Creación y configuración de sistemas de archivos opcionales

Se pueden configurar y administrar sistemas de archivos en soportes extraíbles o gestionados por servidores remotos en los sistemas Linux.

1. Servicio de montaje automático.

Como se ha visto, el montaje de un sistema de archivos puede ser configurado para que se efectúe durante el arranque del sistema. Sin embargo, algunas categorías de sistemas de archivos pueden ser difícilmente montadas sistemáticamente, en particular:

- Sistemas de archivos en soporte extraíble

  Es el caso de los CD-ROM, pendrive USB y otros soportes físicos que pueden no encontrarse conectados durante el arranque del sistema, o que pueden ser cambiados durante la actividad.

- Sistemas de archivos en red

  Estos sistemas de archivos, gestionados por sistemas remotos, no son obligatoriamente accesibles desde el arranque del sistema, y su inaccesibilidad podría obstaculizar un arranque correcto en caso de configuración en montaje inicial.

Hay que usar otro método, para esas categorías de sistemas de archivos, para hacer que sean accesibles para las aplicaciones. Se puede recurrir al comando `mount`, pero a menudo este comando está reservado a los administradores, y su sintaxis puede no ser sencilla para un usuario normal.

Una solución puede ser configurar esas categorías de sistemas de archivos en montaje automático: esto consiste en asociar un punto de montaje a un directorio; en el momento en que una aplicación solicita acceso a un elemento de ese directorio o al mismo directorio, el sistema de archivos correspondiente se montará.

Esta funcionalidad la realiza un servicio de montaje automático, gestionado por un

servicio específico o por `systemd`.

La mayoría de los entornos gráficos de usuario (Gnome, KDE, etc.) pueden gestionar el montaje automático de los sistemas de archivos en soportes extraíbles y en sistemas de archivos en red.

a. Configuración del servicio autofs/automount

Este servicio lo garantiza el servicio `automountd`. El paquete de software correspondiente se llama `autofs`, que viene del nombre del script de gestión de ese servicio.

Varios archivos de configuración, llamados **archivos de mapa** (*map files*), definen los sistemas de archivos que tienen que ser gestionados automáticamente, sus puntos de montaje y los parámetros de esos montajes.

Hay dos niveles principales de configuración:

- El archivo de mapa principal (*master map*): se lee durante el arranque del servicio, es donde se definen diferentes tipos de montajes.
- Los archivos de mapa secundarios: estos archivos, opcionales, permiten configurar diferentes montajes de sistemas de archivos, en relación con las configuraciones definidas en el archivo de mapa principal.

Mapa principal

Por defecto, se trata del archivo `/etc/auto.master`.

Especifica los diferentes elementos en montaje automático tomados en cuenta por el servicio de montaje automático.

Formato de línea

```
punto_montaje mapa [ -opciones ... ]
```

`punto_montaje`	Camino de acceso del directorio que se tiene que vigilar o valor predefinido.
`mapa`	Mapa asociado al punto de montaje.
`-opciones`	Opcion(es) de montaje automático.

Descripción

El campo `punto_montaje` contiene a menudo:

- Un camino de acceso de directorio

  Se trata del punto de montaje principal, a partir del cual serán montados los sistemas de archivos definidos en el archivo de mapa especificado en el segundo campo (mapa indirecto) o sistemas de archivos exportados por servidores remotos (mapa de red). Si el directorio no existe, se creará durante el arranque del servicio y será suprimido cuando este servicio se pare.

- /-

  Este valor predefinido se usa cuando el punto de montaje principal está definido en el archivo de mapa especificado en el segundo campo (mapa directo).

El campo `mapa` indica el mapa que se tiene que asociar al punto de montaje. Estos mapas pueden ser de tipos distintos, entre los cuales se encuentran a menudo:

- Camino de acceso del archivo de configuración del mapa que contiene el punto de montaje (mapa directo), el campo anterior contiene el valor predefinido `/-`.
- Camino de acceso al archivo de configuración del mapa indirecto: el punto de montaje principal está definido en el campo anterior, el archivo de mapa define uno o distintos subdirectorios del directorio de ese punto de montaje principal.
- Valor predefinido `-hosts`: mapa de red. Las exportaciones de red (NFS, CIFS...) de un servidor remoto se montarán, en el punto de montaje especificado definido en el campo anterior, en cuanto sea necesario acceder al subdirectorio del punto de montaje principal que lleva el nombre del servidor.

El campo `-opciones` puede contener opciones de montaje automático, como por ejemplo:

```
--timeout numero_segundos
```

Número de segundos de inactividad que se esperará antes de desmontar el sistema de archivos. Sin esta opción, se utilizará el valor por defecto definido en `/etc/sysconfig/autofs` .

Hay otros formatos de definiciones de montajes automáticos, mapas NIS, LDAP, etc. Para obtener más información, ejecutar el comando `man 5 auto.master` .

Mapa indirecto

Un mapa indirecto (*indirect map*) asocia un sistema de archivos a un subdirectorio de un directorio gestionado por el servicio. Provoca el montaje automático de ese sistema de archivos en ese subdirectorio en el momento en que una aplicación quiera acceder a este subdirectorio. Este último se creará automáticamente si fuera necesario, en este caso será suprimido cuando ya no sea usado (después de un tiempo de espera) y el sistema de archivos se haya desmontado automáticamente.

Formato de línea

```
NombreDir [ opciones ] ArchivoEspecial
```

`NombreDir`	Subdirectorio virtual para el montaje.
`opciones`	Opciones de montaje.
`ArchivoEspecial`	Sistema de archivos que se montará.

Descripción

El campo `NombreDir` contiene el nombre del subdirectorio donde se montará el sistema de archivos.

Las opciones especificadas opcionalmente en el campo `opciones` serán las del montaje del sistema de archivos.

El campo `ArchivoEspecial` define el sistema de archivos que se va a montar. Puede contener el camino de acceso a un archivo especial de tipo bloque o a un volumen lógico, o incluso un camino de acceso de red, lo que permitirá configurar el montaje automático de sistemas de archivos gestionados por sistemas remotos (Linux u otros).

b. Arranque y paro del servicio

El script de control `/etc/init.d/autofs` (o su equivalente `systemd`) gestiona el servicio de montaje automático `automountd`, el cual es ejecutado en el arranque del sistema.

Si se modifica el archivo de mapa principal mientras el servicio está activo, habrá que reiniciarlo para que tenga en cuenta las modificaciones hechas.

c. Ejemplo

Este ejemplo utiliza la configuración por defecto del servicio `automountd` de una distribución CentOS 8.

Instalación del servicio

Si este último no ha sido incluido en la estación inicial, habrá que instalar el paquete `autofs`:

```
yum search autofs
autofs.x86_64 : A tool for automatically mounting and unmounting filesystems
yum install autofs.x86_64
```

La instalación crea dos archivos de configuración: `/etc/auto.master` y `/etc/auto.misc`.

Mapa principal

El archivo de mapa principal `/etc/auto.master` contiene:

```
/net -hosts
/misc   /etc/auto.misc
```

La primera línea define los automontajes de red, en el directorio `/net`. Por ejemplo, si una aplicación quisiera acceder al directorio `/net/srv1`, y que el servidor `srv1` exportara sistemas de archivos de red, estos serían accesibles automáticamente a partir del directorio `/net/srv1` (ver capítulo Compartición de archivos, sección Configuración de un servidor NFS).

La segunda línea define un mapa indirecto. El servicio de auto montaje gestiona el punto de montaje principal `/misc` (creado y suprimido automáticamente si no existiera) y gestionará los montajes automáticos en los subdirectorios de ese punto de montaje en el archivo de mapa indirecto `/etc/auto.misc`.

Mapa indirecto

El archivo de mapa indirecto `/etc/auto.misc` contiene:

```
cd    -fstype=ISO 9660,ro,nosuid,nodev       /dev/cdrom
```

El primer campo especifica el subdirectorio del punto de montaje principal, asociado al sistema de archivos que se va a montar. Si una aplicación solicita acceder al directorio `cd` del punto de montaje principal (es decir `/misc/cd`), el sistema de archivos `/dev/cdrom` se montará automáticamente en ese directorio. Si éste no existe, será creado automáticamente durante el montaje, y suprimido cuando se desmonte. Este desmontaje se efectuará automáticamente una vez que pase el tiempo de inactividad en el interior del subdirectorio.

El segundo campo especifica las opciones de montaje, en particular:

`-fstype=ISO 9660` : tipo de sistema de archivos (CD-ROM).

`ro`: solo lectura.

Gestión del servicio

Comprobar si el servicio está iniciado:

```
service autofs status
not started
```

El servicio no está iniciado.

```
ls -ld /misc
ls: no se puede acceder a '/misc: No existe el fichero o el directorio
```

El punto de montaje principal no existe.

Se inicia el servicio:

```
service autofs start
Redirecting to /bin/systemctl start autofs.service
ls -ld /misc
```

```
drwxr-xr-x. 2 root root 0  3 marzo  17:01 /misc
```

El directorio se ha creado automáticamente, está vacío:

```
ls -l /misc
```

Se solicita el acceso al subdirectorio cd:

```
ls -l /misc/cd
total 6
dr-xr-xr-x. 3 root root 2048  3 enero 21:30 EFI
dr-xr-xr-x. 3 root root 2048  3 enero 21:30 images
drwxrwxr-x. 2 root root 2048  3 enero 21:30 isolinux
```

El sistema de archivos del CD-ROM ha sido montado automáticamente en el directorio `/misc/cd` creado también automáticamente.

Después de un minuto:

```
sleep 60
ls -l /misc
```

El sistema de archivos del CD-ROM ha sido desmontado automáticamente y el directorio `/misc/cd` suprimido.

Se para el servicio:

```
service autofs stop
Redirecting to /bin/systemctl stop autofs.service
ls -ld /misc

ls: no se puede acceder a '/misc: No existe el fichero o el directorio
```

El punto de montaje principal ha sido suprimido.

2. Montaje automático con systemd

`systemd` también permite gestionar el montaje automático del sistema de archivos.

Usa para ello unidades de tipo `automount`.

a. Unidad de montaje automático systemd

Cada punto de montaje automático es gestionado por un archivo **unidad de montaje automático**, cuyo nombre debe presentarse con un sufijo `.automount` y debe corresponder al camino de acceso del punto de montaje, reemplazando el separador `/` del camino de acceso por un `-`.

Por ejemplo, para el punto de montaje `/home/pba/datos/autocd`, el nombre será: `home-pba-datos-autocd.automount`.

Para cada unidad de montaje automático debe haber una **unidad de montaje systemd**.

Por ejemplo, para la unidad de montaje automático `home-pba-datos-autocd.automount`, se debe encontrar una unidad de montaje `home-pba-datos-autocd.mount`.

Una vez que la unidad de montaje ha sido activada e iniciada, en el momento en que una aplicación solicite acceder al punto de montaje, la unidad de montaje correspondiente será activada por `systemd`, lo que provocará el montaje del sistema de archivos correspondiente en el punto de montaje.

La unidad de montaje automático debe contener una sección `[Automount]`. Dicha sección puede contener los campos siguientes:

- `Where=PuntoMontaje`: este campo es obligatorio. `PuntoMontaje` es el camino de acceso al punto de montaje, que debe corresponder al nombre de la unidad de montaje automático. Si el directorio no existe, será creado automáticamente.
- `DirectoryMode=Permisos`: este campo es opcional y define los permisos (en binario) del directorio de montaje si tiene que ser creado. Por defecto: `0755`.

- `TimeoutIdleSec=IdleTime` : este campo, opcional, define la duración del tiempo de inactividad que provocará el desmontaje del sistema de archivos. Se puede especificar `IdleTime` en número de segundos o en formato "`Xmin Ys`", `0` desactiva el desmontaje automático. Por defecto, el desmontaje automático está desactivado.

b. Ejemplo de montaje automático con systemd

Se utilizará la unidad de montaje del ejemplo relativo a los montajes `systemd`, modificándola para asociarle una unidad de automontaje en el directorio `/var/cd`.

Antes hay que modificar la unidad de montaje para que ya no sea montada automáticamente:

```
vi /etc/systemd/system/var-cd.mount
[Unit]
Description=CD-ROM
 [Mount]
What=/dev/sr0
Where=/var/cd
#### No se montará automáticamente
#[Install]
#WantedBy=multi-user.target
```

Se desmonta el sistema de archivos CD-ROM:

```
umount /var/cd
```

Se reactiva la unidad de montaje para que los cambios sean tomados en cuenta:

```
systemctl disable var-cd.mount
Removed /etc/systemd/system/multi-user.target.wants/var-cd.mount.
systemctl enable var-cd.mount
The unit files have no installation config (WantedBy, RequiredBy, Also, Alias
settings in the [Install] section, and DefaultInstance for template units).
```

```
This means they are not meant to be enabled using systemctl.
Possible reasons for having this kind of units are:
1) A unit may be statically enabled by being symlinked from another unit's
 .wants/ or .requires/ directory.
2) A unit's purpose may be to act as a helper for some other unit which has
 a requirement dependency on it.
3) A unit may be started when needed via activation (socket, path, timer,
 D-Bus, udev, scripted systemctl call, ...).
4) In case of template units, the unit is meant to be enabled with some
 instance name specified.
```

Ahora hay que crear la unidad de montaje automático y configurarla para qué se inicie automáticamente en modo multiusuario:

```
vi /etc/systemd/system/var-cd.automount
[Unit]
Description=Montaje automático CD-ROM
[Automount]
Where=/var/cd
DirectoryMode=0444
TimeoutIdleSec=60
[Install]
WantedBy=multi-user.target
```

Se activa la unidad de montaje:

```
systemctl enable var-cd.automount
Created symlink /etc/systemd/system/multi-user.target.wants/
var-cd.automount ? /etc/systemd/system/var-cd.automount.
```

Se comprueba que el directorio `/var/cd` esta vacío y que el sistema de archivos no está montado:

```
ls /var/cd
```

```
mount | grep sr0
```

Se inicia manualmente la unidad de montaje automático:

```
systemctl start var-cd.automount
```

Se comprueba que, en cuanto accedemos al directorio `/var/cd`, ya no está vacío:

```
ls /var/cd
EFI  images  isolinux
```

Se espera un minuto.

```
sleep 60
mount | grep sr0
```

El sistema de archivos ha sido desmontado automáticamente.

Un usuario solicita acceso al directorio `/var/cd`, provocando de esta manera el montaje del sistema de archivos:

```
su pba
ls /var/cd
EFI  images  isolinux
exit
```

El CD-ROM ha sido montado automáticamente.

```
mount | grep sr0
/dev/sr0 on /var/cd type ISO 9660 (ro,relatime,nojoliet,check=s,
map=n,blocksize=2048)
```

3. Sistemas de archivos de dispositivos extraíbles

Los CD-ROM están generalmente estructurados mediante un sistema de archivos normalizado, de tipo ISO 9660. Este tipo de sistema de archivos está soportado por los diferentes sistemas operativos, lo que permite poder compartir su contenido entre distintas máquinas.

Linux soporta este formato, incluso con algunas extensiones de la norma.

El formato UDF (*Universal Disk Format*) es el sucesor de ISO 9660 para los soportes de tipo CD-ROM/DVD y pendrive USB. Permite trabajar con volúmenes y archivos más grandes. También está soportado en Linux.

Se puede copiar la integralidad del sistema de archivos de un CD-ROM, bajo la forma de una imagen ISO almacenada como un archivo. Esta imagen puede ser copiada o descargada desde otro sistema, para ser grabada después en un CD-ROM.

También se puede crear una imagen ISO copiando dentro de ella directorios y archivos desde la arborescencia global del sistema de archivos.

Los sistemas de archivos de los CD-ROM/DVD no son tomados en cuenta de la misma manera por los distintos sistemas aplicativos. Los sistemas operativos implementan extensiones de la norma ISO 9660, la mayoría de estas extensiones pueden ser soportadas por Linux, a menudo a través del uso de opciones. Las principales extensiones son las siguientes:

- El Torito: se trata de una extensión de la norma ISO 9660, que define cómo hacer que un CD-ROM/DVD sea arrancable.
- Joliet: la norma ISO 9660 limita los nombres de los archivos y de los directorios al formato 8.3 (8 caracteres máximo, seguidos opcionalmente por un punto y 3 caracteres máximo). La extensión Joliet, de origen Windows, permite gestionar nombres largos de hasta 64 caracteres UNICODE.

- Rock Ridge: esta extensión de la norma, de origen POSIX, permite gestionar archivos y directorios con atributos de tipo Unix: nombres largos, hasta 255 caracteres, derechos de acceso, propietario y grupo, etc.
- HFS (*Hierarchical File System*): se trata del sistema de archivos desarrollado por Apple para su sistema operativo macOS. Se utiliza en los discos duros, pero también en CD-ROM/DVD.

a. Recuperar la lista de información de un CD-ROM/DVD

El comando `isoinfo` permite obtener las características de un sistema de archivos de tipo CD-ROM/DVD o de una imagen ISO 9660.

Sintaxis

```
isoinfo [ -d ] [ dev=ArchivoEspecial | -i ArchivoImagen ]
```

Parámetros principales

`-d`	Muestra los atributos del sistema de archivos.
`dev=ArchivoEspecial`	Archivo especial asociado al lector de CD-ROM/DVD.
`-i ArchivoImagen`	Camino de acceso del archivo imagen.

Descripción

El comando muestra los atributos del sistema de archivos del CD-ROM/DVD (opcion `dev=`), o de una imagen ISO (opcion `-i`). La opción `-d` muestra los atributos.

Ejemplo

Se utiliza el comando `lsscsi` para identificar el archivo especial asociado al lector de CD-ROM/DVD:

```
lsscsi
[0:0:0:0]   disk    ATA     ST320LT020-9YG14 HPM1  /dev/sda
[1:0:0r:0]  cd/dvd  hp      DVDRAM GT50N     MP00  /dev/sr0
```

El lector está asociado al archivo especial `/dev/sr0`.

Para mostrar las características del CD-ROM introducido en el lector, se utiliza `isoinfo`:

```
isoinfo -d dev=/dev/sr0
CD-ROM is in ISO 9660 format
System id: LINUX
Volume id: CentOS-8-1-1911-x86_64-dvd
Volume set id:
Publisher id:
Data preparer id:
Application id: GENISOIMAGE ISO 9660/HFS FILESYSTEM CREATOR (C) 1993
E.YOUNGDALE (C) 1997-2006 J.PEARSON/J.SCHILLING (C) 2006-2007 CDRKIT TEAM
Copyright File id:
Abstract File id:
Bibliographic File id:
Volume set size is: 1
Volume set sequence number is: 1
Logical block size is: 2048
Volume size is: 305158
El Torito VD version 1 found, boot catalog is in sector 44
Joliet with UCS level 3 found
Rock Ridge signatures version 1 found
Eltorito validation header:
  Hid 1
  Arch 0 (x86)
  ID ''
  Key 55 AA
  Eltorito defaultboot header:
    Bootid 88 (bootable)
    Boot media 0 (No Emulation Boot)
    Load segment 0
    Sys type 0
    Nsect 4
    Bootoff 4936B 299883
```

Se ve que se trata de un CD-ROM arrancable (`Eltorito`), con un tamaño de bloques de 2048 bytes y un tamaño de 305158 bloques, por lo tanto unos 620 MB.

b. Copia de un CD-ROM/DVD en una imagen ISO

Para copiar el sistema de archivos de un CD-ROM o de un DVD en una imagen ISO en el disco, se puede usar un programa en modo gráfico (GNOME Disks, Brasero...) o usar el comando de gestión de discos de bajo nivel `dd`.

Una vez creada la imagen ISO, esta puede ser grabada en un CD-ROM/DVD o montada directamente en la arborescencia global, como si se tratara de un CD-ROM/DVD.

Ejemplo

Se utiliza el comando `dd` para copiar el sistema de archivos del CD-ROM del ejemplo anterior en un archivo de imagen ISO:

```
dd if=/dev/sr0 of=./CentOS-8.1.1911-x86_64-boot.iso
1223256+0 registros leídos
1223256+0 registros escritos
626307072 bytes (626 MB, 597 MiB) copiados, 254,328 s, 2,5 MB/s
ls -l *iso
-rw-r--r--. 1 root root 626307072  5 marzo  18:18 CentOS-8.1.1911-x86_64-boot.iso
```

Se muestran los atributos de la imagen:

```
isoinfo -d -i ./CentOS-8.1.1911-x86_64-boot.iso
CD-ROM is in ISO 9660 format
System id: LINUX
Volume id: CentOS-8-1-1911-x86_64-dvd
```

```
Volume set id:
Publisher id:
Data preparer id:
Application id: GENISOIMAGE ISO 9660/HFS FILESYSTEM CREATOR (C) 1993
E.YOUNGDALE (C) 1997                -2006 J.PEARSON/J.SCHILLING (C)
2006-2007 CDRKIT TEAM
Copyright File id:
Abstract File id:
Bibliographic File id:
Volume set size is: 1
Volume set sequence number is: 1
Logical block size is: 2048
Volume size is: 305158
El Torito VD version 1 found, boot catalog is in sector 44
Joliet with UCS level 3 found
Rock Ridge signatures version 1 found
Eltorito validation header:
  Hid 1
  Arch 0 (x86)
  ID ''
  Key 55 AA
  Eltorito defaultboot header:
    Bootid 88 (bootable)
    Boot media 0 (No Emulation Boot)
    Load segment 0
    Sys type 0
    Nsect 4
    Bootoff 4936B 299883
[root@beta ~]# isoinfo  -i ./CentOS-8.1.1911-x86_64-boot.iso
[root@beta ~]# isoinfo -d -i ./CentOS-8.1.1911-x86_64-boot.iso
CD-ROM is in ISO 9660 format
System id: LINUX
Volume id: CentOS-8-1-1911-x86_64-dvd
Volume set id:
Publisher id:
Data preparer id:
Application id: GENISOIMAGE ISO 9660/HFS FILESYSTEM CREATOR (C) 1993
E.YOUNGDALE (C) 1997 -2006 J.PEARSON/J.SCHILLING (C) 2006-2007 CDRKIT TEAM
Copyright File id:
Abstract File id:
```

```
Bibliographic File id:
Volume set size is: 1
Volume set sequence number is: 1
Logical block size is: 2048
Volume size is: 305158
El Torito VD version 1 found, boot catalog is in sector 44
Joliet with UCS level 3 found
Rock Ridge signatures version 1 found
Eltorito validation header:
  Hid 1
  Arch 0 (x86)
  ID ''
  Key 55 AA
  Eltorito defaultboot header:
    Bootid 88 (bootable)
    Boot media 0 (No Emulation Boot)
    Load segment 0
    Sys type 0
    Nsect 4
    Bootoff 4936B 299883
```

Se monta la imagen:

```
mount CentOS-8.1.1911-x86_64-boot.iso ./cd
mount: /root/cd: ATENCIÓN: origen protegido contra escritura; se monta como solo lectura.
ls -l cd
total 6
dr-xr-xr-x. 3 root root 2048  3 enero 21:30 EFI
dr-xr-xr-x. 3 root root 2048  3 enero 21:30 images
drwxrwxr-x. 2 root root 2048  3 enero 21:30 isolinux
```

El archivo de imagen contiene un sistema de archivos ISO 9660 coherente.

c. Creación de una imagen ISO

El comando `mkisofs` permite crear una imagen ISO, no a partir de un CD-ROM o de un

DVD, sino a partir de directorios o de archivos de la arborescencia global. Esta imagen ISO creada de esta manera se puede montar con el comando `mount`, se puede grabar en un CD-ROM o DVD, y puede usarse desde una máquina virtual.

Sintaxis

```
mkisofs [ -opciones ] -o imagen directorio
```

Parámetros principales

`-J`	Genera registros para la extensión Joliet.
`-apple`	Asegura la compatibilidad con MacOS.
`-o imagen`	El archivo ISO generado.
`-follow-links`	Seguir los enlaces simbólicos.
`directorio`	Directorio a partir del cual se generará la imagen ISO.

Descripción

La opción `-J` permite una mejor compatibilidad con Windows, la opción `-apple` asegurará una mejor compatibilidad con macOS. La opción `-follow-links` fuerza a copiar los enlaces simbólicos, si no se ignoran.

El directorio especificado estará en la raíz de la imagen y toda arborescencia será copiada.

`mkisofs` es el nombre histórico del comando. Ha sido renombrado como `genisoimage` . Se mantiene el nombre `mkisofs` para la retrocompatibilidad, bajo la forma de un enlace al ejecutable de `genisoimage` .

Ejemplo

Creación de una imagen ISO a partir de un directorio:

```
mkisofs -follow-links -o srvetc.iso /etc
Warning: -follow-links does not always work correctly; be careful.
I: -input-charset not specified, using utf-8 (detected in locale settings)
Already cached directory seen (/etc/ssl/certs)
Already cached directory seen (/etc/xdg/systemd/user)
Already cached directory seen (/etc/rc0.d)
Already cached directory seen (/etc/rc1.d)
[...]
 28.96% done, estimate finish Thu Mar  5 19:15:31 2020
 57.86% done, estimate finish Thu Mar  5 19:15:31 2020
 86.80% done, estimate finish Thu Mar  5 19:15:31 2020
Total translation table size: 0
Total rockridge attributes bytes: 0
Total directory bytes: 811008
Path table size(bytes): 5918
Max brk space used 1be000
17293 extents written (33 MB)
```

Se ha creado la imagen ISO y se muestran sus atributos:

```
isoinfo -d -i ./srvetc.iso
CD-ROM is in ISO 9660 format
System id: LINUX
Volume id: CDROM
Volume set id:
```

Publisher id:
Data preparer id:
Application id: GENISOIMAGE ISO 9660/HFS FILESYSTEM CREATOR (C) 1993
E.YOUNGDALE (C) 1997-2006 J.PEARSON/J.SCHILLING (C) 2006-2007 CDRKIT TEAM
Copyright File id:
Abstract File id:
Bibliographic File id:
Volume set size is: 1
Volume set sequence number is: 1
Logical block size is: 2048
Volume size is: 17293
NO Joliet present
NO Rock Ridge present

Se monta la imagen:

mount srvetc.iso ./cd
mount: /root/cd : ATENCIÓN: origen protegido contra escritura; se monta como
solo lectura.
ls -l cd | more
total 1402
-r-xr-xr-x. 1 root root 18 29 feb 16:00 adjtime
-r-xr-xr-x. 1 root root 1518 10 sept 2018 aliases
dr-xr-xr-x. 1 root root 2048 29 feb 15:52 alsa
dr-xr-xr-x. 1 root root 2048 29 feb 15:56 alternat
-r-xr-xr-x. 1 root root 541 8 nov 16:47 anacront
-r-xr-xr-x. 1 root root 55 8 nov 16:21 asound.con
-r-xr-xr-x. 1 root root 1 11 may 2019 at.den
dr-xr-xr-x. 1 root root 2048 29 feb 16:03 audit
dr-xr-xr-x. 1 root root 2048 29 feb 16:00 authsele
-r-xr-xr-x. 1 root root 14790 8 nov 16:25 autofs.con
-r-xr-xr-x. 1 root root 232 8 nov 16:25 autofs_l.con
-r-xr-xr-x. 1 root root 1040 8 nov 16:25 auto.mas
dr-xr-xr-x. 1 root root 2048 8 nov 16:25 auto_mas.d
-r-xr-xr-x. 1 root root 524 8 nov 16:25 auto.mis
-r-xr-xr-x. 1 root root 902 8 nov 16:25 auto.net
-r-xr-xr-x. 1 root root 2191 8 nov 16:25 auto.smb
dr-xr-xr-x. 1 root root 2048 29 feb 15:52 avahi
[...]

El directorio raíz de la imagen corresponde a `/etc` de la arborescencia global:

```
diff cd/hosts /etc/hosts
```

Los dos archivos son idénticos.

4. Cifrado de sistemas de archivos

Los derechos de acceso de un sistema de archivos permiten determinar qué usuarios pueden acceder, escribir, leer o suprimir un archivo o un directorio. Incluso usando atributos extendidos gracias a ACL (*Access Control List*) o a través de los controles suplementarios de SELinux, no se puede asegurar la confidencialidad de los datos almacenados en un sistema de archivos:

- El administrador del sistema (UID 0) siempre podrá acceder al contenido de un directorio o de un archivo, directa o indirectamente.
- Se podría desplazar un disco duro físicamente y conectarlo a otro sistema. Esto haría que el administrador de este sistema pudiese acceder a los datos contenidos en el disco.

Una solución eficaz para proteger los datos almacenados es cifrarlos. Para poder descifrar estos datos hay que proporcionar una clave de descifrado, generalmente bajo la forma de una contraseña (passphrase). Incluso el administrador del sistema (UID 0) deberá proporcionar la clave de descifrado para acceder a los datos y, por lo tanto, aunque el disco se cambie de máquina físicamente, los datos quedarán protegidos.

Se puede hacer el cifrado de datos directamente en el archivo, el espacio de almacenaje o el sistema de archivos. Puede ser transparente para los usuarios o no.

a. Principios del cifrado en Linux

Hay diferentes técnicas de cifrado de datos implementadas en Linux. Las más usadas se basan en dos estándares:

dm-crypt (device mapper encryption)

Este módulo del núcleo se encarga de las operaciones de gestión del cifrado en un periférico en modo bloque. Puede tratar de manera transparente el cifrado de discos fijos o extraíbles, a nivel de disco, de la partición, del volumen lógico o del archivo.

Solo opera en modo núcleo y puede usarse con dos aplicaciones principales: cryptsetup y cryptmount.

LUKS (Linux Unified Key Setup)

Se trata de una especificación estándar de cifrado de discos. Este modo de cifrado es interoperable con otros sistemas operativos y gestiona contraseñas múltiples para un mismo conjunto de datos. El programa `cryptsetup` de Linux utiliza esta especificación.

b. Cifrado de un archivo

Para proteger un archivo sensible, una solución simple consiste en cifrar su contenido. Se puede usar para ello un comando de tipo `crypt`.

El comando `crypt`, originario de Unix, ha sido reemplazado en Linux por una implementación más moderna, `mcrypt`. Este último está presente en las distribuciones Debian en el paquete `libmcrypt4`. En las distribuciones de tipo CentOS, se encuentra en el paquete `mcrypt`, en el repositorio opcional EPEL (Extra Packages for Enterprise Linux). Con el paquete, además del comando `mcrypt`, un script implementa un pseudocomando `crypt`, para asegurar la retrocompatibilidad.

Ejemplo

En una distribución Debian, cifrado de un archivo con el comando `mcrypt`:

Instalación del paquete:

```
apt-get install  mcrypt
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
 libmcrypt4
Paquetes sugeridos:
 libmcrypt-dev
Se instalarán los siguientes paquetes NUEVOS:
 libmcrypt4 mcrypt
0 actualizados, 2 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 136 kB de archivos.
Se utilizarán 421 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
[...]
```

Uso del comando por un usuario:

```
cp /etc/hosts mis-hosts.txt
mcrypt mis-hosts.txt
Enter the passphrase (maximum of 512 characters)
Please use a combination of upper and lower case letters and numbers.
Enter passphrase: XXXXXXX
Enter passphrase: XXXXXXX
File mis-hosts.txt was encrypted.
```

El comando ha creado una versión cifrada del archivo original:

```
ls -l mis-hosts.txt*
-rw-r--r-- 1 pba pba 191 marzo   8 14:05 mis-hosts.txt
-rw------- 1 pba pba 301 marzo   8 14:05 mis-hosts.txt.nc
od -c  mis-hosts.txt.nc
0000000  \0   m 003   @   r   i   j   n   d   a   e   l   -   1   2   8
0000020  \0    \0   c   b   c  \0   m   c   r   y   p   t   -   s   h
0000040   a   1  \0 025 255   L   S 242 220 204 306 247 027   M 347 336
0000060 322 207 272 252 244 246 366 271   s   h   a   1  \0   p 003   ,
```

```
0000100 303 023 321 342   v   X   h 023 270 315 343   `   M 250 373 316
0000120   F   L 257 353 236 354   ) 202 223   l   =   K     307 332 274
0000140 317 242   c 324 227 206 366 357 225   j 300 334 372   f 273 036
0000160 346   ,   ` 365 321 022 336 376   /   y   e   :   & 034 216   A
0000200 275   Y   M   B 236 304     223   v   k 375   % 306 266   = 245
0000220 332   $ 202 002   r 034   H   f   C 251 302 263 303   k 264 255
0000240 357 371 332   r 327 273   z   [ 225 300   j   X 030 020 242 006
0000260 371 004 274 242 245  \n   _   8 367 234 017   8 256 335 253 037
0000300   j 256   e 207   w   ) 222   { 217   ; 365 375   q 030   `   X
0000320 340   r   9   e   )   !   . 257   &   # 262 346 356  \f 252   :
0000340 352   1   $ 025 002 233 250  \v 317 267   w   R   !   & 216   [
0000360 246 217 021 205 232 371 214   Z 377   W   . 340 207 211  \v 342
0000400 315 231   W  \f   ?   N 225 312 261   } 205  \a   t   < 331   g
0000420   #   `   d  \f   N   T   v 213   k 276 004 344 021   Q   V 264
0000440   k 273 256   l 300   0   ( 342 225 243 343 340 332
0000455
```

Se renombra el archivo original:

```
mv mis-hosts.txt mis-hosts.txt.ori
```

Para descifrar el archivo, hay que usar el comando `mcrypt`, con la opción `-d` (decrypt) y teclear la frase de la contraseña (passphrase). El comando recreará el archivo original:

```
mcrypt -d mis-hosts.txt.nc
Enter passphrase: XXXXXXX
File mis-hosts.txt.nc was decrypted.
```

Se comprueba que el archivo ha sido descifrado:

```
ls -l mis*
-rw------- 1 pba pba 191 marzo  8 14:05 mis-hosts.txt
-rw------- 1 pba pba 301 marzo  8 14:05 mis-hosts.txt.nc
-rw-r--r-- 1 pba pba 191 marzo  8 14:05 mis-hosts.txt.ori
cmp mis-hosts.txt mis-hosts.txt.ori
```

Los dos archivos son idénticos.

c. Cifrado del espacio de almacenamiento

Para una protección a la vez general y menos limitada para el usuario, se puede cifrar el espacio de almacenamiento de un sistema de archivos: disco completo, partición y volumen lógico. Esta operación es transparente para las aplicaciones, que siguen pudiendo leer y escribir normalmente los objetos del sistema de archivos, pero estos están almacenados bajo una forma cifrada.

Este mecanismo puede utilizarse en todos los sistemas de archivos, excepto en el que contiene el núcleo Linux que tendrá que cargarse.

El comando `cryptsetup` permite cifrar un espacio de disco. Tenga precaución, el contenido presente en el disco se perderá.

Sintaxis

```
cryptsetup luksFormat [ -opciones ] ArchivoEspecial
```

Las opciones permiten especificar los diferentes algoritmos que se utilizarán para el cifrado (`-c`) y el hash (`-h`). El argumento `ArchivoEspecial` indica el espacio de almacenamiento que se cifrará. No tiene que estar montado.

El comando solicita una frase de contraseña que será asociada al cifrado del espacio de almacenamiento. Esta frase de contraseña tendrá que ser tecleada para poder integrar este espacio cifrado en la arborescencia global del sistema de archivos de Linux. Por lo tanto, incluso si el disco que contiene este espacio de almacenamiento se instala en otra máquina, el administrador no podrá acceder a estos datos si no conoce la frase de contraseña.

Ejemplo

En una distribución Debian, cifrado de una partición en un pendrive USB:

Instalación del paquete `cryptsetup`, si fuera necesario:

```
apt-get install cryptsetup
[...]
```

Cifrado de la partición, con las opciones por defecto:

```
cryptsetup luksFormat /dev/sdb1
AVISO: El dispositivo /dev/sdb1 ya contiene una firma de superbloque 'btrfs'.

WARNING!
========
Esto sobreescribirá los datos en /dev/sdb1 de forma irrevocable.

Are you sure? (Type uppercase yes): YES
Introduzca la frase contraseña de /dev/sdb1: XXXXXXXX
Verifique la frase contraseña: XXXXXXXX
```

La partición `/dev/sdb1` ha sido cifrada con el formato crypto_LUKS:

```
blkid /dev/sdb1
/dev/sdb1: UUID="e6300776-0429-4803-966c-25e8e3dce7ed" TYPE="crypto_LUKS"
PARTUUID="c3072e18-01"
```

d. Uso de un espacio de almacenamiento cifrado

Una vez cifrado, el espacio de almacenamiento no puede utilizarse directamente. Primero hay que usar el comando `cryptsetup` con el subcomando `luksOpen`, para crear un archivo especial de bloques particular (gestionado por el device mapper) el cuál será asociado al espacio de almacenamiento y se podrá manipular si tener que teclear la frase de contraseña.

El comando inverso, `cryptsetup` con el subcomando `luksClose`, tiene que

ejecutarse después de desmontar el sistema de archivos. Este comando cierra el archivo especial device mapper, haciendo que la partición cifrada ya no sea accesible.

Creación del archivo especial asociado a la partición cifrada

```
cryptsetup luksOpen ArchivoEspecialOrig Nombre
```

El comando asocia el espacio de almacenamiento cifrado `ArchivoEspecialOrig` a un archivo especial llamado `Nombre`, gestionado por el *device mapper* y creado en el directorio `/dev/mapper`. Para realizar esta creación, tendrá que indicar la frase de contraseña asociada al espacio de almacenamiento cifrado.

Una vez creado, el archivo especial puede ser utilizado como si se tratara de un espacio de almacenamiento no cifrado.

Si el archivo especial device mapper ya existe, el comando "lo abre" para que se pueda acceder a los datos sin cifrar.

Ejemplo

En la partición del pendrive USB cifrado en el ejemplo anterior, se creará un sistema de archivos ext4 y se montará en la arborescencia:

Antes que nada, se intenta montar directamente la partición cifrada:

```
mount /dev/sdb1 /mnt
mount: /mnt: tipo de sistema de ficheros 'crypto_LUKS' desconocido.
```

El montaje ha fallado porque la partición está cifrada.

Primero se tiene que crear el archivo especial device mapper:

```
cryptsetup luksOpen /dev/sdb1 USBSecreta
Introduzca la frase contraseña de /dev/sdb1: XXXXXXXX
```

Ahora hay que crear el sistema de archivos ext4 que será montado en la arborescencia.

```
mkfs -t ext4 /dev/mapper/USBSecreta
mke2fs 1.44.5 (15-Dec-2018)
Creating filesystem with 1971986 4k blocks and 493856 inodes
Filesystem UUID: 19724471-610b-4063-85e3-6ef8d691befd
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632
```

El sistema de archivos ha sido creado:

```
blkid /dev/mapper/USBSecreta
/dev/mapper/USBSecreta: UUID="19724471-610b-4063-85e3-6ef8d691befd" TYPE="ext4"
```

Podemos montarlo y usarlo normalmente; el cifrado de los bloques del espacio de disco es transparente para las aplicaciones.

```
mount /dev/mapper/USBSecreta /mnt
```

Una vez montado, el sistema de archivos es accesible en lectura y escritura, de manera transparente:

```
cp /etc/host* /mnt
cd /mnt cat hosts
127.0.0.1       localhost
192.168.0.39    debian10

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
```

```
ff02::2 ip6-allrouters
echo "mensaje no cifrado" > archivo
```

e. Cifrado de un sistema de archivos

Otra posibilidad es la de gestionar el cifrado en el interior de un sistema de archivos.

El principio es montar en un directorio del sistema de archivos un tipo de sistema de archivos específico, ecryptfs, gestionado por el núcleo, almacenado en un directorio cifrado. Los datos, cifrados, se encuentran físicamente en el directorio cifrado, pero son accesibles de manera transparente para las aplicaciones, a partir del directorio de montaje.

Para poder efectuar el montaje y poder acceder a los datos descifrados, hay que introducir la frase de contraseña asociada inicialmente al directorio cifrado.

Montaje del directorio ecryptfs

```
mount -t ecryptfs  [ -o Opciones ] DirCrypt PuntoMontaje
```

El comando monta el directorio cifrado `DirCrypt` en el directorio `PuntoMontaje`. Sin opciones, el comando solicita una frase de contraseña asociada al directorio cifrado. Esta frase se puede usar como argumento del comando con la opción `-o passphrase_passwd="XXXXX"`. Esto, sin embargo, está desaconsejado por motivos de seguridad. También se puede encontrar en un archivo especificado en el comando con la opción `-o passphrase_passwd_file=CaminoArchivo`.

A menudo, se especifica el mismo directorio para los dos argumentos `DirCrypt` y `PuntoMontaje`, el contenido del directorio es automáticamente descifrado cuando una aplicación necesita acceder a él.

Por ahora, el paquete `ecryptfs-utils` no se proporciona con la versión 10 (Buster) de la distribución Debian ni con la versión 8 de la distribución CentOS, esto está provocado por un fallo de seguridad. El ejemplo que se ha presentado utiliza una versión de distribución más antigua.

Ejemplo

Se crean dos directorios:

```
mkdir /root/datos /root/.datos
```

Se efectúa el montaje cifrado del directorio cifrado `/root/.datos` *en el directorio no cifrado* `/root/datos` *:*

```
mount -t ecryptfs /root/.datos /root/datos
Passphrase:  XXXXXX
Select cipher:
1) aes: blocksize = 16; min keysize = 16; max keysize = 32 (not loaded)
2) blowfish: blocksize = 16; min keysize = 16; max keysize = 56 (not loaded)
3) des3_ede: blocksize = 8; min keysize = 24; max keysize = 24 (not loaded)
4) twofish: blocksize = 16; min keysize = 16; max keysize = 32 (not loaded)
5) cast6: blocksize = 16; min keysize = 16; max keysize = 32 (not loaded)
6) cast5: blocksize = 8; min keysize = 5; max keysize = 16 (not loaded)
Selection [aes]:
Select key bytes:
1) 16
2) 32
3) 24
Selection [16]:
Enable plaintext passthrough (y/n) [n]:
Enable filename encryption (y/n) [n]:
Attempting to mount with the following options:
 ecryptfs_unlink_sigs
 ecryptfs_key_bytes=16
```

```
 ecryptfs_cipher=aes
 ecryptfs_sig=9033219342b4df62
Mounted eCryptfs
```

Se usa el sistema de archivos cifrado, a través del directorio de montaje:

```
cd /root/datos
echo "Un secreto importante" > cajafuerte
cat cajafuerte
Un secreto importante
```

Se desmonta el sistema de archivos:

```
cd
umount /root/datos
ls /root/.datos
cajafuerte
file /root/.data/cajafuerte
archivo: data
cat /root/.datos/cajafuerte
??b??b?:####?##888"3DUfw`Ln}!??J#$O?eH9?]?#_ZZZ?3!?BmmB5
[...]
```

El contenido del archivo está cifrado.