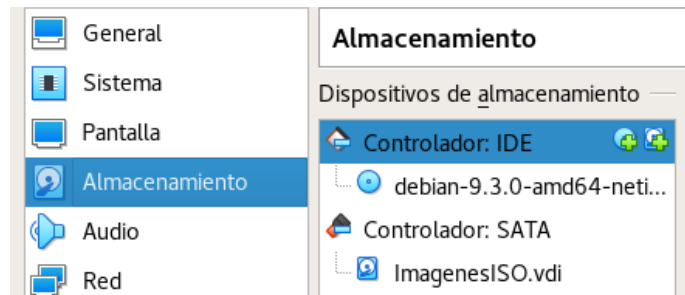


TRABAJO CON IMÁGENES ISO



1) Crea una imagen ISO a partir de un CD. Monta dicha imagen y comprueba su contenido.

En Virtualbox accedemos a **Configuración** → **Almacenamiento**, el **Controlador: IDE** estará vacío y le cargaremos la ISO de debian para que nos lo reconozca como un CD-ROM.



El siguiente paso en nuestra máquinas nos aparecerá el dispositivo **CD-ROM(sr0)** con el nombre de la imagen que hemos cargado.

```
imageniso@imageniso:~$ lsblk -f
NAME      FSTYPE LABEL                                UUID                                MOUNTPOINT
sda
├─sda1    ext4
├─sda2
├─sda5    swap
└─sr0     iso9660 Debian 9.3.0 amd64 n 2017-12-09-12-12-33-00
```

Para crear una imagen ISO a partir de un cd he ejecutado el siguiente comando:

#dd if=/dev/sr0 of=Debian.iso y luego comprobamos que la imagen está creada con un **#ls -l**.

```
imageniso@imageniso:~$ dd if=/dev/sr0 of=Debian.iso
593920+0 registros leídos
593920+0 registros escritos
304087040 bytes (304 MB, 290 MiB) copied, 3,24385 s, 93,7 MB/s
imageniso@imageniso:~$ ls -l
total 296964
-rw-r--r-- 1 imageniso imageniso 304087040 may 29 11:58 Debian.iso
```

Luego he creado un punto de montaje con **#mkdir /mnt/debianiso** para montar posteriormente la imagen y poder comprobar el contenido.

```
root@imageniso:/home/imageniso# mkdir /mnt/debianiso
```

El siguiente paso ha sido montar la imagen en el punto de montaje que he creado usando **#mount Debian.iso /mnt/debianiso/**.

```
root@imageniso:/home/imageniso# mount Debian.iso /mnt/debianiso/
mount: /dev/loop0 está protegido contra escritura; se monta como sólo lectura
```

Comprobamos que realmente se ha montado la imagen ISO en el punto de montaje creado.

```
NAME      FSTYPE LABEL                                UUID                                MOUNTPOINT
loop0     iso9660 Debian 9.3.0 amd64 n 2017-12-09-12-12-33-00 /mnt/debianiso
sda
├─sda1    ext4
├─sda2
├─sda5    swap
└─sr0     iso9660 Debian 9.3.0 amd64 n 2017-12-09-12-12-33-00
```

Y por último accedemos a la imagen con `#cd /mnt/debianiso` y hacemos un `#ls -l` para comprobar el contenido.

```
root@imageniso:/home/imageniso# cd /mnt/debianiso/
root@imageniso:/mnt/debianiso# ls -l
total 934
-r--r--r-- 1 root root    146 dic  9 2017 autorun.inf
dr-xr-xr-x 1 root root   2048 dic  9 2017 boot
dr-xr-xr-x 1 root root   2048 dic  9 2017 css
lr-xr-xr-x 1 root root      1 dic  9 2017 debian -> .
dr-xr-xr-x 1 root root   2048 dic  9 2017 dists
dr-xr-xr-x 1 root root   2048 dic  9 2017 doc
dr-xr-xr-x 1 root root   2048 dic  9 2017 efi
dr-xr-xr-x 1 root root   2048 dic  9 2017 firmware
-r--r--r-- 1 root root 183992 dic  5 2017 g2ldr
-r--r--r-- 1 root root   8192 dic  5 2017 g2ldr.mbr
dr-xr-xr-x 1 root root   2048 dic  9 2017 install
dr-xr-xr-x 1 root root   2048 dic  9 2017 install.amd
dr-xr-xr-x 1 root root   4096 dic  9 2017 isolinux
-r--r--r-- 1 root root   84780 dic  9 2017 md5sum.txt
dr-xr-xr-x 1 root root   4096 dic  9 2017 pics
dr-xr-xr-x 1 root root   2048 dic  9 2017 pool
-r--r--r-- 1 root root   8473 dic  9 2017 README.html
-r--r--r-- 1 root root    291 mar  4 2017 README.mirrors.html
-r--r--r-- 1 root root     86 mar  4 2017 README.mirrors.txt
-r--r--r-- 1 root root    473 dic  9 2017 README.source
-r--r--r-- 1 root root   5181 dic  9 2017 README.txt
-r--r--r-- 1 root root  632347 dic  5 2017 setup.exe
dr-xr-xr-x 1 root root   2048 dic  9 2017 tools
-r--r--r-- 1 root root    233 dic  9 2017 win32-loader.ini
```

2) Crea una imagen ISO a partir de un directorio que contenga una copia del directorio /etc (y todos los directorios contenidos). Monta dicha imagen y comprueba que contiene toda la información. ¿Es necesario utilizar opciones especiales en la creación de la imagen?

Para hacer este segundo ejercicio he tenido que instalar el **paquete genisoimage**.

`#apt-get install genisoimage`.

```
root@imageniso:/# apt-get install genisoimage
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Paquetes sugeridos:
  wodim cdrkit-doc
Se instalarán los siguientes paquetes NUEVOS:
  genisoimage
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 1 no actualizados.
Se necesita descargar 376 kB de archivos.
Se utilizarán 1.655 kB de espacio de disco adicional después de esta operación.
Des:1 http://ftp.es.debian.org/debian stretch/main amd64 genisoimage amd64 9:1.1.11-3+b2 [376 kB]
Descargados 376 kB en 0s (1.471 kB/s)
Seleccionando el paquete genisoimage previamente no seleccionado.
(Leyendo la base de datos ... 28671 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../genisoimage_9%3a1.1.11-3+b2_amd64.deb ...
Desempaquetando genisoimage (9:1.1.11-3+b2) ...
Configurando genisoimage (9:1.1.11-3+b2) ...
Procesando disparadores para man-db (2.7.6.1-2) ...
```

El siguiente paso ha sido crear la **imagen ISO** del directorio /etc con el siguiente comando.

`#genisoimage -r -J -o etc.iso /etc`.

```
root@imageniso:/# genisoimage -r -J -o etc.iso /etc
```

luego he comprobado que se ha creado correctamente con `#ls -l`.

```
root@imageniso:/# ls -l
total 2920
drwxr-xr-x  2 root root    4096 may 29 10:57 bin
drwxr-xr-x  3 root root    4096 may 29 10:58 boot
drwxr-xr-x 77 root root    4096 may 29 12:26 copiaetc
drwxr-xr-x 18 root root    3160 may 29 12:00 dev
drwxr-xr-x 77 root root    4096 may 29 11:56 etc
-rw-r--r--  1 root root 2906112 may 29 12:34 etc.iso
```

He creado el punto de montaje `#mkdir /mnt/etc_iso`.

```
root@imageniso:/# mkdir /mnt/etc_iso
```

Y lo he montado para ver el contenido usando `#mount etc.iso /mnt/etc_iso`.

```
root@imageniso:/# mount etc.iso /mnt/etc_iso/
mount: /dev/loop0 está protegido contra escritura; se monta como sólo lectura
root@imageniso:/# lsblk -f
NAME        FSTYPE LABEL                UUID                                MOUNTPOINT
loop0       iso9660 CDROM                2019-05-29-12-34-26-00            /mnt/etc_iso
sda
├─sda1      ext4                     588d9723-b427-4cad-bc9d-26d0088c036f /
├─sda2
└─sda5      swap                    9d879867-2c7e-4f32-acdf-6e8e528e3c9b [SWAP]
sr0         iso9660 Debian 9.3.0 amd64 n 2017-12-09-12-12-33-00
```

Y luego he comprobado que tiene el mismo contenido que `/etc`. para ello he accedido al punto de montaje `#cd /mnt/etc_iso` y por último un `#ls -l` para comprobar el contenido.

```
-r-xr-xr-x 1 root root    268 may 29 12:26 rmt
-r--r--r-- 1 root root    887 may 29 12:26 rpc
-r--r--r-- 1 root root   1963 may 29 12:26 rsyslog.conf
dr-xr-xr-x 2 root root   2048 may 29 12:26 rsyslog.d
-r--r--r-- 1 root root   4179 may 29 12:26 securetty
dr-xr-xr-x 4 root root   2048 may 29 12:26 security
dr-xr-xr-x 2 root root   2048 may 29 12:26 selinux
-r--r--r-- 1 root root  19183 may 29 12:26 services
dr-xr-xr-x 2 root root   2048 may 29 12:26 sgml
-r--r--r-- 1 root root    970 may 29 12:26 shadow
-r--r--r-- 1 root root    873 may 29 12:26 shadow-
-r--r--r-- 1 root root     73 may 29 12:26 shells
dr-xr-xr-x 2 root root   2048 may 29 12:26 skel
dr-xr-xr-x 2 root root   2048 may 29 12:26 ssh
dr-xr-xr-x 4 root root   2048 may 29 12:26 ssl
-r--r--r-- 1 root root    771 may 29 12:26 staff-group-for-usr-local
-r--r--r-- 1 root root     23 may 29 12:26 subgid
-r--r--r-- 1 root root     0 may 29 12:26 subgid-
-r--r--r-- 1 root root     23 may 29 12:26 subuid
-r--r--r-- 1 root root     0 may 29 12:26 subuid-
-r--r--r-- 1 root root   2683 may 29 12:26 sysctl.conf
dr-xr-xr-x 2 root root   2048 may 29 12:26 sysctl.d
dr-xr-xr-x 5 root root   2048 may 29 12:26 systemd
dr-xr-xr-x 2 root root   2048 may 29 12:26 terminfo
-r--r--r-- 1 root root     14 may 29 12:26 timezone
dr-xr-xr-x 2 root root   2048 may 29 12:26 tmpfiles.d
-r--r--r-- 1 root root   1260 may 29 12:26 ucf.conf
dr-xr-xr-x 4 root root   2048 may 29 12:26 udev
dr-xr-xr-x 3 root root   2048 may 29 12:26 ufw
dr-xr-xr-x 2 root root   2048 may 29 12:26 update-motd.d
dr-xr-xr-x 2 root root   2048 may 29 12:26 vim
-r--r--r-- 1 root root   4942 may 29 12:26 wgetrc
dr-xr-xr-x 2 root root   2048 may 29 12:26 wpa_supplicant
dr-xr-xr-x 4 root root   2048 may 29 12:26 X11
dr-xr-xr-x 3 root root   2048 may 29 12:26 xdg
dr-xr-xr-x 2 root root   2048 may 29 12:26 xml
```

¿Es necesario utilizar opciones especiales en la creación de la imagen?

Las opciones especiales son -r, -J, -o donde:

-r →

```
-r This is like the -R option, but file ownership and modes are set to more useful values. The uid and gid are set to zero, because they are usually only useful on the author's system, and not useful to the client. All the file read bits are set true, so that files and directories are globally readable on the client. If any execute bit is set for a file, set all of the execute bits, so that executables are globally executable on the client. If any search bit is set for a directory, set all of the search bits, so that directories are globally searchable on the client. All write bits are cleared, because the filesystem will be mounted read-only in any case. If any of the special mode bits are set, clear them, because file locks are not useful on a read-only filesystem, and set-id bits are not desirable for uid 0 or gid 0. When used on Win32, the execute bit is set on all files. This is a result of the lack of file permissions on Win32 and the Cygwin POSIX emulation layer. See also -uid, -gid, -dir-mode, -file-mode and -new-dir-mode.
```

-J →

```
-J Generate Joliet directory records in addition to regular ISO9660 filenames. This is primarily useful when the discs are to be used on Windows machines. Joliet filenames are specified in Unicode and each path component can be up to 64 Unicode characters long. Note that Joliet is not a standard - only Microsoft Windows and Linux systems can read Joliet extensions. For greater portability, consider using both Joliet and Rock Ridge extensions.
```

-o →

```
-o filename Specify the output file for the the ISO9660 filesystem image. This can be a disk file, a tape drive, or it can correspond directly to the device name of the optical disc writer. If not specified, stdout is used. Note that the output can also be a block device for a regular disk partition, in which case the ISO9660 filesystem can be mounted normally to verify that it was generated correctly.
```

3) Investiga cómo se puede crear una imagen ISO arrancable. Intenta crear una imagen ISO que permita iniciar una máquina virtual con un sistema Linux básico.