de un fichero es aquel usuario que creó dicho fichero. GNU/Linux permite cambiar al
propietario de cualquier fichero o directorio. Opcionalmente se puede cambiar también al grupo al que pertenece dicho fichero o directorio. Para ello se utiliza la orden `chown` que tiene la siguiente sintaxis:

```
chown <NombreUsuario> [.<NombreGrupo>] <fichero>...
```

donde `<NombreUsuario>` identifica el nuevo propietario de fichero o directorio. `<NombreGrupo>` el nuevo grupo y `<fichero>` identifica el fichero o directorio sobre el que se va a actuar.

Por otro lado, para cambiar el grupo al que pertenece un directorio se utiliza `chgrp`. Su sintaxis es:

```
# chgrp <NombreGrupo> <fichero>...
```

donde `<NombreGrupo>` identifica el nuevo nombre de grupo que se le va a asignar al fichero o directorio `<fichero>`. Se puede actuar sobre varios ficheros a la vez.

En los comandos `chmod`, `chown` y `chgrp` la opción `-R` significa que se establecen los permisos al directorio y a todos los datos que contiene. Por ejemplo, el comando

```
# chmod 777 /datos -R
```

establece todos los permisos a la carpeta `datos` y a todo su contenido.

4.- Arranque y parada.

Una de las funciones de un administrador de sistemas es poder contestar en todo momento las siguientes preguntas: ¿qué sistema operativo se ejecuta en nuestro sistema? ¿qué servicios o programas se ejecutan en el sistema? ¿cuándo se ejecutan? Lógicamente, estos factores afectan muy estrechamente a la seguridad y al rendimiento del sistema.

En esta unidad se abordan los temas necesarios para poder tener control total sobre el proceso y arranque del sistema. Cuando se inicia el equipo primero inicia la BIOS que permite detectar y acceder al hardware del sistema. A partir de ahí, carga el gestor de arranque (que en Linux se llama GRUB) y en el caso de iniciar un sistema GNU/Linux accede al directorio /boot donde carga el kernel o núcleo del sistema operativo y ejecuta el proceso init que será el encargado de iniciar todos los servicios para que el sistema funcione correctamente.

A continuación, se analizan cada uno de los elementos que intervienen en el arranque y apagado del sistema: gestor de arranque (GRUB), proceso de arranque, servicios del sistema, planificación de tareas y parada del sistema.

4.1.- Gestor de arranque.

El gestor de arranque es el encargado de iniciar cualquier sistema operativo que haya sido previamente instalado en el sistema (por ejemplo, Windows, GNU/Linux, FreeBSD). De forma tradicional el gestor de arranque utilizado en GNU/Linux era LILO, aunque actualmente el gestor de arranque más utilizado en la actualidad es GRUB.

GRUB (Grand Unified Bootloader) fue diseñado por Erich Stefan Boleyn y es un gestor de arranque que permite gestionar el inicio de nuestro equipo entre diferentes sistemas operativos.

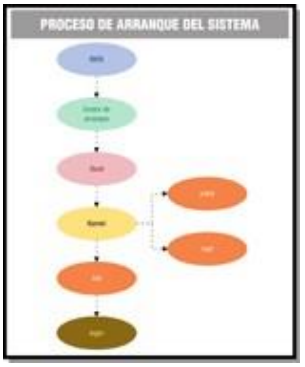
Siempre que realices operaciones sobre el gestor de arranque es muy importante estar seguros de las opciones y parámetros introducidos, ya que es posible dañar el arranque del sistema. Aún así, siempre es posible utilizar alguna utilidad de recuperación del arranque, como por ejemplo **Super GRUB Disk**, de libre distribución. Esta herramienta además permite a usuarios avanzados realizar operaciones potencialmente peligrosas en el *MBR (Master Boot Record o Registro de Arranque Principal)* de forma segura.

En el sitio oficial de GRUB puede encontrar más información sobre el gestor de arranque:
http://www.gnu.org/software/grub/

4.2.- Proceso de arranque y parada del sistema.

Una vez que se ha encontrado el kernel y se ha iniciado. El sistema operativo comienza a cargarse, se inicia el hardware, los discos están preparados, se asignan direcciones IP, se inician servicios, y se realizan otras muchas tareas. Para ello, Linux ejecuta el progra na `init`, cuya función es iniciar el sistema operativo y sus servicios. Las tareas que realiza el proceso `init` son

- ✓ Comprueba los sistemas de ficheros.
- ✓ Monta los sistemas de ficheros permanentes.
- ✓ Activa la zona de memoria swap o de intercambio.
- ✓ Activa los demonios o servicios del sistema (por ejemplo, *atd* y *syslog*).
- ✓ Activa la red.
- ✓ Inicia los demonios o servicios de red del sistema (por ejemplo,*sendmail* y *httpd*).
- ✓ Limpia los sistemas de ficheros temporales.
- ✓ Finalmente, habilita el login a los usuarios del sistema.



El proceso `init` es el estándar para iniciar y apagar equipos Linux y Unix llamado `SysV`. `SysV` es un modo de definir qué estado debe tener el equipo en un momento determinado. Para ello se emplea un concepto denominado modo de ejecución (o *runlevels)*.

`SysV` utiliza siete modos de ejecución que van del 0 al 6, y cada distribución utiliza los modos de ejecución para diferentes fines aunque hay varios niveles que son comunes. Los niveles que son comunes son:

- ✓ 0 se utiliza para apagar el equipo
- ✓ 1 es el modo monousuario
- ✓ 6 se utiliza para reiniciar el equipo. Los demás niveles
- ✓ 2 al 5, en Ubuntu, permiten iniciar el equipo en modo multiusuario.

A continuación se van a ver las tareas más frecuentes sobre el nivel de ejecución del sistema:

- ✓ Si quieres, puedes **cambiar el nivel de ejecución del sistema por defecto** modificando el fichero `/etc/init/rc-sysinit.conf` de la siguiente forma:

```
env DEFAULT_RUNLEVEL=2
```

- ✓ Para ver el nivel de ejecución que tiene actualmente el sistema debes ejecutar:

```
# runlevel
```

- ✓ Para **cambiar manualmente el nivel de ejecución del sistema** hay que ejecutar:

```
# telinit 3
```

o

```
# init 3
```

Cada nivel de ejecución, tiene asociado un directorio donde se especifican los servicios que se deben ejecutar o parar. Por ejemplo, el directorio `/etc/rc0.d` corresponde al **nivel 0**, el directorio `/etc/rc1.d` al **nivel 1**, etcétera.

Ahora bien, ¿cómo puedo ver los scripts que se ejecutan en un determinado nivel? Existen varias formas de ver los servicios asociados a un determinado nivel. Por ejemplo, si muestras el contenido del directorio:

```
cd /etc/rc3.d
ls -l
```

obtienes una salida como la siguiente:

```
lrwxrwxrwx 1 root root 17 3:11 S10network -> ../init.d/network
lrwxrwxrwx 1 root root 16 3:11 S30syslog -> ../init.d/syslog
lrwxrwxrwx 1 root root 14 3:32 S40cron -> ../init.d/cron
lrwxrwxrwx 1 root root 14 3:11 S50inet -> ../init.d/inet
lrwxrwxrwx 1 root root 13 3:11 S60nfs -> ../init.d/nfs
lrwxrwxrwx 1 root root 15 3:11 S70nfsfs -> ../init.d/nfsfs
lrwxrwxrwx 1 root root 18 3:11 S90lpd -> ../init.d/lpd.init
lrwxrwxrwx 1 root root 11 3:11 S99local -> ../rc.local
```

Como se puede observar, el directorio contiene enlaces simbólicos a scripts del directorio `/etc/init.d`. Cada enlace tiene una letra (***S*** o ***K)*** y un número al principio. El número establece el

orden en el que se van a ejecutar los servicios mientras que la letra "***S***" significa que se inicia y la "***K***" que se para el servicio correspondiente.

¿Cómo hace el proceso `init` para arrancar y parar los servicios? Sencillo. Cada uno de los scripts se escribe para aceptar un argumento que suele ser `start`, `stop`, `status`, `restart` o `relaod`. Si lo desea puedes ejecutar los scripts manualmente.

Por ejemplo, si quieres ver las opciones de un determinado servicio puedes ejecutarlo directamente:

```
# /etc/init.d/apache2
```

Uso:

```
./httpd {start|stop|restart|condrestart|reload|status|fullstatus|graceful|help|configtest}
```

Por lo tanto, si quiere parar el servidor de páginas web manualmente ejecuta:

```
# /etc/init.d/apache2 stop
```

También puedes administrar los servicios utilizando el comando *service* de la siguiente forma:

```
# service apache2 stop
```

Una vez realizados todos los pasos que establece el nivel de ejecución, se procesa el fichero `/etc/rc.local`. Este fichero es un "cajón desastre" donde se pueden escribir todos los comandos que el sistema ejecuta al iniciarse.

4.3.- Servicios del sistema.

Los servicios son aplicaciones que se ejecutan, en segundo plano, independientemente del usuario y ofrecen una determinada funcionalidad. Normalmente se asocia el término "servicio" sólo a servicios de red (por ejemplo, servidor web, servidor ftp) pero existen servicios que ofrecen todo tipo de funcionalidades (gestionan las conexiones de red, monitorizan el sistema, comprueban las actualizaciones y seguridad del sistema, permiten utilizar el hardware del equipo, etcétera).

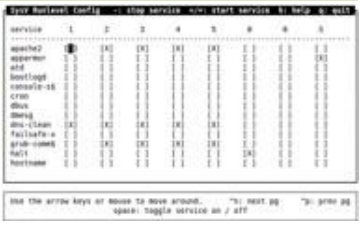
El administrador de servicios permite establecer los servicios que se van a ejecutar al iniciar el sistema, y permite parar, ejecutar o reanudar los servicios que se ejecutan actualmente en el sistema.

Si deseas administrar los servicios del sistema en modo terminal entonces debes instalar la herramienta `sysv-rc-config`:

```
# apt-get install sysv-rc-config
```

Una vez instalada ejecuta en el terminal

```
# sysv-rc-config
```



y en la pantalla que se muestra en la siguiente figura habilita o deshabilita los servicios que estimes oportunos

Por otro lado, `chkconfig` permite administrar los servicios que se van a iniciar automáticamente cuando arranca el sistema. Para utilizar `chkconfig` en Ubuntu antes es necesario instalarlo ejecutando:

```
# apt-get install chkconfig
```

Para ver el estado de los diferentes servicios ejecutamos:

```
# chkconfig --list
```

Donde en cada fila muestra un determinado servicio y en cada columna se indica, si el servicio se inicia automáticamente en ese modo de ejecución (del modo 0 al 6).

Por ejemplo, si quieres que el servidor web se ejecute automáticamente ejecutamos:

```
# chkconfig apache2 on
```

Si deseas activarlo en los niveles 235 ejecutamos:

```
# chkconfig –levels 235 apache2 on
```

Y si desea deshabilitarlo ejecuta:

```
# chkconfig apache2 off
```

4.4.- Procesos.

En los sistemas GNU/Linux se ejecutan una gran cantidad de servicios que permiten realizar una determinada actividad en el sistema. Cada servicio o demonio consiste en uno o más procesos que se ejecutan en el equipo. Además de los procesos vinculados a servicios, en el sistema se encuentran los procesos que ejecuta un usuario. Por ejemplo, un editor de textos, un navegador Web, etcétera.

Para ver los procesos que se ejecutan en el equipo hay que ejecutar el comando ps. Tal y como se muestra en la siguiente figura, para cada proceso se muestra su identificador (PID), terminal donde se ejecuta (TTY), tiempo de uso de CPU (TIME) y el comando que ejecuta (CMD).

Si quires ver todos los procesos que se ejecutan en el sistema utiliza la opción `-A`:

```
# ps -A
```

Si deseas eliminar un proceso que se está ejecutando en el sistema puede utilizar el comando `kill` de la siguiente forma:

```
# kill -9 <ID del proceso>
```

Otra aplicación que permite ver los procesos que se ejecutan en el sistema es `top`. Top es una aplicación que, en tiempo real, informa sobre la actividad del sistema. Proporciona información sobre la carga del sistema operativo, grado de utilización de la CPU, memoria y swap, y los procesos que se encuentran en ejecución.

4.5.- Programación de tareas.

La programación de tareas permite programar la ejecución de un determinado programa en un momento determinado. Por ejemplo, se puede programar una copia de seguridad, enviar un fichero, comprobar la seguridad del sistema, enviar un informe, etcétera.

Antes de programar las tareas hay que comprobar que el servicio crond se encuentra en ejecución mediante el comando:

```
# service crond status
```

Para modificar el fichero de configuración de *crond,* ejecuta el comando:

```
# crontab -e
```

y aparece un fichero con el siguiente formato:

```
PATH=/bin
0 0 * * * /root/comprobar_seguridad.sh
0 0 1 * * /root/copia_seguridad.sh

La sintaxis de las tareas programadas es:
# .---------------- minuto (0 - 59)
# |   .------------- hora (0 - 23)
# |   |   .---------- día del mes (1 - 31)
# |   |   |   .------- mes (1 - 12) o jan,feb,mar,apr ... (los meses en inglés)
# |   |   |   |   |   .---- día de la semana (0 - 6) (Domingo = 0 o 7) OR
sun,mon,tue,wed,thu,fri,sat (los días en inglés)
# |   |   |   |   |
# |   |   |   |   |
  *   *   *   *   *  Comando a ejecutar
```

En el ejemplo anterior se ejecuta el script `comprobar_seguridad.sh` **todos los días a las 0:00h** y se ejecuta `copia_seguridad.sh` el **primer día de cada mes**.

Otra forma de poder programar tareas es guardar el script que quiere ejecutar en las siguientes carpetas de configuración de `cron`:

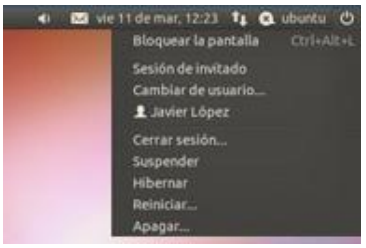
```
/etc/cron.hourly          # Ejecuta el script cada hora
/etc/cron.daily           # Ejecuta el script diariamente
/etc/cron.weekly          # Ejecuta el script semanalmente
/etc/cron.monthly         # Ejecuta el script mensualmente
```

Para asegurar el sistema sólo el usuario `root` puede modificar los scripts que ejecuta `crontab`.

Una ventaja muy interesante que permite `crontab` es que cada vez que se ejecuta la tarea manda un correo electrónico con el resultado de la ejecución de dicha tarea.

4.6.- Reinicio y parada del sistema.

El proceso de parada y reinicio del sistema se puede realizar de forma gráfica o por terminal. Para hacerlo de forma gráfica tan sólo hay que pulsar en el botón de apagar que se encuentra en la esquina superior derecha y en el menú que aparece seleccionar la operación a realizar.



Además, puedes utilizar comandos específicos para apagar el equipo como `halt` o `shutdown`, o se puede reiniciar el equipo ejecutando `reboot`.

5.- Monitorización del sistema.

Para conocer el comportamiento del sistema es necesario obtener información sobre las prestaciones de los diferentes subsistemas que lo componen. En GNU/Linux se dispone, por una parte, de una serie de comandos que proporcionan datos sobre el rendimiento del hardware y del sistema operativo y, por otra parte, de una aplicación cliente-servidor que registra los eventos que suceden en el equipo (syslog).

Si quieres monitorizar de forma automática muchos equipos lo mejor es que utilices las herramientas *Nagios* y *Centreon*.

Nagios http://www.nagios.org/
Centreon. http://www.centreon.com/

5.1.- Herramientas básicas.

Según el tipo de información que presentan, los comandos se pueden clasificar en:

- ✓ **Procesos**. Muestra información sobre los procesos que se están ejecutando en el sistema.
- ✓ **Almacenamiento**. Proporcionan información sobre la entrada y salida al subsistema de almacenamiento.
- ✓ **Memoria**. Proporcionan información sobre el espacio de memoria real y *swap.*
- ✓ **Red**. Facilitan estadísticas de uso de las interfaces de red.
- ✓ **Polivalentes**. Muestran información sobre distintos subsistemas del equipo.

En la siguiente tabla se muestra un resumen de las herramientas básicas de monitorización en GNU/Linux.

Herramientas básicas de monitorización en GNU/Linux	
Procesos	
ps	Muestra el estado de los procesos que se están ejecutando en el equipo.
Almacenamiento	
df	Muestra el espacio libre del sistema de ficheros.
du	Muestra el espacio ocupado a partir de un determinado directorio.
Memoria	
free	Proporciona información relativa a la cantidad de memoria física, espacio de swap libre y usado por el sistema operativo, estado de los buffers y memoria caché utilizada por el núcleo.
pmap	Proporciona información referente a la utilización de la memoria por parte de un determinado proceso.
Red	
ifstat	Muestra la estadística de tráfico de entrada y salida de las interfaces de red.
iftop	Muestra las conexiones de red de un equipo.
iptraf	Es una completa herramienta que permite mostrar las estadísticas de red en tiempo real.
netstat	Proporciona estadísticas e información de estado sobre tablas de rutas, interfaces de red, conexiones establecidas, etcétera.
ping	Permite comprobar el estado de una conexión.
traceroute	Permite obtener el camino que se sigue un paquete para establecer una comunicación

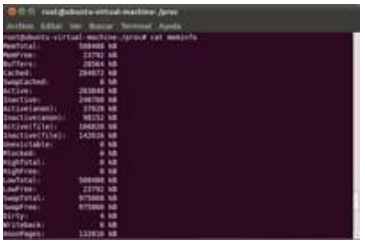
	con un destinatario, es decir, los routers que se atraviesan.
	Polivalentes
dstat	Permite realizar estadísticas de CPU, utilización de disco, red, paginación y estado del sistema.
iostat	Permite ver la carga de CPU y del disco duro.
top	Informa en tiempo real sobre la actividad del sistema. Proporciona información sobre la carga del sistema operativo, grado de utilización de la CPU, memoria y swap, y los procesos que se encuentran en ejecución.
vmstat	Muestra información sobre los procesos que se están ejecutando en el equipo, la memoria, las operaciones de entrada y salida a disco, y la utilización de la CPU. Es una aplicación clásica en los sistemas.
who	Permite ver de forma resumida el tiempo que lleva activo el sistema (uptime), la carga del sistema y la actividad de los usuarios que se encuentran conectados al sistema
xosview	Es una aplicación gráfica que proporciona información sobre el uso de CPU, memoria, cantidad de carga del sistema, red, interrupciones y swap en espacio de usuario.

5.2.- Directorio /proc.

El núcleo de Linux almacena información relativa a su funcionamiento en archivos situados en el directorio `/proc`, de tal forma que, para analizar el comportamiento de un sistema, también se puede recurrir a la consulta de los archivos de este sistema de ficheros. De hecho, prácticamente todas las herramientas analizadas obtienen sus datos de esta fuente.

Un ejemplo de la información que reside en `/proc` es:

- ✓ **Estado de la memoria** disponible en el fichero `/proc/meminfo`.
- ✓ **Sistema de comunicaciones** en `/proc/net`.
- ✓ **Datos referentes a un proceso** que se encuentran en un subdirectorio del estilo a `/proc/pid_del_proceso`.
- ✓ Etcétera.

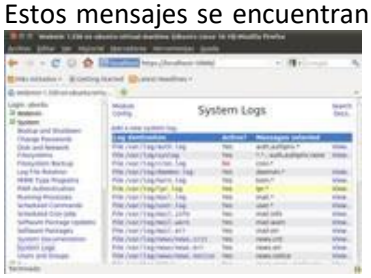


5.3.- Archivos de registro (syslog).

Hasta ahora se ha visto como ver el estado actual del sistema. Pero sin duda es muy importante saber lo que ha pasado en el servidor.

Existen muchos motivos por los que se pueden generar mensajes. Entre los más frecuentes se encuentran los fallos del servidor (por ejemplo, problema de hardware, fallo en un servicio), de autentificación (por ejemplo, fallo en la autentificación de un usuario) o por la utilización de un servicio (por ejemplo, petición de un cliente de una página web).

Estos mensajes se encuentran en los archivos de registro o archivos



log ubicados en el directorio `/var/log`. Por ejemplo, muchos mensajes son guardados en los ficheros */var/log/syslog* o en el `/var/log/messages`. Pero si un servicio genera muchos mensajes lo normal es que sean escritos en un fichero o carpeta separada como lo hace apache (`/var/log/httpd`) o el servidor de correo (`/var/log/mail`).

El registro de todos los mensajes del sistema lo realiza el servicio `syslogd` (o `rsyslogd`), el cuál no es exclusivo de los servicios del sistema sino que nosotros también podemos registrar nuestros propios mensajes usando `syslog`.

6.- Copias de seguridad.

Existen muchas herramientas que permiten realizar copias de seguridad del sistema. Estas herramientas se pueden clasificar en tres categorías: herramientas o comandos básicos, herramientas avanzadas de copias de seguridad y herramientas de clonación de sistemas.

La forma más habitual de realizar las copias de seguridad es utilizando los comandos básicos que proporciona el sistema (por ejemplo, `dump/restore`, `tar`). Con los comandos básicos se pueden realizar copias de seguridad de un equipo de forma individual. Además, existen herramientas avanzadas que permite centralizar y administrar todas las copias de seguridad de un sistema en un único servidor. Un ejemplo de este tipo de herramientas es *amanda*, que permite centralizar todas las copias de seguridad de los sistemas Windows y GNU/Linux de una empresa en un único servidor.

Otra forma muy útil de realizar copias de seguridad de sistemas enteros es la clonación de discos duros. La clonación de discos duros permite realizar una copia exacta de un disco duro o partición para poder restaurarlo en otro equipo de características similares. Este tipo de herramientas es muy útil en el caso de que quieras realizar una copia exacta de un servidor o restaurar muchos equipos con la misma configuración como por ejemplo, un aula de informática. En la tabla se muestran las herramientas de clonación de sistemas más importantes, destacando la herramienta *Clonezilla* que se verá más adelante.

Herramientas de clonación de discos	
Clone Maxx.	http://www.pcinspector.de/clonemaxx/info.htm?language=1
Clonezilla.	http://www.clonezilla.org/
Dubaron DiskImage.	http://www.dubaron.com/diskimage/
g4U.	http://www.feyrer.de/g4u/
NFGdump.	http://sourceforge.net/projects/nfgdump/
Norton Ghost.	http://es.norton.com/ghost
Partition Saving.	http://www.partition-saving.com/
Partimage.	http://www.partimage.org/
WinDD.	http://sourceforge.net/projects/windd/

6.1.- Comandos básicos.

Aunque muchas distribuciones de UNIX/Linux ofrecen sus propias herramientas para realizar copias de seguridad de todo tipo, casi todas estas herramientas suelen presentar un grave problema a la hora de recuperar ficheros cuando se trata de software propietario, por lo que si deseas restaurar total o parcialmente ficheros necesitas el propio programa para hacerlo. En determinadas situaciones, esto no es posible o es muy difícil.

Por este motivo, muchos administradores utilizan herramientas estándar para realizar las copias de seguridad de sus máquinas. Estas herramientas suelen ser tan simples como: `dump/restore`, `tar`, `dd`,`gzip`, `rsync` etcétera. Para mejorar las prestaciones de dichas herramientas se realizan, y programan, scripts para que se realicen las copias de forma automática.

A continuación se van a ver los comandos más utilizados para realizar copias de seguridad en sistemas GNU/Linux.

6.1.1.- El comando tar.

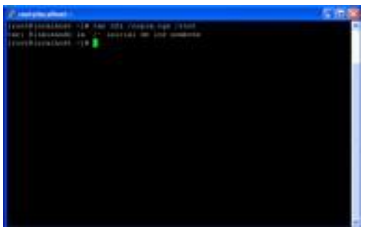
La utilidad tar (*Tape ARchiver*) es una herramienta de fácil manejo disponible en todas las versiones de UNIX/Linux que permite copiar ficheros individuales o directorios completos en un único fichero. Oficialmente fue diseñada para crear ficheros de cinta (esto es, para transferir ficheros de un disco a una cinta magnética y viceversa), aunque en la actualidad casi todas sus versiones pueden utilizarse para copiar a cualquier dispositivo o fichero, denominado *"contenedor"*.

En la siguiente tabla se muestran las opciones de tar más habituales. Algunas de ellas no están disponibles en todas las versiones de tar, por lo que es recomendable consultar la página del manual de esta orden antes de utilizarla.

Opciones de la orden tar	
Opción	**Acción**
c	Crea un contenedor.
x	Extrae ficheros de un contenedor.
t	Testea los ficheros almacenados en un contenedor.
r	Añade ficheros al final de un contenedor.
v	Modo verbose.
f	Especifica el nombre del contenedor.
z	Comprime o descomprime el fichero.

En primer lugar debe saber cómo crear contenedores con los ficheros deseados. Por ejemplo, para copiar el directorio /home/ en el fichero /root/copia.tgz hay que ejecutar el siguiente comando:

```
# tar cvf /root/copia.tgz /home/
```



La opción "`v`" no es necesaria, pero es útil para ver el proceso de empaquetamiento del fichero En muchas situaciones también resulta útil comprimir la información guardada (`tar` no comprime, sólo empaqueta) por lo que hay que utilizar las opciones "`cvfz`".
En lugar de indicar un único directorio con todos sus ficheros y subdirectorios es posible especificar múltiples ficheros (o directorios). Por ejemplo, la siguiente orden crea el fichero `/tmp/backup.tar`, que contiene `/etc/passwd` y `/etc/hosts*`.

```
# tar cvf /tmp/backup.tar /etc/passwd /etc/hosts*
```

Para recuperar los ficheros guardados en un fichero tar se utilizan las opciones "`xvf`" (o "`xvfz`" si se ha utilizado compresión con `gzip`). Puedes indicar el fichero o ficheros a extraer; si no lo haces se extraerán todos los ficheros. A continuación puedes ver un ejemplo:

```
# tar xvf /tmp/backup.tar /etc/passwd
```

En el ejemplo anterior, la restauración se ha realizado desde el directorio de trabajo, creando en él un subdirectorio `/etc` con los ficheros correspondientes en su interior.

Un fichero con extensión ".tar" se llama empaquetado ya que el fichero ocupa lo mismo que su contenido. Mientras que un fichero con extensión ".tar.gz" o ".tgz" esta comprimido y ocupa menos espacio que su contenido.

6.1.2.- El comando dd.

El comando `dd` permite realizar copias exactas (bit a bit) de discos duros, particiones o ficheros. La sintaxis de dd es la siguiente:

```
# dd if=fichero_origen of=fichero_destino
```

Antes de duplicar un disco duro debes saber los discos duros que tiene el sistema por lo que tienes que ejecutar el comando:

```
# fdisk -l
```

Por ejemplo, si desea clonar el disco duro que se encuentra en `/dev/sda` en el disco duro `/dev/sdb` ejecuta el comando:

```
# dd if=/dev/sda of=/dev/sdb
```

6.1.3.- rsync.

`rsync` es una aplicación para sistemas GNU/Linux que permite sincronizar carpetas de forma incremental y permite trabajar con datos comprimidos y cifrados. Mediante una técnica que se conoce como de delta encoding, permite sincronizar archivos y directorios entre dos máquinas de una red o entre dos ubicaciones en una misma máquina, minimizando el volumen de datos transferidos por la red.

Si deseas sincronizar dos carpetas locales ejecuta:

```
$ rsync -avz /carpeta_origen /carpeta_destino
```

donde se sincroniza el contenido de la `/carpeta_origen` en la `/carpeta_destino`. De forma análoga si quieres sincronizar las carpetas de dos equipos ejecuta:

```
$ rsync -avz /carpeta_origen 192.168.0.9:/carpeta_destino
```

Lógicamente, tanto el origen como el destino puede ser un equipo remoto siguiendo la sintaxis anterior.

6.1.4.- Backups sobre CD-ROM.

Cada vez es más común realizar copias de seguridad sobre discos compactos. Para poder grabar datos en un CD o DVD primero es necesario crear la imagen ISO (el "molde" del futuro CD-ROM). Una vez creada la imagen se graba en el disco utilizando un software de grabación.

Por ejemplo, si quieres realizar una copia del directorio `/home/`, en primer lugar ejecutarás `mkisofs` para crear una imagen con todos los ficheros y subdirectorios de los usuarios:

```
# mkisofs -o /root/imagen.iso /home/
```

Con esta orden se ha creado una imagen ISO denominada `/root/imagen.iso` y que contiene toda la estructura de directorios de `/home/`.

Una vez creada la imagen hay que grabarla en un CD-ROM, por ejemplo, mediante `cdrecord`:

```
# cdrecord /root/imagen.iso
```

Con esta orden el sistema detecta la grabadora de CD/DVD disponible en el sistema y realiza la grabación de la imagen ISO.

La mejor forma de automatizar una copia de seguridad es crear un script con todos los pasos de la copia de seguridad y programar su ejecución con crontab.

6.2.- Herramientas gráficas.

Además de realizar las copias de seguridad por comandos puede realizar la copia de seguridad del sistema mediante herramientas gráficas. Las herramientas más utilizadas son:

- ✓ **`Déjà-Dup`** es una aplicación para realizar copias de seguridad de forma sencilla e intuitiva. Entre sus características más importantes destaca la posibilidad de cifrar los datos para asegurar la privacidad, programación de las copias, permite almacenar las copias en diferentes destinos (por ejemplo, el servidor externo, local...).

  La instalación de Déjà-Dup se puede realizar a través de la herramienta `Agregar/quitar software` o ejecutando en el terminal el siguiente comando:

  ```
  # apt-get install deja-dup
  ```

  Para iniciar la aplicación de copias de seguridad puedes ejecutar el comando `deja-dup` en un terminal, o ir al menú `Aplicaciones->Herramientas del sistema` y ejecutar `Herramienta de respaldo Déjà-Dup`.
  Una vez iniciada la aplicación puede realizar dos acciones principales: `Respaldar` *(realizar) o* `restaurar` copias de seguridad.

- ✓ **`Brasero`** es el software de grabación de CD/DVD en sistemas GNU/Linux más utilizado. Su interfaz es bastante sencilla e intuitiva y permite, entre otras opciones, la grabación de CD/DVD de datos, CD de audio, duplicación de CD/DVD, etcétera.

  Normalmente Brasero se instala automáticamente al realizar la instalación del sistema con entorno gráfico, pero si necesitas instalarla hay que ejecutar:

  ```
  # apt-get install brasero
  ```

  Para iniciar `Brasero` puedes ejecutar en un terminal el comando `brasero` o ejecutar la ap icación `Grabador de discos Brasero` que se encuentra en el menú `Aplicaciones -> Sonido & Vídeo`.
  Una vez iniciada la aplicación puede utilizar la aplicación para realizar los diferentes proyectos de grabación.

- ✓ **`Clonezilla`** es la distribución LiveCD más potente y utilizada en la actualidad que permite realizar la clonación y restauración de sistemas. *Clonezilla* esta licenciado bajo GPL y entre sus características más importantes destacan:
  - ➔ Permite la clonación y restauración de particiones o de discos duros completos.

- Utiliza diferentes sistemas de ficheros como FAT32, NTFS y ext3 por lo que permite trabajar con cualquier instalación GNU/Linux o Windows.
- Permite realizar y restaurar las copias de seguridad utilizando diferentes medios como, por ejemplo, discos duros locales, servidores Samba, servidores SSH, llaveros USB, etcétera.
- Es fácil y sencilla de utilizar.

Para empezar a utilizar clonezilla para clonar o restaurar un equipo tienes que descargar la imagen ISO de clonezilla de su web oficial y la grabas en un CD. Inicias el LiveCD en el equipo que deseas clonar y en el menú de arranque seleccione la opción cuya resolución se adapte mejor a nuestras necesidades. A continuación se inicia el asistente que te guía para poder clonar o restaurar una copia del equipo.

En la página oficial de Clonezilla puedes descargar la imagen ISO.
http://clonezilla.org/