Computer Systems

UD 09. DISK IMAGES

Computer Systems
CFGS DAW

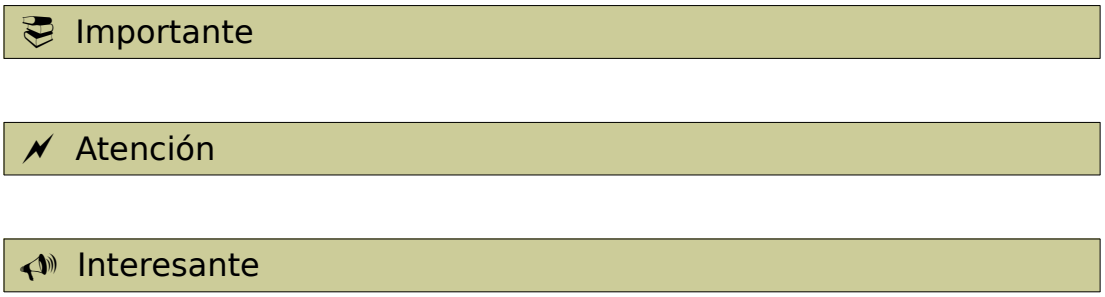
Sergio García / Alfredo Oltra
sergio.garcia@ceedcv.es
alfredo.oltra@ceedcv.es
2022/2023

Versión:220602.1647

Licencia

Nomenclatura

A lo largo de este tema se utilizarán distintos símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:



ÍNDICE DE CONTENIDO

UD09. DISK IMAGES

1. DISK IMAGES

A disk image is a file that contains the exact copy (bit by bit) of a storage system, usually a hard disk, although it can be another media such as a USB storage, a DVD, etc. Being an exact copy, not only contains all the data, also the exact structure of the device. This includes sectors that are not "visible" to the user, but which are critical to the system, such as the boot sectors or the partition table.

The generated image can be restored on other devices (obviously provided the available size is greater than or equal to the size of the original device).

Their main uses are:

- Cloning of computer with the same technical characteristics. The user sets up a new machine on all products: partitions, operating system(s), drivers, external connections, basic applications, user accounts, permissions, etc. and once done, make an image that later restores to multiple computers. Examples of use in a computer room, or departments of companies, in which generally the equipment fleet is identical.
- Recovery of a common starting point. Possibly the image created in the previous section can be used to, in case of system error, return to a known and functional point.
- Migrate from a small hard disk to a larger one.

The great problem with this technique is the space occupied by the copies. The fact of copying bit by bit implies that not only data are copy, but also another type of information that is not necessary to backup and that obviously makes the image files very large. An image of an X GB disk will occupy X GB. To minimize this problem, the programs that make images use two options: do not save unused space and use compression mechanisms.

It is very important to note that this system, although it could be confused with a backup system, is not. Neither the generation time, nor the restoration time, nor the fact that when retrieving it on the same computer all content is overwritten and that, of course, the data backed up includes elements other than the data and that it is not possible to filter that data by importance make both concepts, although similar, are not the same.

Beside to cloning an entire disk, it is possible to clone a single partition

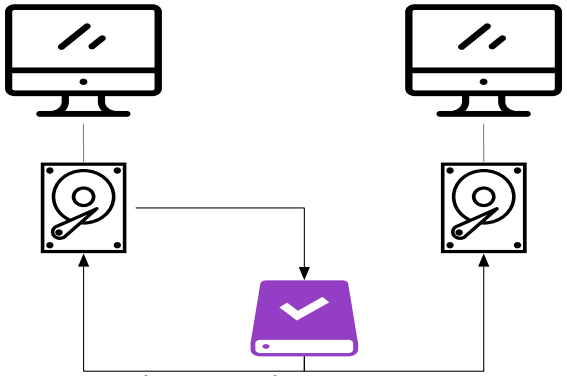


Figure 1. Clone Process

2. PROGRAMS TO MAKE/RESTORE IMAGES

The are a lot of programs to help you to create and restore disk images. Some of them can be:

- *CloneZilla*

  Clonezilla is a free software disaster recovery, disk cloning and deployment solution. Maybe the most famous and used.
- *Redo Backup and Recovery*

  Similar functionality to *CloneZilla* (in fact both of them use the same core). *Redo Backup* has a friendly graphic interface that makes very easy to use it. It is not updated from 2012.
- *Win32 Disk Imager*

  Source code tool for Windows Systems. It works only with USB sticks or SD cards.

3. DRBL

As discussed earlier, the use of images allows the creation of multiple identical computers in configuration from one.

The procedure to perform such kind of cloning is to create the image of the source computer and clone one by one all the target computers. Obviously this process is faster than installing and configuring each of them, but possibly a more optimal solution would be to be able to **simultaneously** clone all computers connecting them on the network.

For this kind of use one of the best solutions is to use *DRBL*.

DRBL is a tool (a Linux distribution) that allows to start an operating system, in the network machines, without having to have any software installed locally. With *Clonezilla* (included in *DRBL*) we will be able to clone in multicast mode, as many machines as we want, simultaneously.

3.1 Multicast

When sending any type of information over a network, there are three ways to do this

Unicast: It is defined by a single sender and a single receiver. The server has a separate connection with each client.

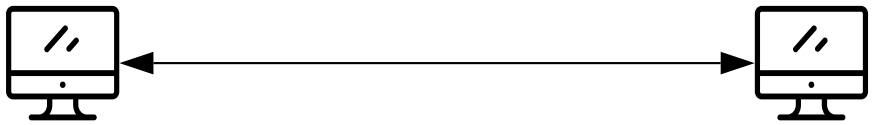


Figure 2. Unicast

Broadcast: In this transmission the signal reaches far and wide and is available to everyone

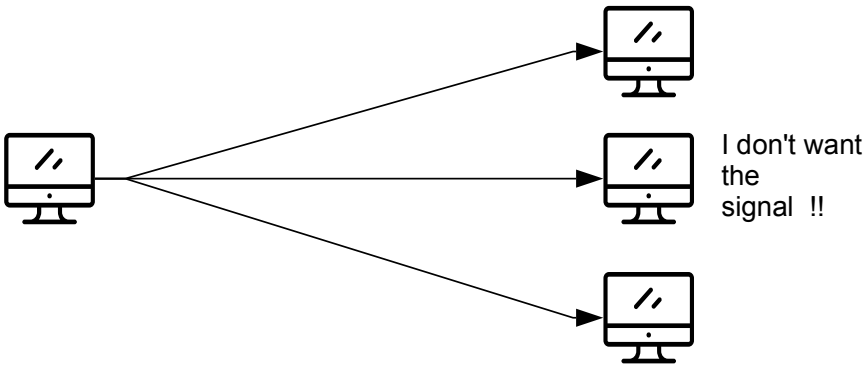


Figure 3. Broadcast

Multicast: Similar to broadcast, but in the multicast if the receiver does not want the signal it does not receive it (in broadcast all the computers in the net receive the signal

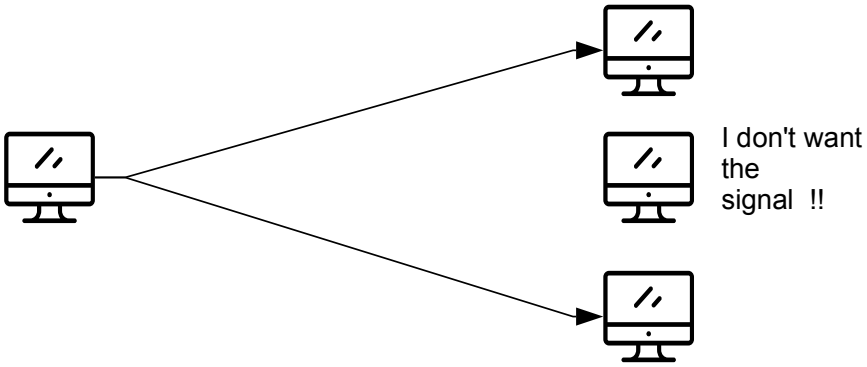


Figure 4. Multicast

3.2 Download

The first step is to get and to burn and ISO of DRBL. You can download it from http://drbl.org/download/ [1]

> DBRL is a live SO, so you run the computer with it and you can use it as a Operative System.

> There are three options amd64 (to 64 bits processors), 686 (to 32 bit processors), and 686-pae (to 32 bit processors but with support to memories larger to 3GB). You have to download the version depending on the processor of the computer to clone.

After the download, the next step is to burn the ISO image to a CD/DVD (with a burning program like Nero, Burn, Toast, ISO Burner, etc.) or in a USB stick (with UltraISO, Rufus)

3.3 Creating the disk Image

The first step is to create the image disk. We start from an installation of *LUbuntu*, where we have created a file on the desktop to verify that the image is properly performed and restored

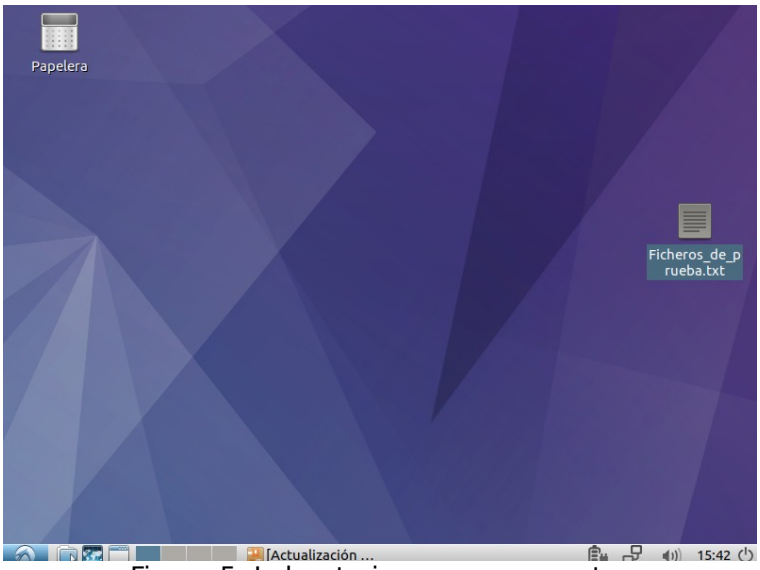


Figure 5. Lubuntu in source computer

Restart the computer with the burned DVD with DRBL in the DVD reader or if, you use a USB, with the stick connected. If we let the system start up, it will end up starting a live version of DRBL in graphic mode.

1 The main page of DBRL project is http://drbl.org/

> To boot from DVD or USB it is necessary to have this option activated and/or modify the system boot order. These options can be configured from SETUP

Figure 6. DRBL

Once the GUI has been loaded, double click *Clonezilla Live* to start the program. This program works in text mode. On the first screen (figure 7) we must choose the working mode:

- From device to device: the contents of a hard disk will be cloned in another disk
- From device to image file: the contents of a hard disk will be copied into a file. This one is the usual option

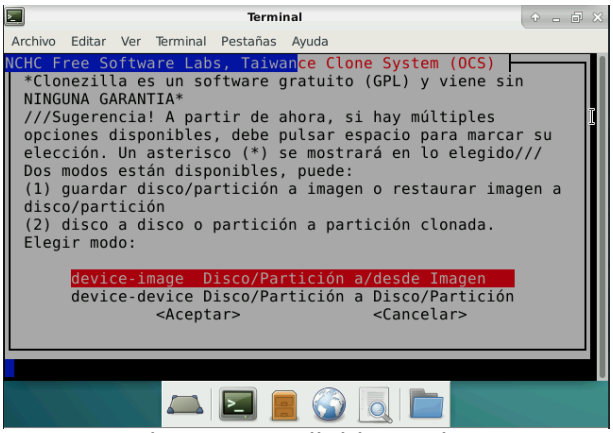


Figure 7. Available modes

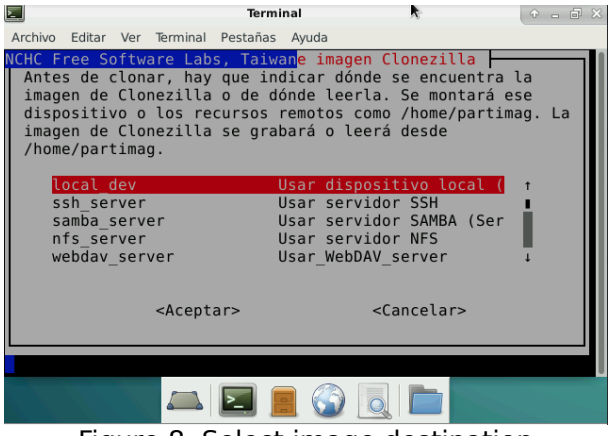


Figure 8. Select image destination

Then we will have to choose the location where we want to store the backup (figure 8). There are five options, one in local (a hard drive or USB physically connected to computer) or some kind of space available in the network, space

that can be accessed in several ways (*ssh*, *samba*, etc)[2]. In our case we are going to select a *local_dev* .

> You can not save an image of a partition (or a disk) on the same partition (or the same disk). For instance, if you are going to make an image of */dev/sda3* you can not save it to */dev/sda3*.

Our *local_dev* is an external hard disk connected by USB. So we have to connect it and wait five or ten seconds for the system can be recognize it.

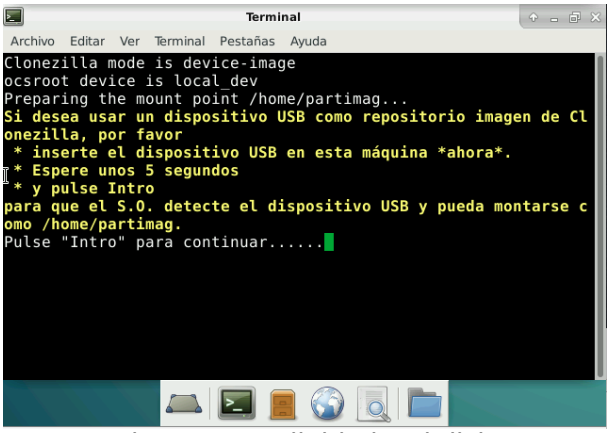


Figure 9. Available hard disk

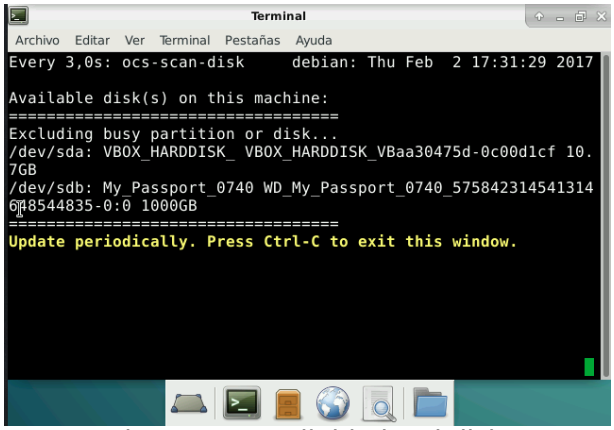


Figure 10. Available hard disk

In our case, our *local_dev* device is located as */dev/sdb, My Passport*. The next step is to mount (to connect) the *local_dev* with *CloneZilla* in a logical way. So we need to select the device */dev/sdb1*.

> In linux the devices are simulated using files that are in the *dev* directory. There are many types, but in terms of connected elements in the PATA or SATA connectors, the way to number them is: *type + order + partition*. The types are *hd* for PATA devices and *sd* for SATA devices. The order refers to the port number element in which they are connected. To do this, letters are used by the a. The partition are numbered with numbers, starting for the number 1. For instance, sdb1 refers the first partition in a device connected in the second SATA port.

Besides, *CloneZilla* allows select a directory inside that partition (figure 11). In our case, we select a *MV* folder. So, our images will save in the folder MV inside the first partition of the hard disk connected in the second SATA port.

On the next screen (figure 12), select the *Beginner* mode, and on the next one (figure 13) we need to select if we want to save a full disk or only a partition. In this case we are going to choose *savedisk*, to create an image of the full disk.

2 We will study these concepts in next units.

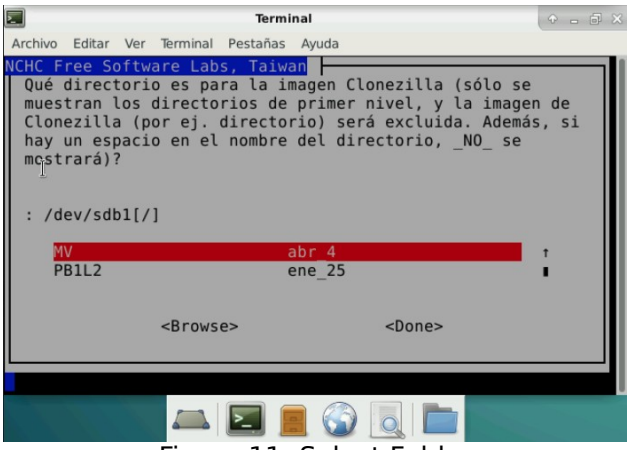


Figure 11. Select Folder

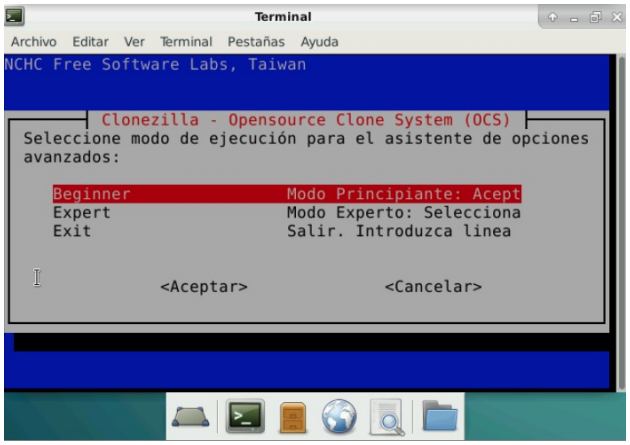


Figure 12. Use mode

In the next screens, you can modify the name of the image file (figure 14) and select the hard disk that is going to be backed up (figure 15).

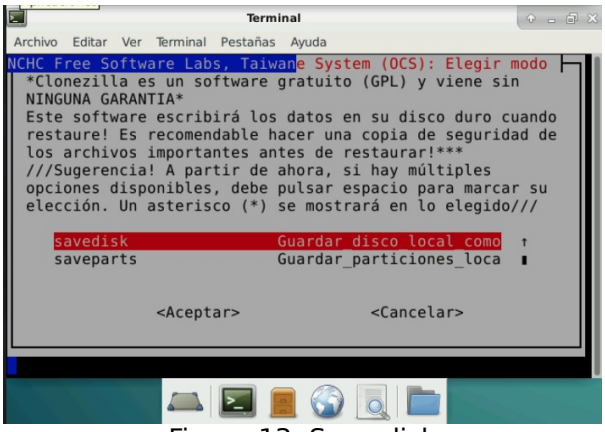


Figure 13. Save disk

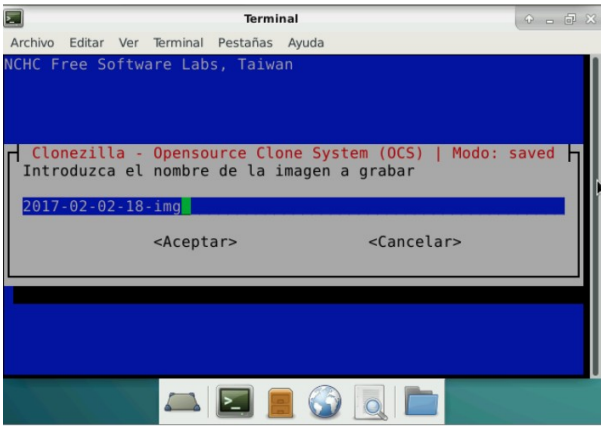


Figure 14. Image name

The last screens refer to checks, either the file system before the copy (figure 16) or the image once it has been created, and the possibility of encrypting the image. If you trust your system they can be discarded because all these processes slow down the creation of the images.

Once all the options are configured the *CloneZilla* will ask for confirmation and will start the creation of the image (figure 17).

Figure 15. Select hard disk

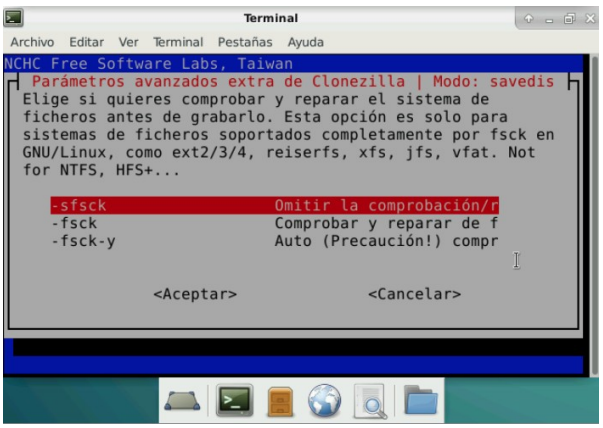


Figure 16. Checks

If everything is ok after a few minutes (figure 18) we will have on our disk a folder with our image ready to be cloned.

3.4 Restoring the image

Once we have created the image the next step is to perform the restoration.

The process can be done by starting the DRBL on the computer to clone, connecting the hard disk where the image is stored and launching the *CloneZilla live* but choosing the restore options.

However we will see how to perform *multicast* restore, so that we can clone multiple computers simultaneously.

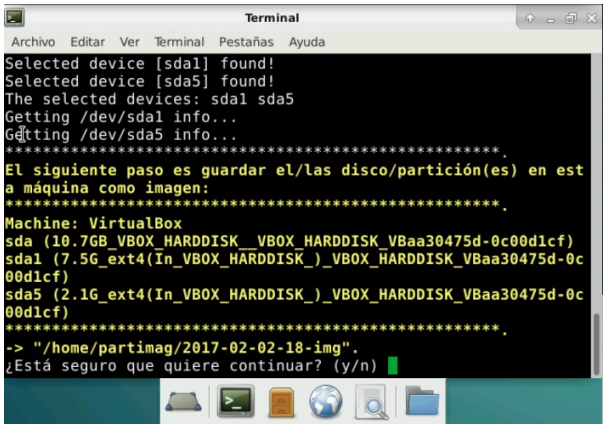


Figure 17. Are you sure?

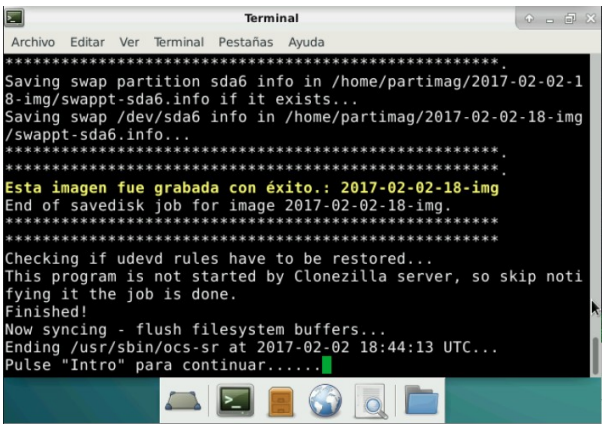


Figure 18. Image is done

We need a computer that has the server functions. In general the computer in which the image is made is usually used. We boot the computer with DRBL in the same way as to create the image, but this time we start *CloneZilla Server*.

The first time we will be shown information regarding the lack of initial configuration (figure 19), especially of another IP. By simply write Y, it automatically sets up the entire system.

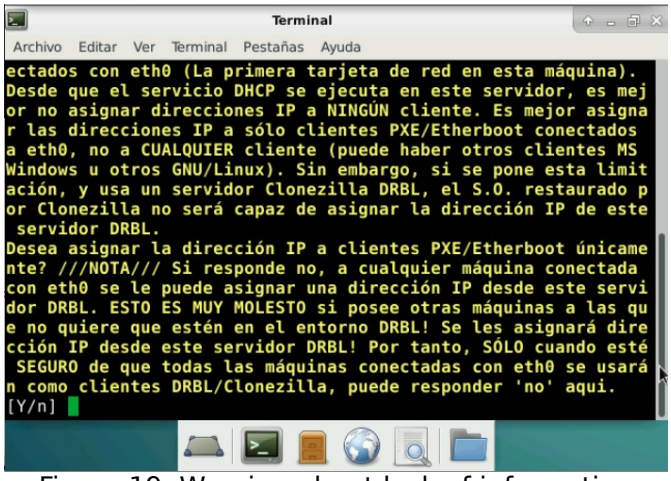


Figure 19. Warning about lack of infromation

Once the *Clonezilla* configures the server, the first step is the choice of the image, which is done in a similar way as it is done in the creation process.

We need to allocate the computers that will serve the server (figure 20), It is possible to tell you all or, if not a range of IP or MAC addresses. The most common is the first option and, to avoid problems, disconnect from the network computers that do not need to be cloned.

The next step is to tell the server that what we want is to restore (figure 21), specifying whether what we want to restore will be a partition or a disk. In our case, the image created is from a disk, so we will choose the option to restore disk.

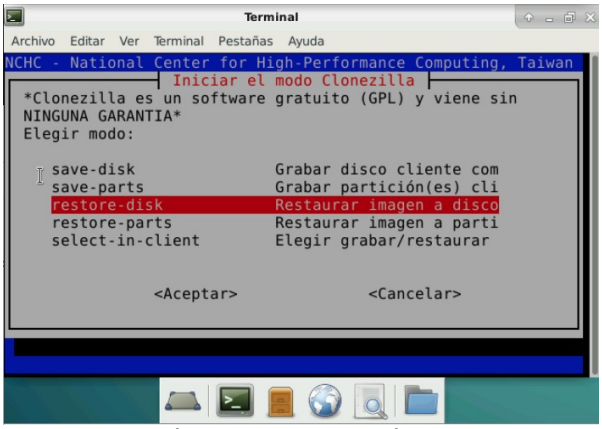


Figure 21. Operation

Of course, we have to select the name of the image (figure 22) and the name of the hard disk (figure 23) where the image will be cloned.

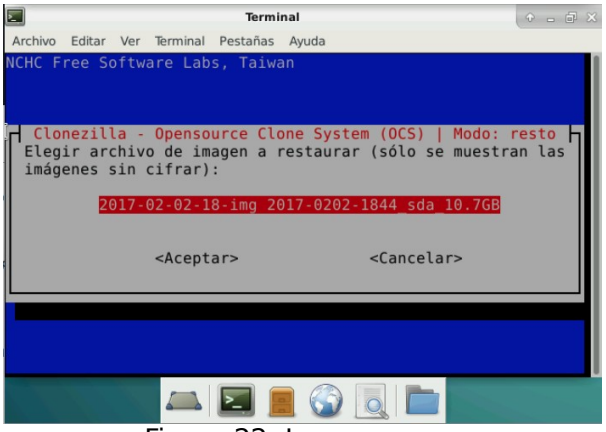


Figure 22. Image name

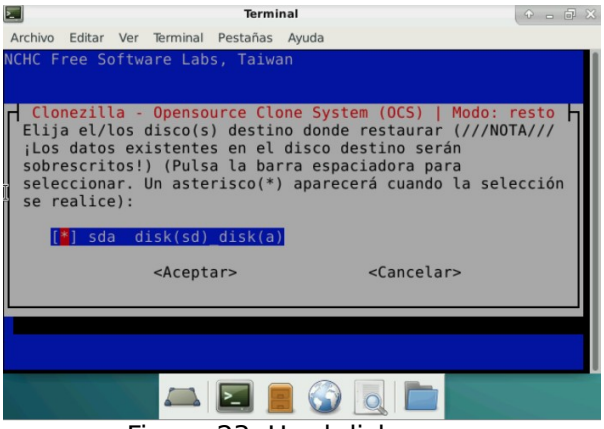


Figure 23. Hard disk name

The last steps are related to communication between teams. The first allows us to choose the mode, be it unicast, multicast or broadcast (figure 25). The second defines the way in which the server waits for clients (figure 26).

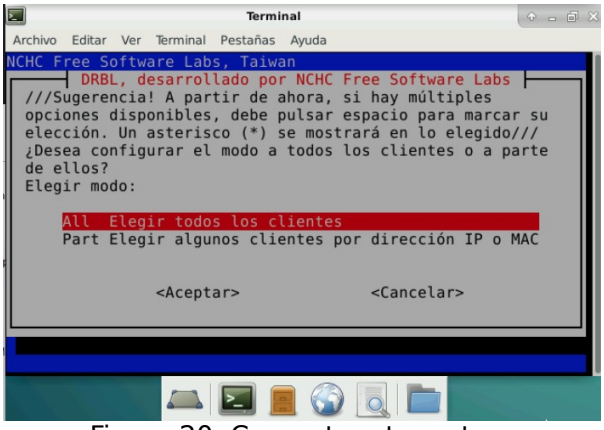


Figure 20. Computers to restore

- *Clients + timetowait*: will ask us for the number of clients that have to wait or a maximum time, whichever happens first. It is my preferred option, in case some client is locked, it even starts after a while and then fix what failed in that one.
- *Time to wait*: wait for seconds to tell you to connect all the clients you want.
- *Clients to wait*: wait for a certain number of clients before starting the process.

Once all the features are defined, the server boots waiting for clients.

It is very important not to close the window in which the server has been configured

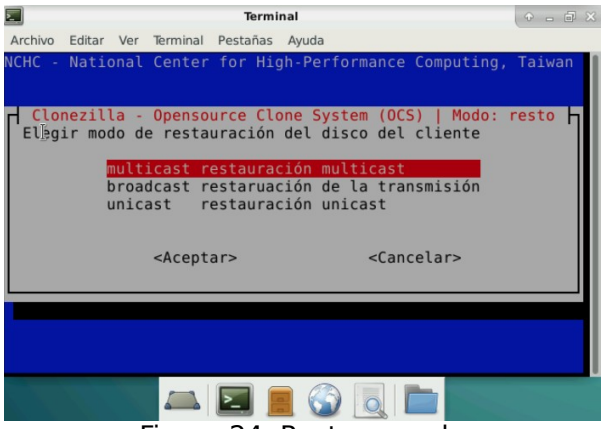


Figure 24. Restore mode

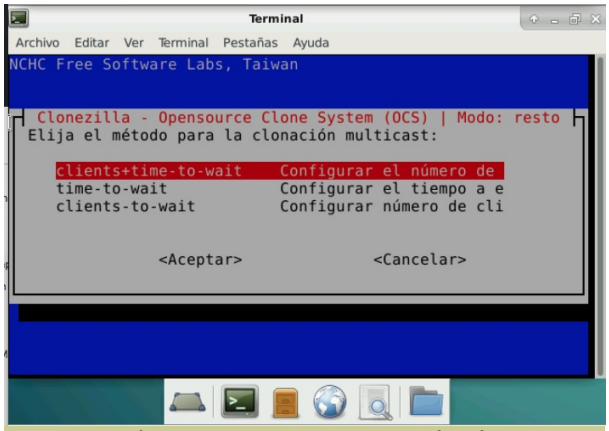


Figure 25. Restore method

3.5 Clients

If the clients have enabled in the BIOS network boot, all you have to do is turn on the computer and the whole process will start automatically. In case you do not support this technique, you will have to start each one with the DRBL CD, but entering the option *Network boot via iPXE* (figure 26)

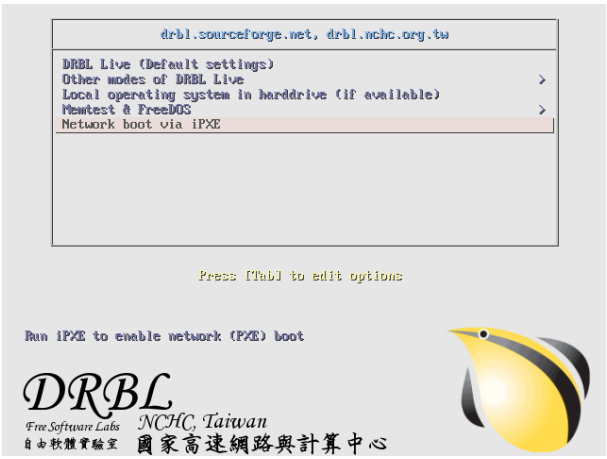


Figure 26. Clients boot with DRBL

4. BIBLIOGRAPHY

[1] Sistemas Informáticos. Isabel Mª Jimenez Cumbreras. Garceta. 2012