

Ingeniería de Servidores. PRACTICA 1

Daniel Monjas Miguélez

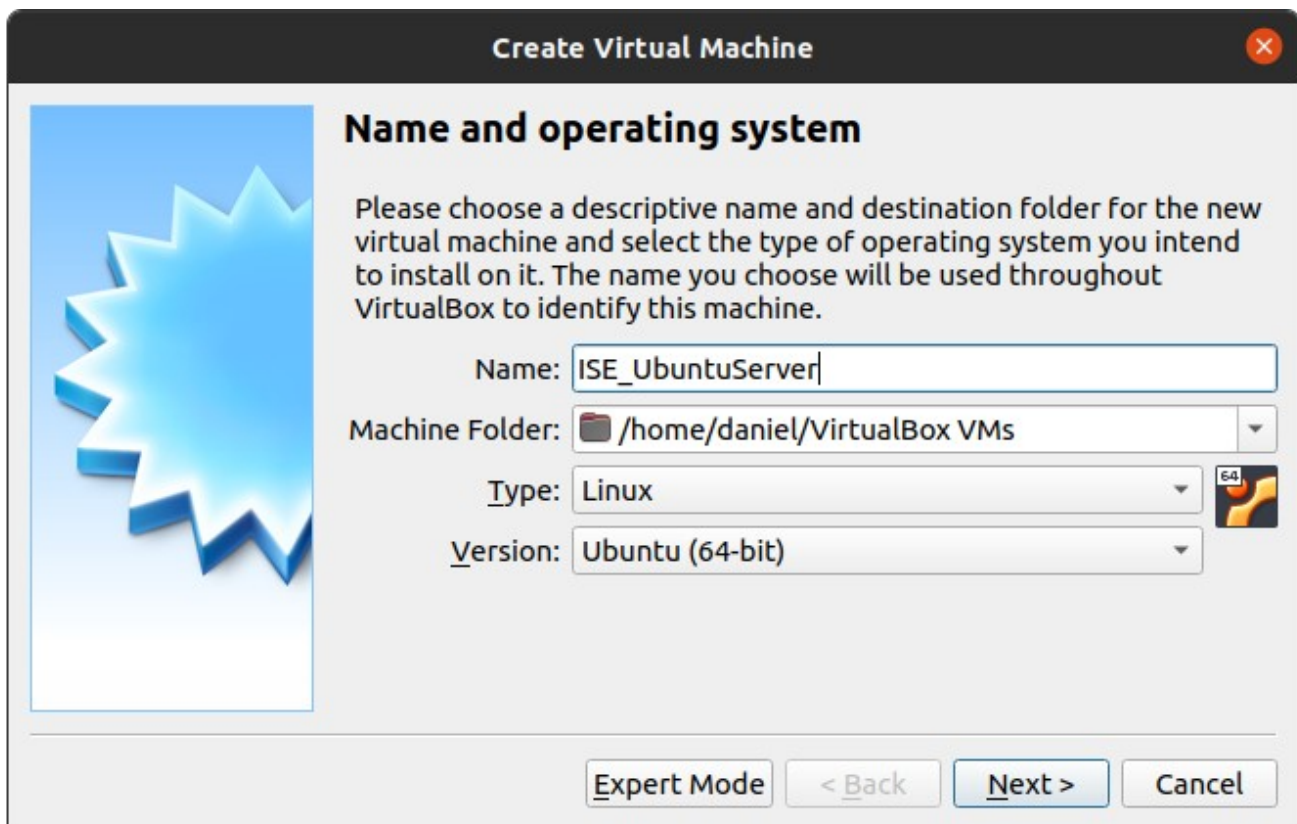
Se adjuntan los links desde los que descargar las imágenes (.iso) de Linux Server y CentOS. ([Ubuntu Server](#) y [CentOS](#)). Para esta práctica se ha utilizado VirtualBox para las máquinas virtuales.

Lección 1.

Usted está trabajando en una empresa proveedora de servicios y recibe la solicitud de un cliente que sea tener un servidor para la implantación de un comercio electrónico mediante un CMS. Sin tener más detalles por parte del cliente, le pregunta a un compañero qué configuración se suele aplicar en estos casos. Este le remite a su jefa de Dpto. que le recomienda la configuración de un RAID1 gestionado con LVM, cifrando toda la información para cumplir con la legislación vigente. También le recomienda crear al menos 3 VL (hogar, raíz y swap) y una partición para el arranque.

Nota: El usuario que crearemos será las dos primeras letras de su primera apellido, las dos segundas letras de su segundo apellido y la primera letra de su nombre. Por ejemplo, Marina Díaz Díaz → didim

Creación de la máquina virtual.



Create Virtual Machine

Name and operating system

Please choose a descriptive name and destination folder for the new virtual machine and select the type of operating system you intend to install on it. The name you choose will be used throughout VirtualBox to identify this machine.

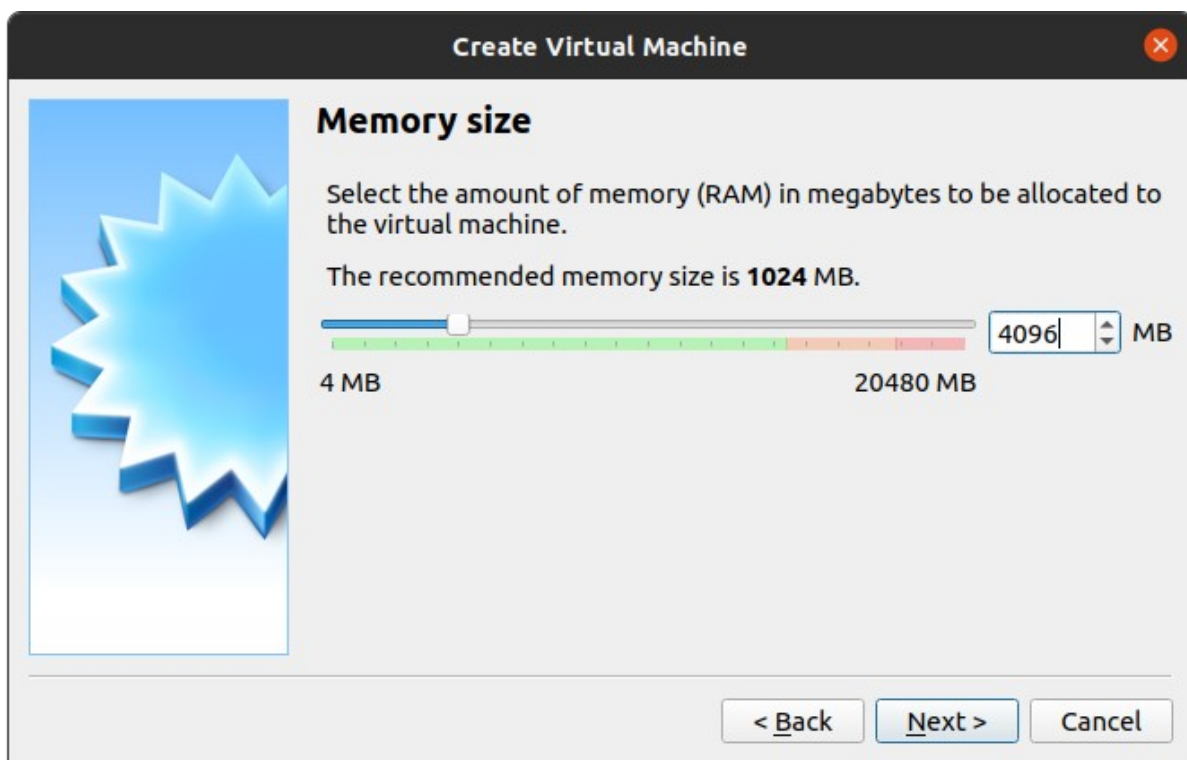
Name: ISE_UbuntuServer

Machine Folder: /home/daniel/VirtualBox VMs

Type: Linux

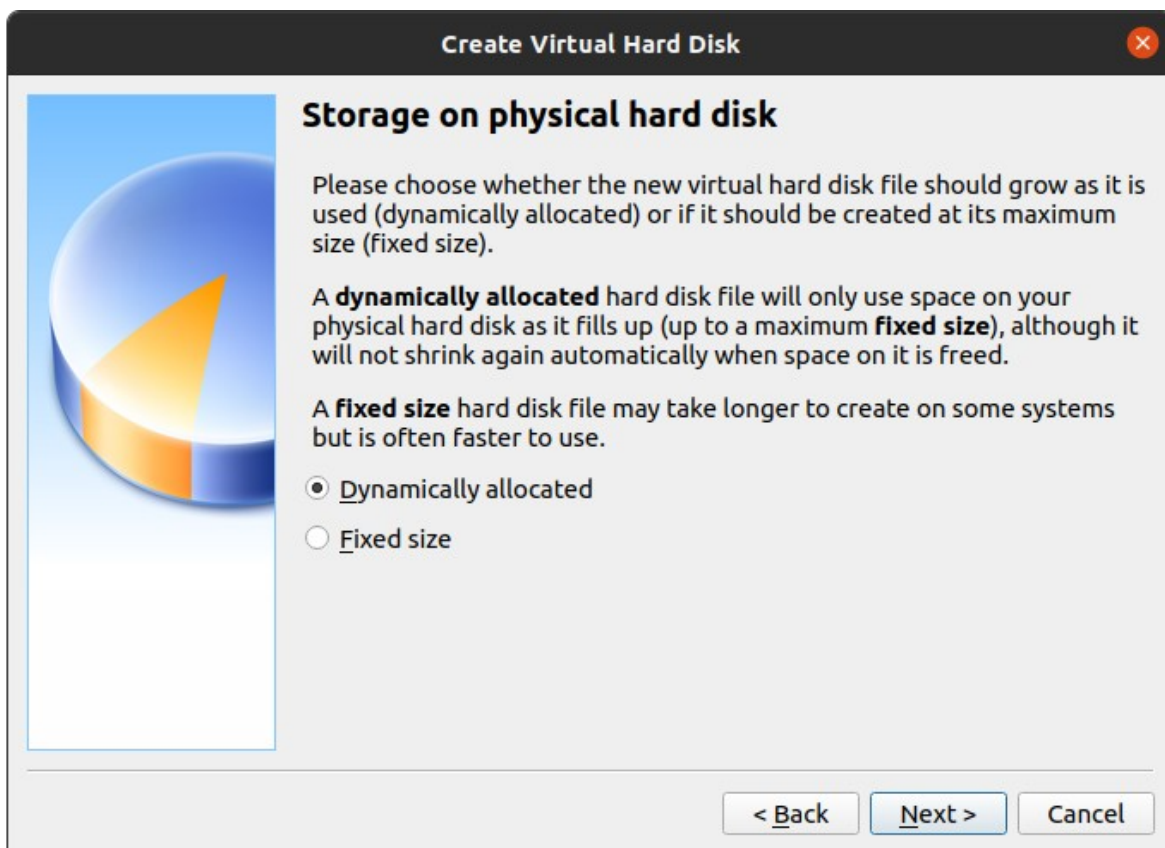
Version: Ubuntu (64-bit)

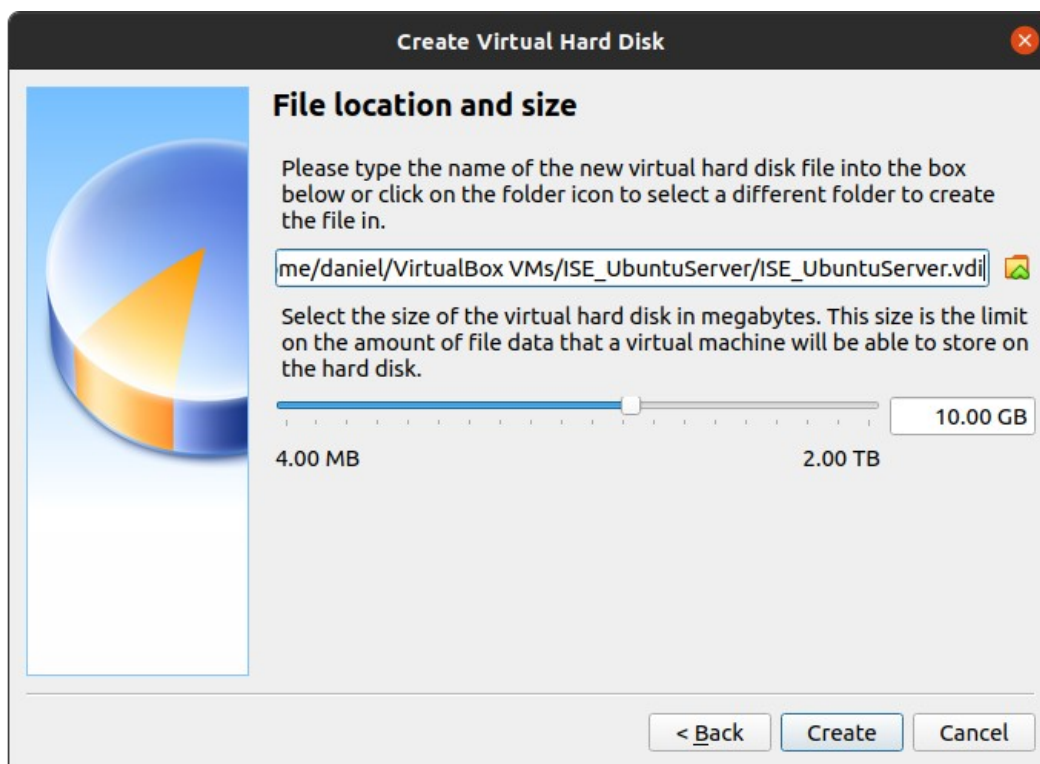
Expert Mode < Back Next > Cancel



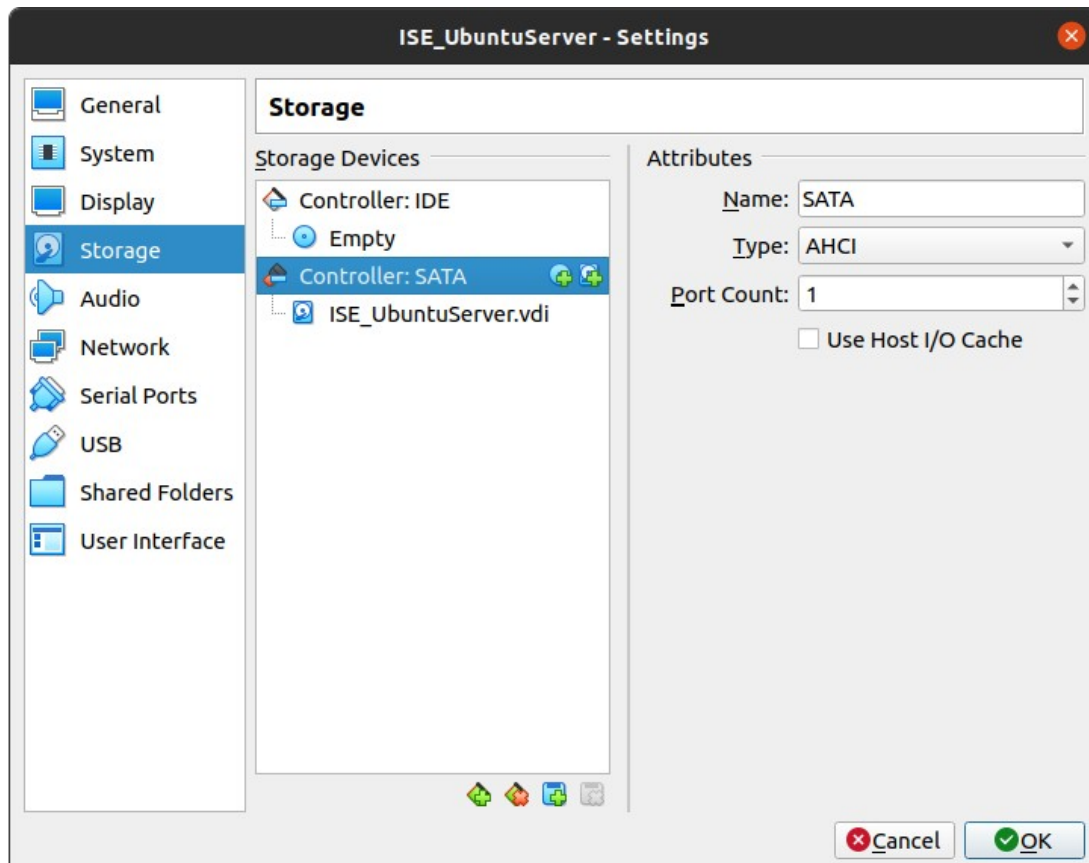
Cuando se asigne la memoria con 1024MB es suficiente, en este caso se han seleccionado 4GB porque se dispone de bastante memoria. En cualquier caso, lo recomendable es que el indicador quede dentro de la zona verde con una cierta holgura.



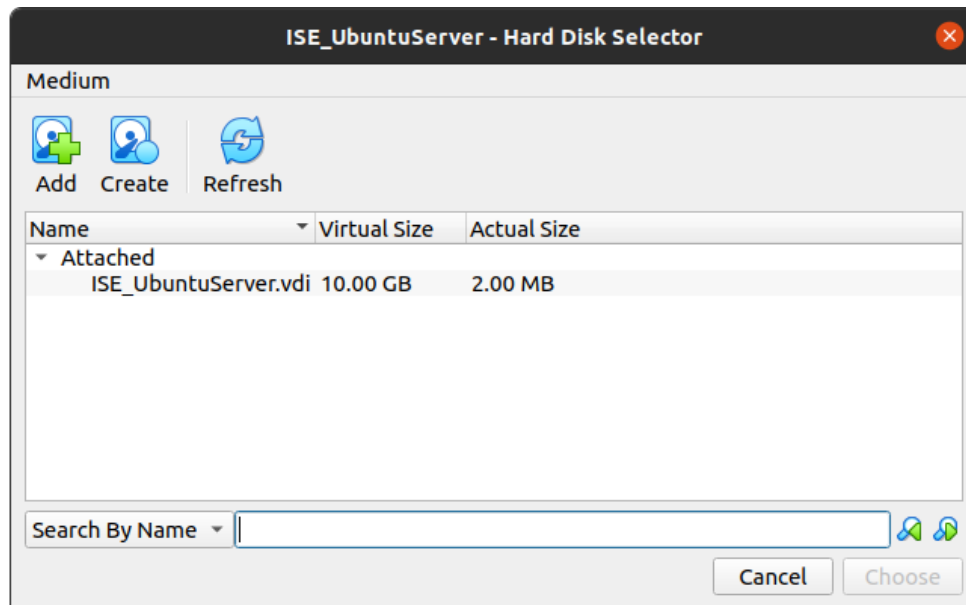




Aunque 10GB parezca excesivo al haber seleccionado “*dinamicamente allocated*” de normal esta memoria solo ocupará una fracción de ese espacio.

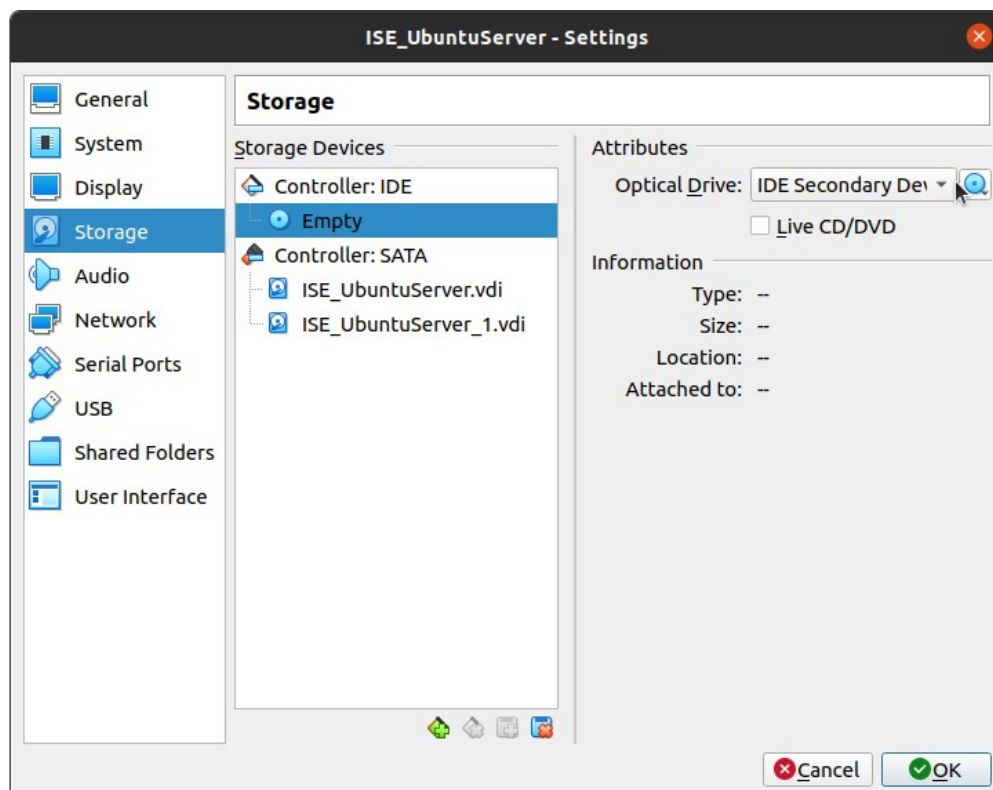


Como se quiere crear un sistema RAID1, requeriremos de dos discos duros, luego pinchamos sobre la máquina y vamos a ajustes. Ahí nos dirigimos a Storage/Almacenamiento y pulsamos en la imagen que es un disco duro con un +, con lo que aparecerá la siguiente imagen.



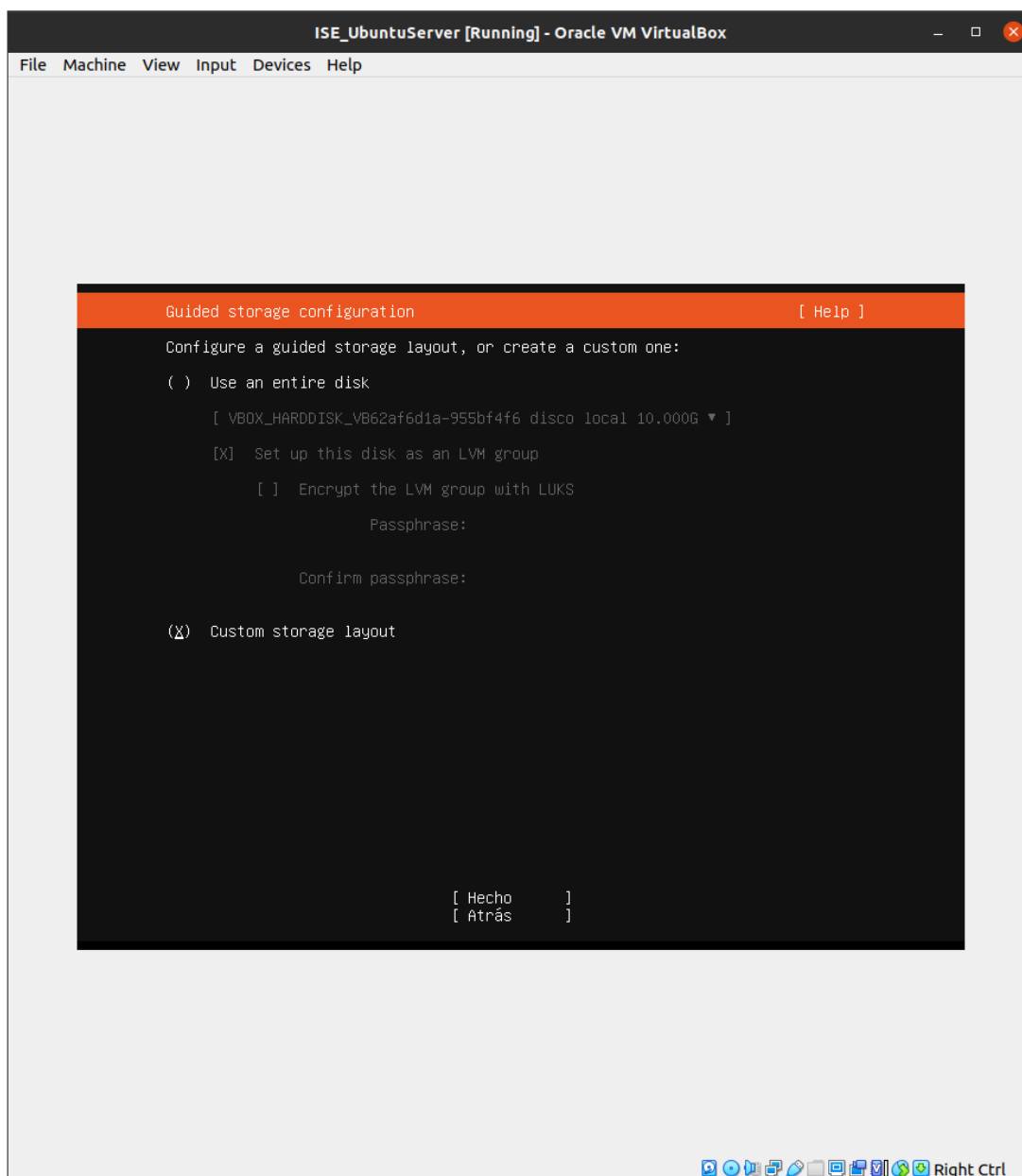
Le pulsamos a crear y seguimos los pasos vistos más arriba para la creación de discos duros. Con esto tendríamos ya los dos discos duros requeridos.

Ahora debemos instalar el sistema, para lo que nos dirigimos de nuevo a la pestaña de almacenamiento dentro de la configuración de la máquina. Esta vez pinchamos sobre “empty/vacío”, justo debajo de “Controller IDE”. Y a la derecha nos aparecen unos atributos y pulsamos sobre el siguiente icono (marcado con el ratón).

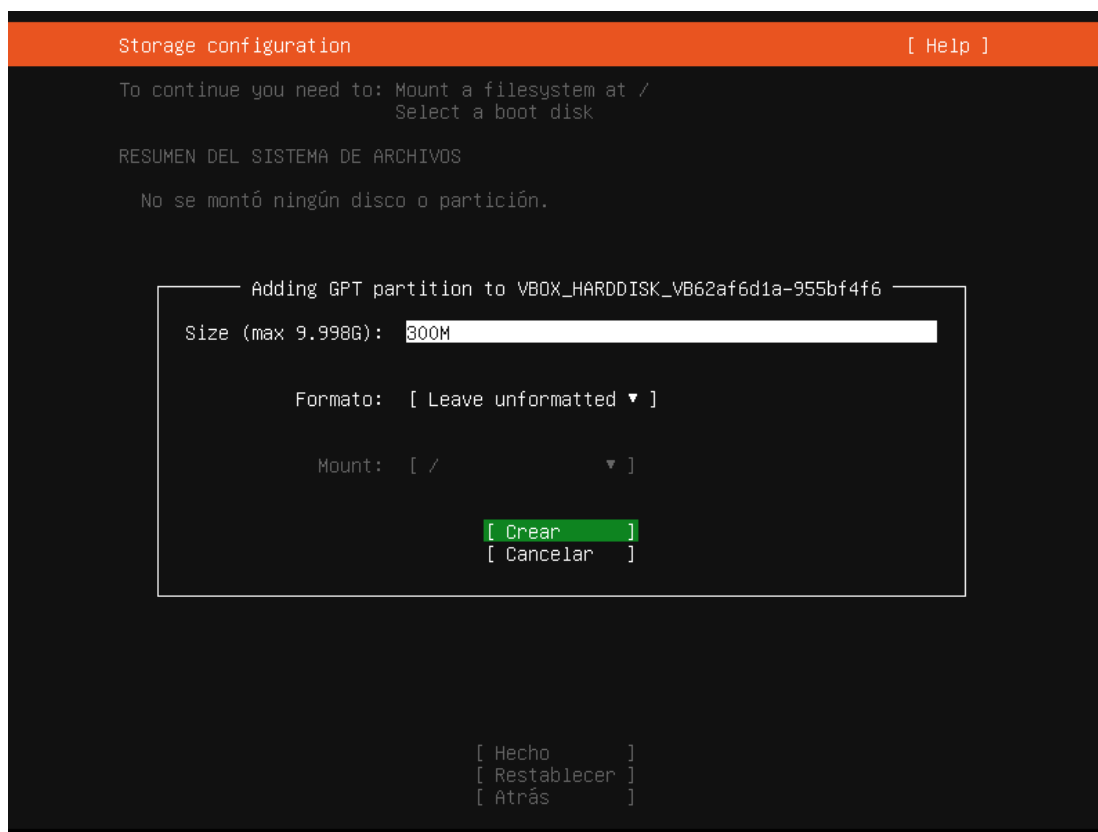
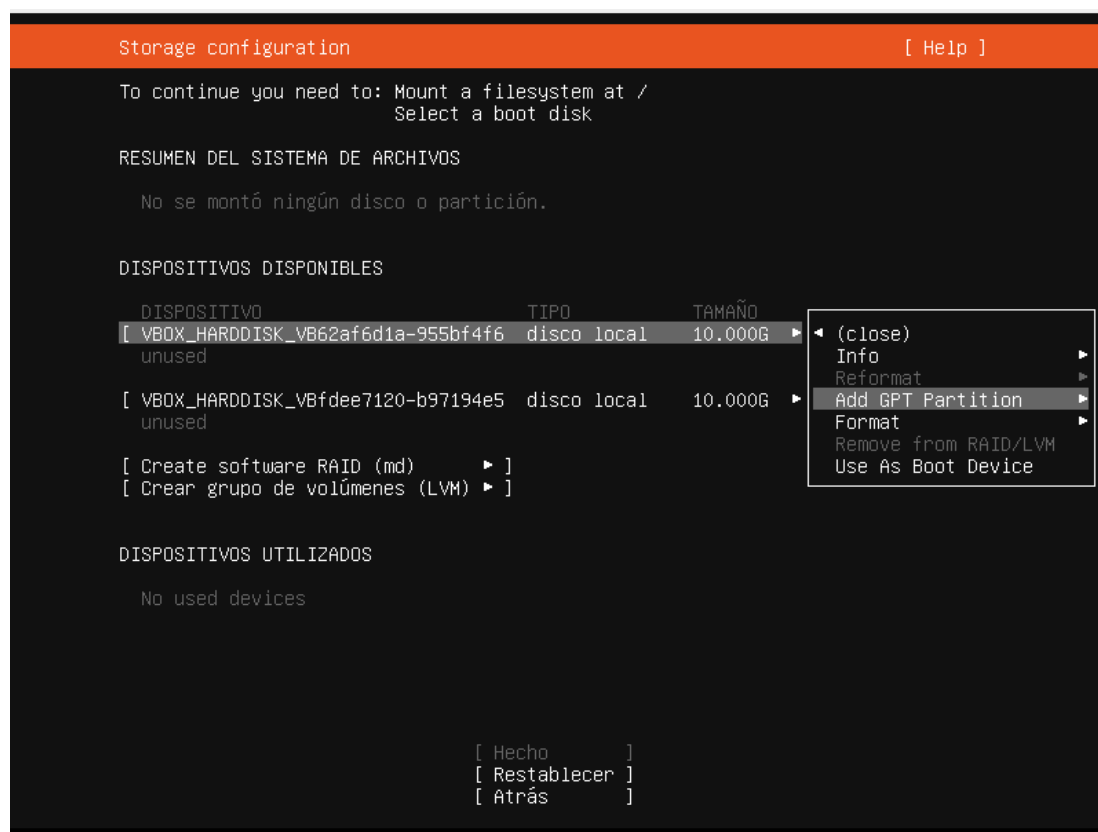


Y tras esto pulsamos sobre elegir imagen de disco y seleccionamos la imagen de Ubuntu Server. Con esto podemos proceder con la instalación.

Iniciamos la máquina y esperamos hasta llegar a la pantalla de selección de idioma. Elegimos español y le damos a hecho. Tras esto pulsamos sobre continuar sin actualizar y en la siguiente pantalla elegimos la distribución del teclado (en nuestro caso español) y le damos a hecho. De las conexiones red no tocamos nada, luego le damos a hecho directamente. Tampoco tocamos nada para la configuración proxy, y tampoco para la mirror adress, en ambos casos pulsamos sobre hecho directamente. En la siguiente pantalla elegimos “*Custom storage layout*”, pues queremos nuestro sistema RAID1 y varios volúmenes lógicos.



En la siguiente pantalla tenemos que crear dos particiones en cada uno de los discos, una de 300M, y otra con lo que quede de espacio (en las cuatro particiones las dejamos sin montar y sin formatear). Para crear las particiones hacemos como sigue



Repetimos el proceso para las tres particiones (1 en este mismo disco y dos en el otro). Si el espacio donde se asigna el tamaño se deja en blanco por defecto se coge todo el espacio libre que le quede al disco. Quedará algo similar a esto

```
Storage configuration [ Help ]

To continue you need to: Mount a filesystem at /

RESUMEN DEL SISTEMA DE ARCHIVOS

No se montó ningún disco o partición.

DISPOSITIVOS DISPONIBLES

DISPOSITIVO TIPO TAMAÑO
[ VBOX_HARDDISK_VB62af6d1a-955bf4f6 disco local 10.000G ▶ ]
partition 2 new, unused 300.000M ▶
partition 3 new, unused 9.704G ▶

[ VBOX_HARDDISK_VBfdee7120-b97194e5 disco local 10.000G ▶ ]
partition 1 new, unused 300.000M ▶
partition 2 new, unused 9.705G ▶

[ Create software RAID (md) ▶ ]
[ Crear grupo de volúmenes (LVM) ▶ ]

DISPOSITIVOS UTILIZADOS

DISPOSITIVO TIPO TAMAÑO
[ VBOX_HARDDISK_VB62af6d1a-955bf4f6 disco local 10.000G ▶ ]
partition 1 new, bios_grub 1.000M ▶

[ Hecho ]
[ Restablecer ]
[ Atrás ]
```

Vemos que en uno de los discos una de las particiones es más grande que la otra. Para solucionar esto pulsamos sobre el disco que tenga la partición más grande y le damos a “Add as another boot device” de forma que añadirá en ese disco una partición de 1 MB que contendrá grub. Ahora creamos los raid, para eso pulsamos sobre “Create software RAID(md)” y agrupamos las de 300MB y luego las dos que queden.

```
Storage configuration [ Help ]

Create software RAID ("MD") disk

Nombre: md0

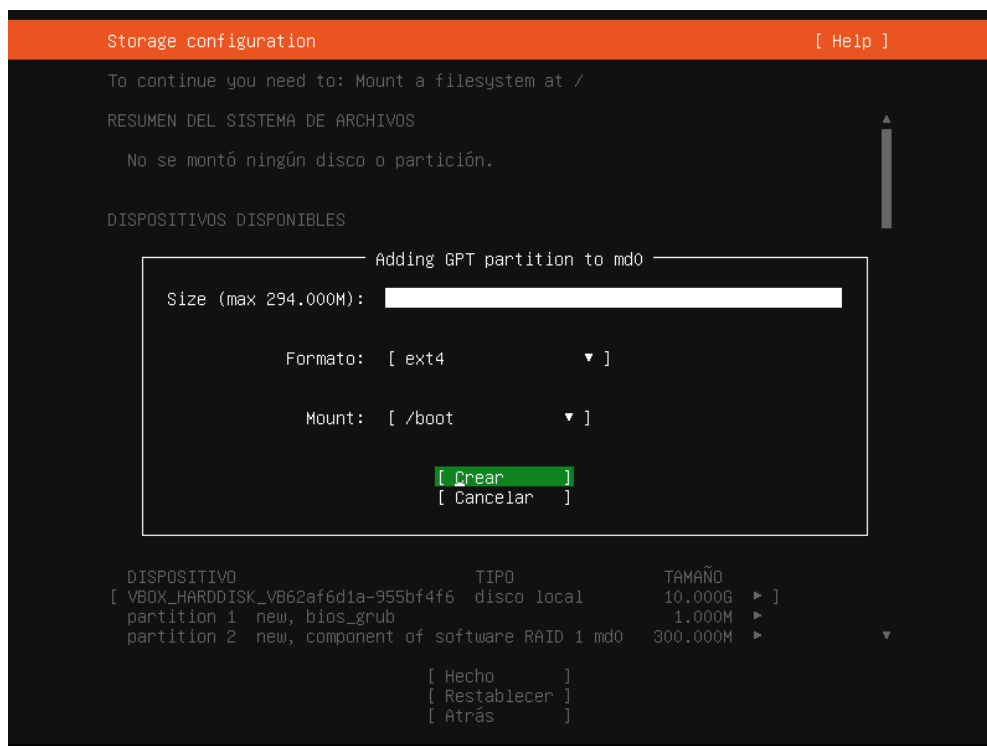
RAID Level: [ 1 (mirrored) ▼ ]

Dispositivos: VBOX_HARDDISK_VB62af6d1a-955bf4f6 10.000G
               disco local
               [X] partition 2 300.000M
                   [ active ▼ ]
                   unused partition of disco local
               [ ] partition 3 9.704G
                   [ active ▼ ]
                   unused partition of disco local
               VBOX_HARDDISK_VBfdee7120-b97194e5 10.000G
               disco local
               [X] partition 2 300.000M
                   [ active ▼ ]
                   unused partition of disco local
               [ ] partition 3 9.704G
                   [ active ▼ ]
                   unused partition of disco local

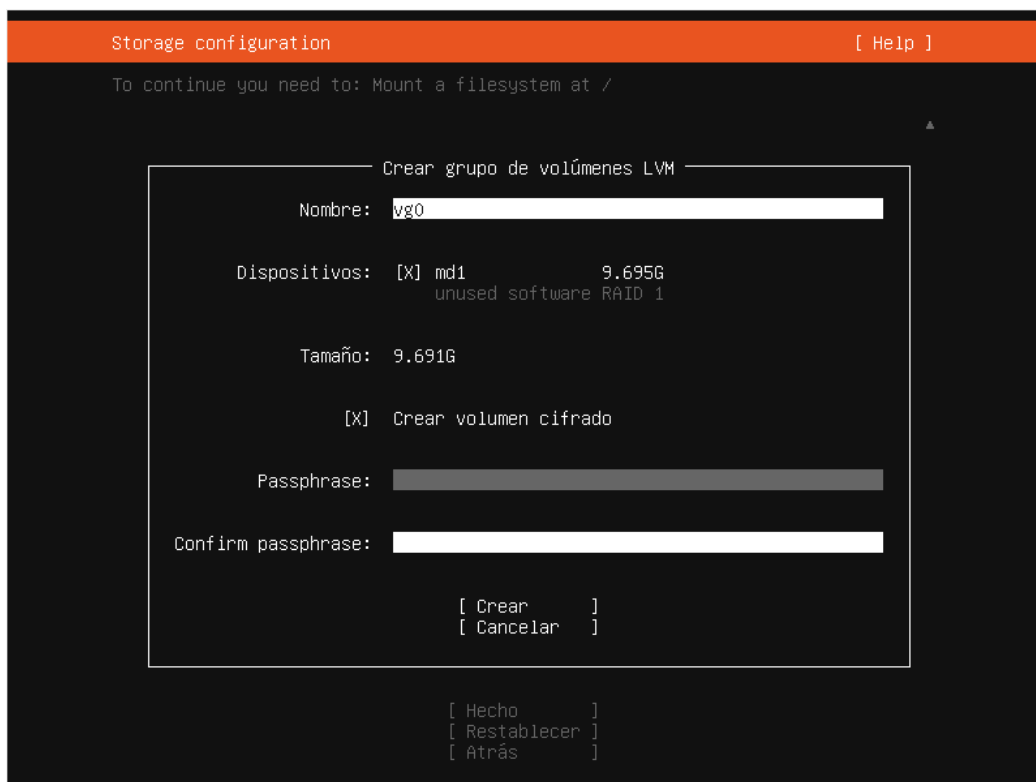
Tamaño: 298.000M

[ Crear ]
[ Cancelar ]
```

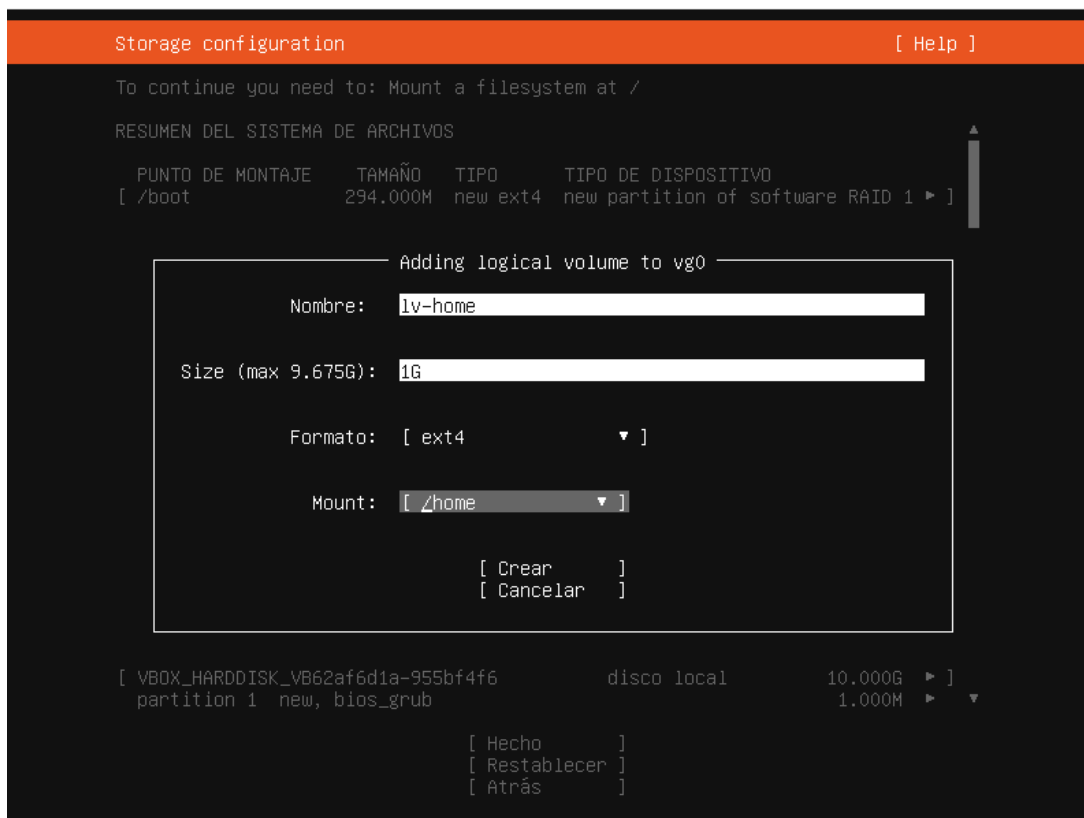
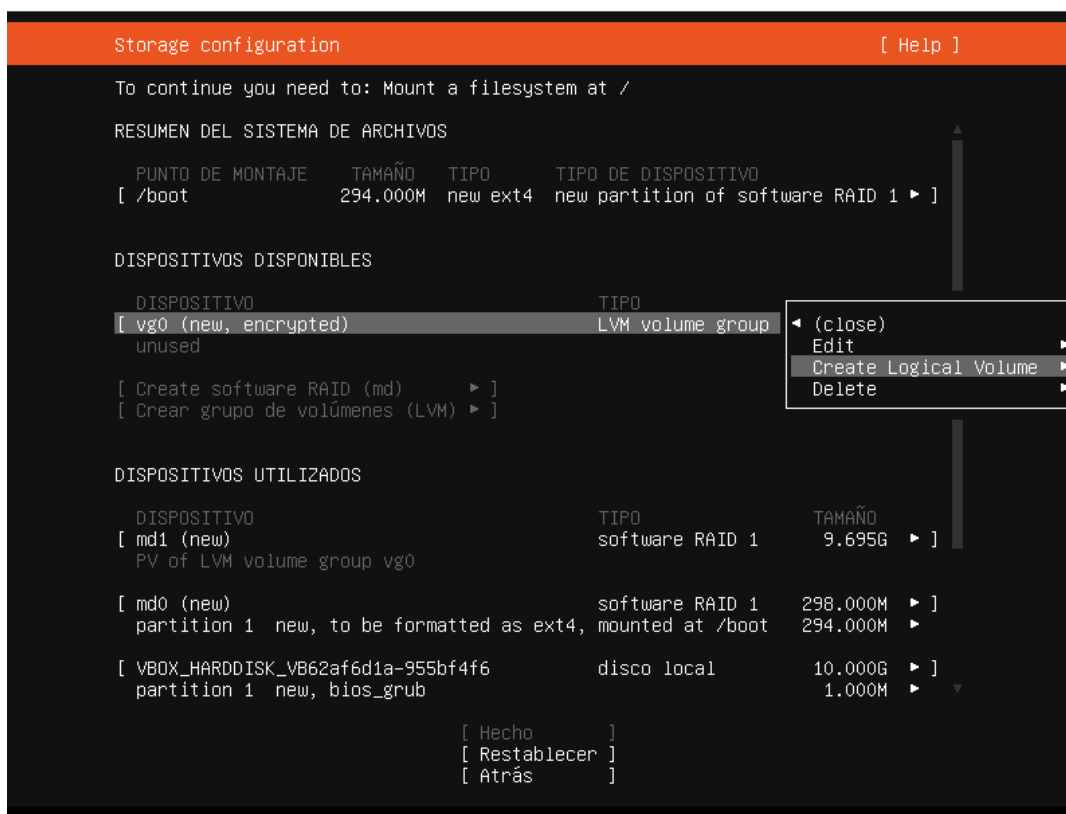

Una vez creado el RAID con las dos particiones de 300MB la formateamos añadiendo una partición GPT sobre el mismo como ya se ha hecho anteriormente.



Ahora creamos otro sistema RAID1 con las dos particiones restantes, pero este a diferencia del anterior no lo formateamos ni particionamos. Vamos a crear grupo de volúmenes y seleccionamos el raid que acabamos de crear y marcamos la opción de crear volumen cifrado.



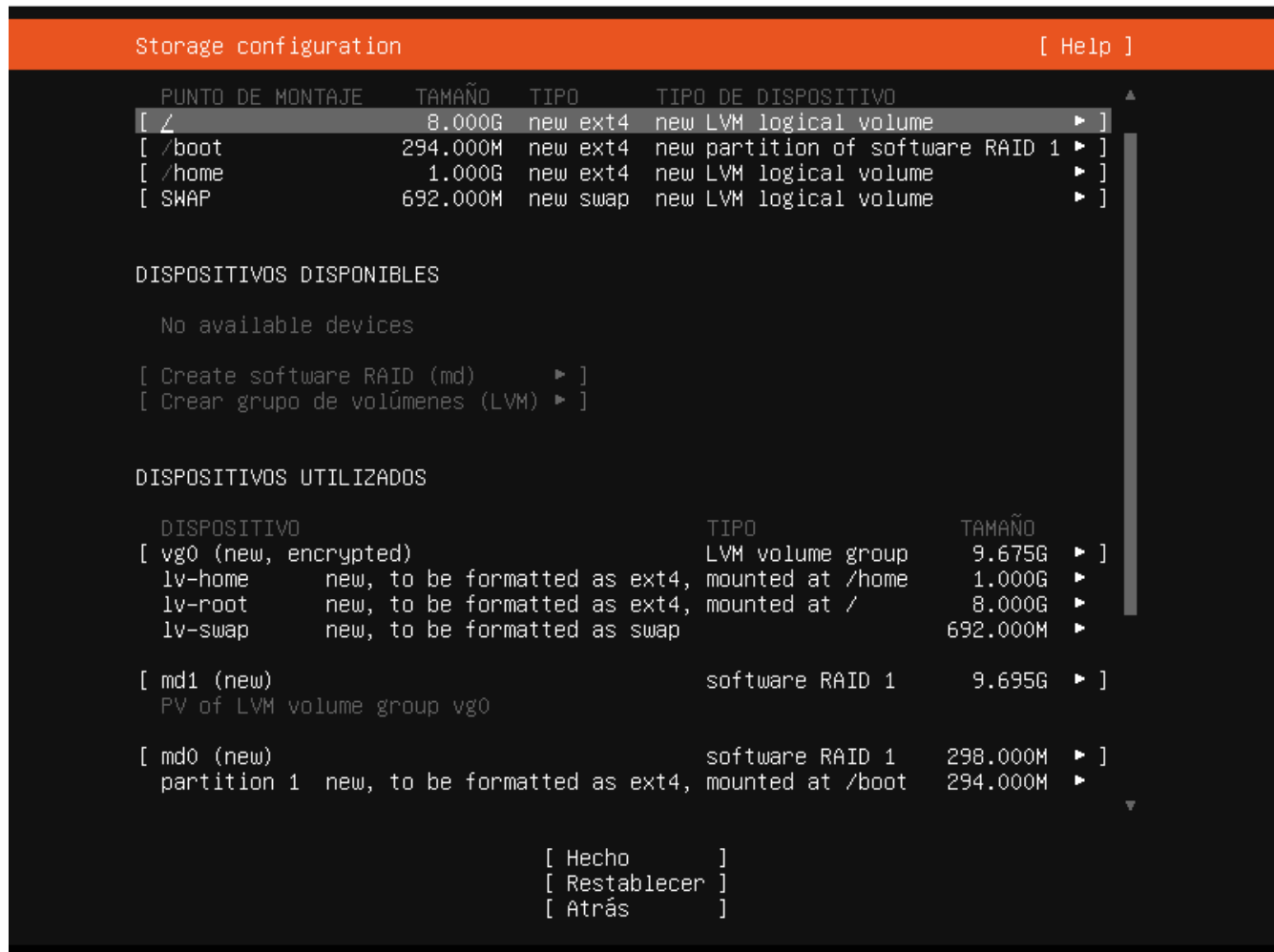
Como contraseña para el volumen cifrado se recomienda *ISE*, aunque se puede elegir cualquier otra siempre y cuando se recuerde. Sobre el grupo de volúmenes que hemos creado creamos volúmenes lógicos como sigue



De estos volúmenes lógicos se crean tres:

- lv-home, de tamaño 1G, formato ext4 y punto de montaje /home
- lv-root, de tamaño 8G, formato ext4 y punto de montaje /
- lv-swap, de tamaño lo que quede y format swap.

Tras todo esto nos debería quedar algo así



Tras todo esto se le da a hecho, y seguidamente a continuar. Justo después se llega a la pantalla de configuración de perfil, se recomienda que para la elección de nombre se mire si la práctica indica algún caso en concreto, sino configurelo como prefiera. Una vez establecido los nombres, nombres de usuario, contraseñas, etc, se le da a hecho y en la siguiente pantalla de nuevo se le da a hecho sin tocar nada. En la siguiente pantalla una vez más se le da a hecho sin tocar nada. Tras esto empezará la instalación de Ubuntu Server.

Debemos estar atentos, pues en un punto se dará la opción de “cancelar actualizaciones y reiniciar” que es la que debemos pulsar. Una vez se haya pulsado esta opción solo queda esperar a que termine y entonces ya tendremos nuestro Ubuntu Server instalado.

```
Se ha completado la instalación. [ Help ]

Ha finalizado la instalación.
/snap/subiquity/1966/usr/bin/python3 true'
  curtin command apt-config
  curtin command in-target
  running 'curtin curthooks'
  curtin command curthooks
    configuring apt configuring apt
    installing missing packages
    configuring iscsi service
    configuring raid (mdadm) service
    installing kernel
    setting up swap
    apply networking config
    writing etc/fstab
    configuring multipath
    updating packages on target system
    configuring pollinate user-agent on target
    updating initramfs configuration
    configuring target system bootloader
    installing grub to target devices
  finalizing installation
  running 'curtin hook'
  curtin command hook
  executing late commands
final system configuration
  configuring cloud-init
  restoring apt configuration
  downloading and installing security updates
  cancelling update |

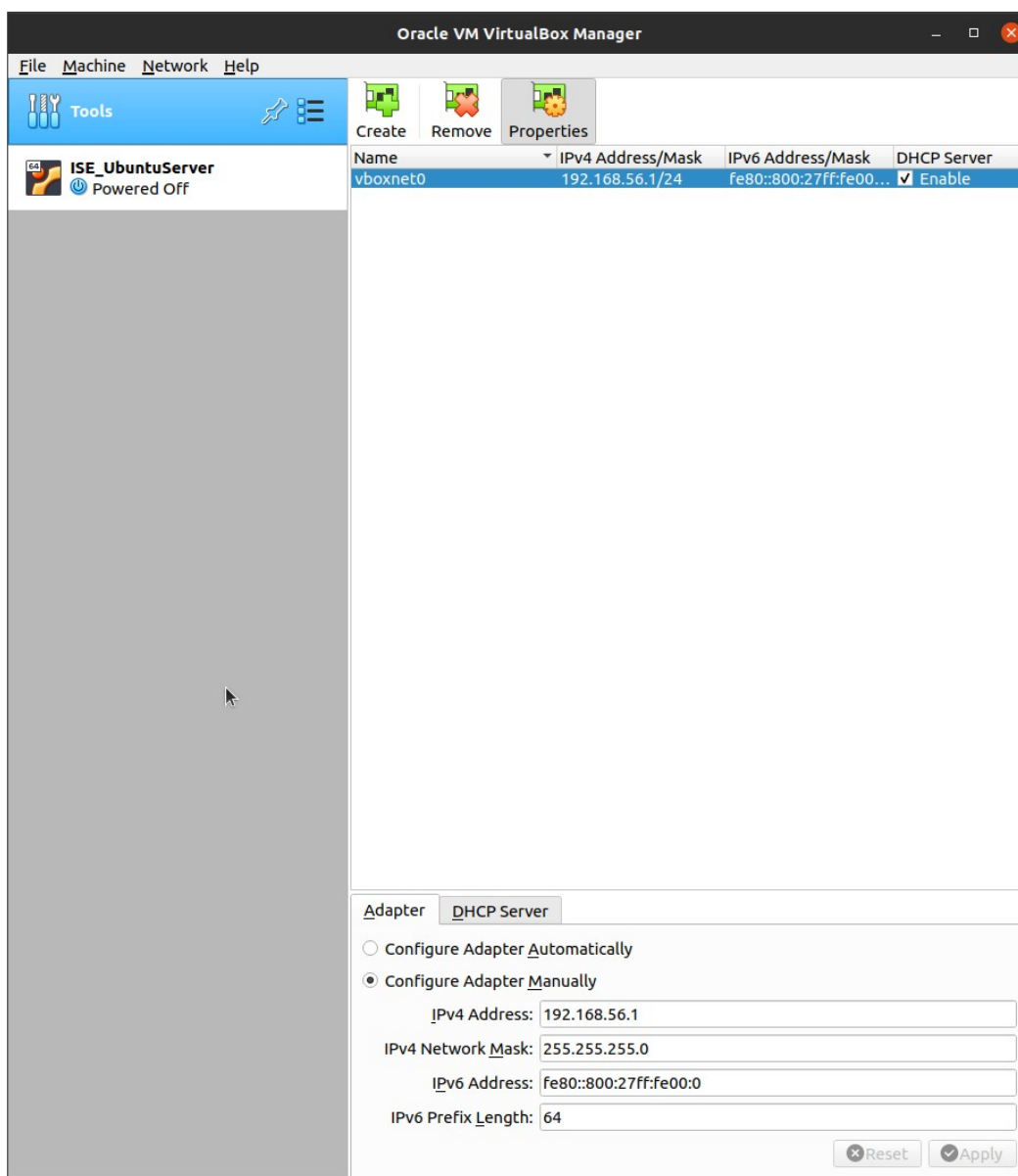
[ View full log ]
[ Rebooting... ]
```

Para comprobar que se haya instalado correctamente simplemente reiniciamos la máquina, introducimos la contraseña que hayamos puesto para desencriptar el disco, iniciamos sesión y ejecutamos el comando “*lsblk*” que debe dar algo similar a esto.

```
momid@momid:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
loop0       7:0      0  29.9M 1 loop  /snap/snapd/8542
loop1       7:1      0   55M 1 loop  /snap/core18/1880
loop2       7:2      0  71.3M 1 loop  /snap/lxd/16099
sda         8:0      0   10G 0 disk
├─sda1       8:1      0    1M 0 part
├─sda2       8:2      0  300M 0 part
├─md0        9:0      0  299M 0 raid1
├─md0p1      259:0    0  294M 0 part  /boot
├─sda3       8:3      0   9.7G 0 part
├─md1        9:1      0   9.7G 0 raid1
├─dm_crypt-0 253:0    0   9.7G 0 crypt
│   └─vg0-lv--home 253:1    0    1G 0 lvm  /home
│       └─vg0-lv--root 253:2    0    8G 0 lvm  /
│           └─vg0-lv--swap 253:3    0  692M 0 lvm  [SWAP]
└─sdb        8:16     0   10G 0 disk
    ├─sdb1     8:17     0    1M 0 part
    ├─sdb2     8:18     0  300M 0 part
    ├─md0      9:0      0  299M 0 raid1
    ├─md0p1    259:0    0  294M 0 part  /boot
    ├─sdb3     8:19     0   9.7G 0 part
    ├─md1      9:1      0   9.7G 0 raid1
    ├─dm_crypt-0 253:0    0   9.7G 0 crypt
│       └─vg0-lv--home 253:1    0    1G 0 lvm  /home
│           └─vg0-lv--root 253:2    0    8G 0 lvm  /
│               └─vg0-lv--swap 253:3    0  692M 0 lvm  [SWAP]
└─sr0       11:0     1 1024M 0 rom

momid@momid:~$ _
```

Ahora procedemos a configurar la interfaz de red. Para ello en VirtualBox, sobre “tools/herramientas” pulsamos en los tres puntos y le damos a redes.



Asegurarse que DHCP Server está activado, IPv4 Address coincide y IPv4 Network Mask también coincide. El resto saldrá más adelante. Tras esto vamos a la configuración de la máquina, a red y en el adaptador dos seleccionamos host-only adapter, y entonces seleccionamos la red que hemos creados. Iniciamos la máquina, desencriptamos el disco e iniciamos sesión. Entonces escribimos lo hacemos “*sudo vi etcnetplan/00-installer-config.yaml*” (si no se hace con sudo el comando no hará efecto). Entonces el archivo nos deberá quedar como sigue

```
network:
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: false
      addresses: [192.168.56.105/24]
  version: 2

"netplan/00-installer-config.yaml" 9L, 191C written
momid@momid:/etc$
```

Le hemos añadido una interfaz de red enp0s8 con dhcp desactivado, y dirección ip 192.168.56.105. Tras esto todo esto ponemos “*netplan apply*” y con un “*ifconfig*” comprobamos que la interfaz está activada. Si “*ifconfig*” no está instalada hacemos “*sudo apt install net-tools*”. Más adelante configuraremos también la interfaz de red para una máquina en CentOS, lo que permitirá comprobar la conexión por medio de un ping entre las máquinas.

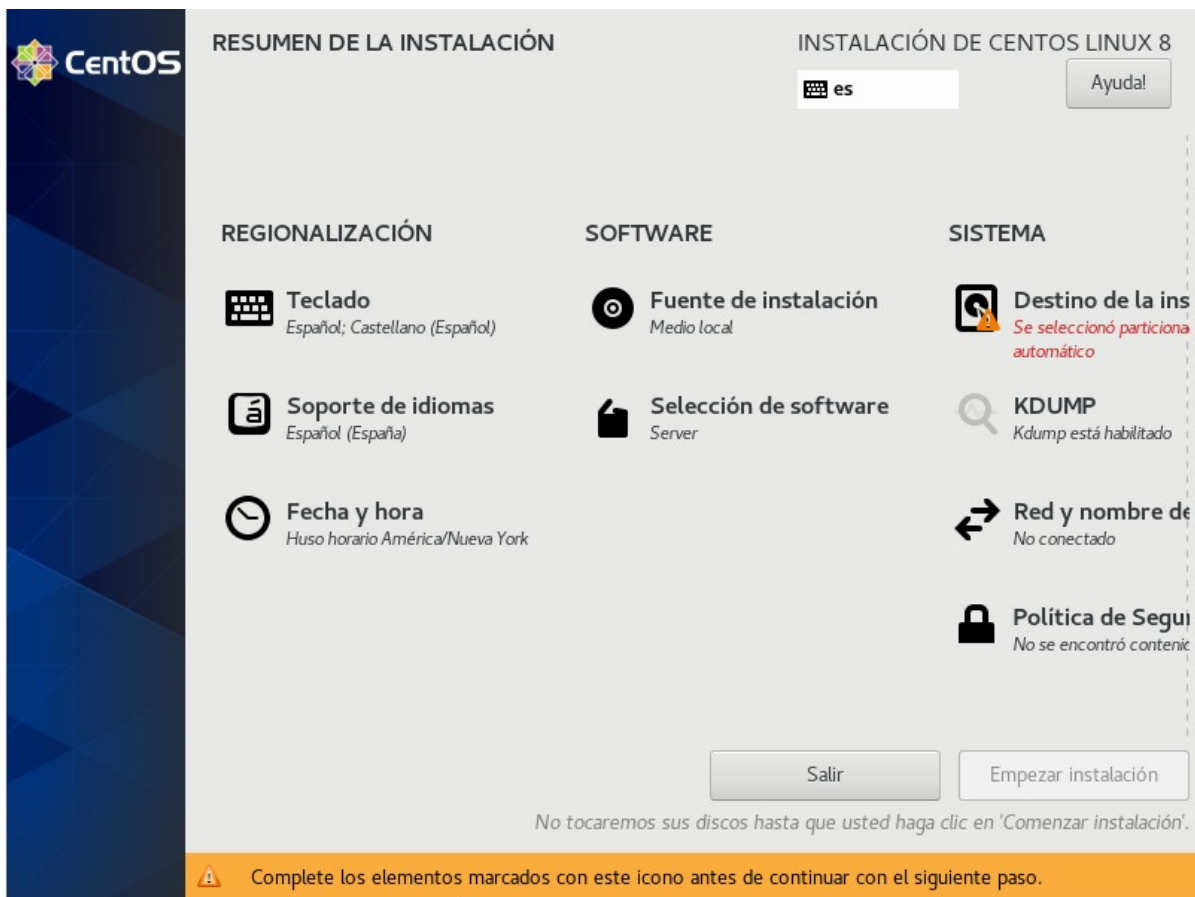
Lección 2.

En esta ocasión, en la empresa en la que le acaban de contratar tenían adquirido un servidor y su predecesor había realizado la instalación del S.O. CentOS, según le han comentado los compañeros, él solía hacer instalaciones por defecto y luego aplicar scripts de configuración. Sin más información, nuestro jefe nos informa que esa máquina va a alojar unos cursos con vídeos de alta calidad y relativamente largos. Por tanto, viendo la configuración del sistemas, prevemos que /var necesitará más espacio, incluso es conveniente asignarle un LV exclusivamente. Para ello, incluiremos un nuevo disco y configuraremos LVM para que /var se monte en el nuevo VL que crearemos para él.

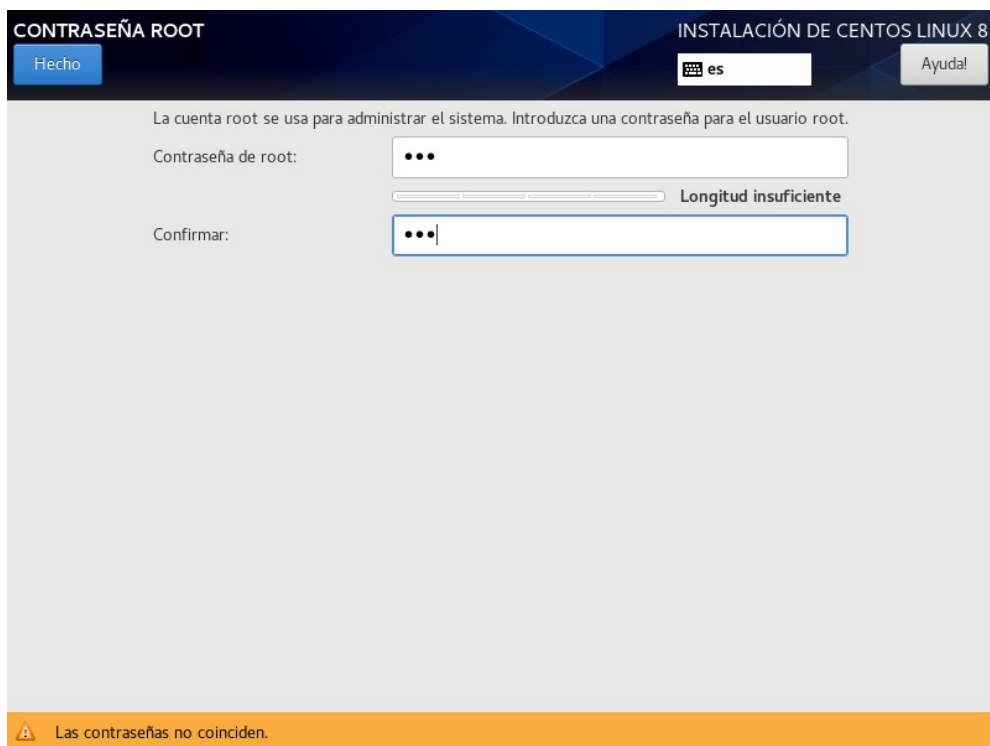
Siguiendo los mismos pasos que para la otra máquina creamos una nueva, esta vez con sistema CentOS. En este caso no tenemos que crear otro disco duro, simplemente cargamos la imagen de CentOS. Entonces iniciamos la máquina y nos saldrá algo así



Le damos a instalar CentOS, y en la siguiente pantalla le damos a idioma español y le damos a hecho. En la siguiente pantalla sale un aviso



Para quitar esta aviso pulsamos sobre el destino de la instalación, y luego sobre hecho. Este aviso sale para que se compruebe el disco sobre el que vamos a instalar el sistema. En este caso solo tenemos un sistema luego es simple. Tras esto empezamos la instalación. Pulsamos sobre contraseña root y la establecemos



Tras poner la contraseña de root, vamos a creación de usuario y creamos nuestro usuario, asegurandonos que lo incluimos como administrador

Tras esto únicamente nos queda esperar a que se complete la instalación. Cuando complete la instalación cerramos la máquina y quitamos la imagen de CentOS, pues sino cuando volvamos a iniciar la máquina lo hará con la instalación. Tras esto ya podemos iniciar la máquina con normalidad.

```
[momid@localhost ~]$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0   8G  0 disk
├─sda1       8:1    0    1G  0 part /boot
├─sda2       8:2    0    7G  0 part
└─c1-root    253:0   0  6.2G  0 lvm  /
   └─c1-swap  253:1   0  820M  0 lvm  [SWAP]
sr0         11:0    1 1024M  0 rom
```

Una vez ya esté instalado el sistema podemos iniciarlo y hacer un “*lsblk*” y quedará algo como lo anterior. Ahora creamos un disco duro para así poder hacer un RAID. Una vez esté creado nos quedará algo así

```
CentOS Linux 8 (Core)
Kernel 4.18.0-193.el8.x86_64 on an x86_64

Activate the web console with: systemctl enable --now cockpit.socket

localhost login: momid
Password:
Last login: Thu Mar 18 17:58:48 on tty1
[momid@localhost ~]$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0   8G  0 disk
├─sda1       8:1    0    1G  0 part /boot
├─sda2       8:2    0    7G  0 part
│   └─cl-root 253:0    0  6,2G  0 lvm  /
│       └─cl-swap 253:1    0  820M  0 lvm  [SWAP]
sdb          8:16   0    8G  0 disk
sr0         11:0    1 1024M  0 rom
[momid@localhost ~]$ _
```

Una vez hecho esto hacemos un “*sudo su*” para entrar en modo root. Tras esto utilizamos el comando “*pvccreate /dev/sdb*”, con lo que creamos un volumen físico. Para comprobar que el volumen físico se ha creado correctamente podemos usar los comandos “*pvddisplay*” o “*pvs*”, lo que nos dará algo así

```
#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

[sudo] password for momid:
[root@localhost momid]# pvccreate /dev/sdb
Physical volume "/dev/sdb" successfully created.
[root@localhost momid]# pvddisplay
--- Physical volume ---
PU Name                /dev/sda2
VG Name                 cl
PU Size                 <7,00 GiB / not usable 3,00 MiB
Allocatable             yes (but full)
PE Size                 4,00 MiB
Total PE                1791
Free PE                 0
Allocated PE            1791
PU UUID                 n9Eynw-zEoL-JjWg-0P2a-jvdQ-gPYq-Radzme

"/dev/sdb" is a new physical volume of "8,00 GiB"
--- NEW Physical volume ---
PU Name                /dev/sdb
VG Name                 cl
PU Size                 8,00 GiB
Allocatable             NO
PE Size                 0
Total PE                0
Free PE                 0
Allocated PE            0
PU UUID                 9h3Hfv-Wxfy-Z1JZ-lUyz-olsi-S8r0-HZ1E8g

[root@localhost momid]# pvs
PU      VG Fmt Attr PSize PFree
/dev/sda2 cl lvm2 a-- <7,00g  0
/dev/sdb cl lvm2 ---  8,00g  8,00g
[root@localhost momid]#
```

Con el comando “*vgextend cl /dev/sdb*” incluimos el volumen físico creado en el grupo de volúmenes *cl*. De nuevo podemos comprobar que ha funcionado con los comando “*vgdisplay*” o “*pvs*”. Nos quedará algo como

```
[root@localhost momid1# vgdisplay
--- Volume group ---
VG Name                cl
System ID
Format                 lvm2
Metadata Areas         2
Metadata Sequence No   4
VG Access               read/write
VG Status               resizable
MAX LV                 0
Cur LV                 2
Open LV                 2
Max PV                 0
Cur PV                 2
Act PV                 2
VG Size                14,99 GiB
PE Size                4,00 MiB
Total PE               3838
Alloc PE / Size        1791 / <7,00 GiB
Free PE / Size         2047 / <8,00 GiB
VG UUID                NA1Ww0-ycxc-ROIF-C0Je-glhm-toZo-ad80oz

[root@localhost momid1# pvs
PV          VG Fmt Attr PSize PFree
/dev/sda2   cl  lvm2 a--  <7,00g    0
/dev/sdb    cl  lvm2 a--  <8,00g  <8,00g
[root@localhost momid1# _
```

Creamos un volumen lógico con el comando “*lvcreate -L 1G -n newvar cl*”. Se puede comprobar que se haya creado con el comando “*lvdisplay*” o “*lvs*”. Usamos el comando “*mkfs -t ext4 /dev/mapper/cl-newvar*”. Tras todo esto nos debería quedar algo así

```
[root@localhost momid1# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0   8G  0 disk
├─sda1       8:1    0    1G  0 part /boot
└─sda2       8:2    0    7G  0 part
   └─cl-root 253:0    0   6,2G  0 lvm  /
      └─cl-swap 253:1  0   820M  0 lvm  [SWAP]
sdb          8:16   0    8G  0 disk
└─cl-newvar 253:2    0    1G  0 lvm
sr0         11:0    1 1024M  0 rom
[root@localhost momid1# _
```

Usamos los comandos “`mkdir /mnt/newvar`” para crear el punto donde montaremos el volumen lógico creado y con “`ls /mnt/`” podemos comprobar que se haya creado la carpeta correctamente. Tras esto usamos el comando “`mount /dev/cl/newvar /mnt/newvar`” con el cual montamos el volumen que habíamos creado en la carpeta que hemos creado anteriormente. En este punto no podemos copiar directamente pues no se copiarían los contextos, además en el sistema podrían haber varios usuarios, luego antes de nada entraremos en modo mantenimiento. Para esto usamos “`systemctl isolate runlevel1.target`” (hay un bug luego es probable que haya que ejecutar esto volver a hacer login como root y volver a ejecutar el comando). Podemos comprobar que ha funcionado

```
[root@localhost ~]# systemctl status
● localhost.localdomain
  State: maintenance
    Jobs: 0 queued
  Failed: 0 units
   Since: Thu 2021-03-18 18:02:35 EDT; 21min ago
  CGroup: /
          └─init.scope
              └─1 /usr/lib/systemd/systemd --switched-root --system --deserialize 17
                  └─system.slice
                      └─rngd.service
                          └─832 /sbin/rngd -f --fill-watermark=0
                              └─systemd-udevd.service
                                  └─698 /usr/lib/systemd/systemd-udevd
                                      └─systemd-journald.service
                                          └─670 /usr/lib/systemd/systemd-journald
                                              └─rescue.service
                                                  └─2498 /usr/lib/systemd/systemd-sulogin-shell rescue
                                                      └─2500 bash
                                                          └─2523 systemctl status
                                                              └─2524 (pager)
[root@localhost ~]#
```

Ahora usamos el comando “`cp -a /var/. /mnt/newvar/`”, con la opción -a nos aseguramos de que se copien los contextos. Tras esto vamos al archivo “`/etc/fstab`” y añadimos una línea para montar el volumen lógico que hemos creado en “`/var`” (falta el formato ext4 entre el punto de montaje y defaults).

```

#
# /etc/fstab
# Created by anaconda on Thu Mar 18 17:52:03 2021
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
/dev/mapper/cl-root    /          xfs     defaults        0 0
UUID=6b59ca3b-5039-42a0-8244-f89414d4c107 /boot      ext4     defaults        1 2
/dev/mapper/cl-swap    swap       swap     defaults        0 0
/dev/mapper/cl-newvar  /var       defaults 0 0

```

-- INSERT --

Usamos el comando “`umount /mnt/newvar`” para desmontar el volumen lógico de donde lo habíamos montado anteriormente. Tras esto se usa el comando “`mount -a`” para que se aplique los cambios puestos en fstab, luego se monta el volumen físico en `var`. Podemos comprobar que ha funcionado usando el comando “`mount`”, y nos debería aparecer en la última línea que nuestro volumen lógico se ha montado en `var` con formato `ext4`. Tras esto usamos “`umount -l /dev/mapper/cl-newvar`” para que el sistema desmonte cuando pueda (opcion `-l`) el volumen. Ahora hacemos “`mv /var /var_OLD`” para mover `/var` a `/var_OLD`, con lo que ya no hay carpeta `var`, pues se ha movido a `var_OLD`, luego si intentamos un “`mount -a`” va a dar un error pues no existe el punto de montaje. Creamos una carpeta “`mkdir /var`” para trabajar con ella. Para restaurar los contextos usamos “`restorecon /var`”. Y por último con “`mount -a`” montamos el volumen lógico nuevo en `/var`, quedando `lsblk` así

```

[root@localhost ~]# mount -a
[ 2258.354522] EXT4-fs (dm-2): mounted filesystem with ordered data mode. Opts: (null)
[root@localhost ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0   8G  0 disk
├─sda1       8:1    0    1G  0 part /boot
├─sda2       8:2    0    7G  0 part
│   └─cl-root 253:0    0  6.2G  0 lvm /
│       └─cl-swap 253:1    0  820M  0 lvm [SWAP]
sdb          8:16   0    8G  0 disk
└─cl-newvar 253:2    0    1G  0 lvm /var
sr0         11:0    1 1024M  0 rom

```

Si hacemos un reboot vemos que el sistema reinicia y el volumen lógico que hemos creado está montado en /var.

Tras esto procederemos a la configuración de redes. Una vez más en la máquina como adaptador 2 ponemos la red que hayamos creado. Después, vamos al archivo “/etc/sysconfig/network-scripts/ifcfg-enp0s3” y en el campo ONBOOT cambiamos el “no” por “yes”. Tras esto creamos un archivo “/etc/sysconfig/network-scripts/ifcfg-enp0s8” y incluimos las siguientes líneas

```
CentOS Linux 8 (Core)
Kernel 4.18.0-193.el8.x86_64 on an x86_64

Activate the web console with: systemctl enable --now cockpit.socket

localhost login: momid
Password:
Last login: Thu Mar 18 18:42:52 on tty1
[momid@localhost ~]$ cat /etc/sysconfig/network-scripts/ifcfg-enp0s8
TYPE=Ethernet
BOOTPROTO=none
NAME=enp0s8
DEVICE=enp0s8
ONBOOT=yes
IPADDR=192.168.56.110
NETMASK=255.255.255.0
[momid@localhost ~]$
```

Luego podemos comprobar con el comando “*ip addr*” que ambas interfaces estén activadas y podremos ejecutar el ping entre máquinas (obviamente ambas máquinas deben estar encendidas) para comprobar la conexión.

Lección 3.

Tras ver el éxito de los vídeos alojados en el servidor configurado en la práctica anterior, un amigo de su cliente quiere proceder del mismo modo pero va a necesitar alojar información sensible así que le pide explícitamente que cifre la información y que ésta esté siempre disponible. Por tanto, la decisión que toma es configurar un RAID1 por software y cifrar el VL en el que /var estará alojado.

En este caso hacemos como en el anterior, una instalación limpia de CentOS. Una vez se haya instalado, sacamos la imagen de CentOS, para que al iniciar no nos salga el instalador, y entonces creamos dos discos duros a parte del ya existente, pues tendremos que crear un RAID sobre ellos (seguir los pasos para creación de discos ya vistos). Una vez ya hayamos hecho todo esto encendemos la máquina y crearemos el RAID. Para ello usaremos la herramienta “mdadm” que en principio no estará instalada. Usamos sudo su para entrar en modo administrador y entonces hacemos “ifup enp0s3” para activar la interfaz de red. Tras esto usamos “yum install mdadm” para instalar la herramienta, y entonces usamos “mdadm –create /dev/md0 –raid-devices=2 –level=1 /dev/sdb /dev/sdc” para crear el RAID, y comprobamos con “lsblk” que se haya creado correctamente. Tras esto creamos un volumen físico para el RAID con “pvcreate /dev/md0” este volumen no pertenece a ningún grupo de volúmenes y podemos comprobar que se ha creado con las herramientas vistas en las lecciones anteriores. Ahora creamos un grupo de volúmenes para este volumen físico con “vgcreate raidvg /dev/md0” y con esto habremos creado el grupo raidvg y habremos enganchado a este el volumen físico que hemos creado. Tras esto crearemos un volumen lógico con “lvcreate -n newvar -L 1G raidvg” y con esto habremos creado un volumen lógico de un GB llamado newvar enganchado al grupo raidvg. Tras todo esto nos debería quedar algo así

```
[root@localhost momidl# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda          8:0    0   8G  0 disk
├─sda1       8:1    0    1G  0 part  /boot
├─sda2       8:2    0    7G  0 part
│   └─cl-root 253:0    0  6,2G  0 lvm    /
│       └─cl-swap 253:1    0  820M  0 lvm    [SWAP]
sdb          8:16   0    8G  0 disk
└─md0        9:0    0    8G  0 raid1
   └─raidvg-newvar 253:2    0    1G  0 lvm
sdc          8:32   0    8G  0 disk
└─md0        9:0    0    8G  0 raid1
   └─raidvg-newvar 253:2    0    1G  0 lvm
sr0         11:0    1 1024M  0 rom

[root@localhost momidl# lvs
LV      VG      Attr      LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
root    cl       -wi-ao--- <6,20g
swap    cl       -wi-ao--- 820,00m
newvar  raidvg  -wi-a----- 1,00g

[root@localhost momidl#
```

Tras esto deberemos encriptar, y para encriptar usaremos la herramienta “cryptsetup” que en principio debería estar instalada aunque podemos actualizarla con “yum install cryptsetup”.

Usamos “*cryptsetup luksFormat /dev/raidvg/newvar*” y nos saldrá algo así

```
[root@localhost momid]# cryptsetup luksFormat /dev/raidvg/newvar

WARNING!
=====
Esto sobrescribirá los datos en /dev/raidvg/newvar de forma irrevocable.

Are you sure? (Type 'yes' in capital letters): YES
Introduzca la frase contraseña de /dev/raidvg/newvar:
Verifique la frase contraseña:
[root@localhost momid]# _
```

Tras esto usaremos “*cryptsetup luksOpen /dev/mapper/raidvg-newvar raidvg-newvar_crypt*” para abrir o desencriptar la partición encriptada. Nos pedirá la contraseña y lo introducimos y entonces nos quedará algo así

```
[root@localhost momid]# cryptsetup luksOpen /dev/mapper/raidvg-newvar raidvg-newvar_crypt
Introduzca la frase contraseña de /dev/mapper/raidvg-newvar:
[root@localhost momid]# lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	8G	0	disk	
└─sda1	8:1	0	1G	0	part	/boot
└─sda2	8:2	0	7G	0	part	
└─┬─cl-root	253:0	0	6,2G	0	lvm	/
└─┬─cl-swap	253:1	0	820M	0	lvm	[SWAP]
sdb	8:16	0	8G	0	disk	
└─md0	9:0	0	8G	0	raid1	
└─┬─raidvg-newvar	253:2	0	1G	0	lvm	
└─┬─raidvg-newvar_crypt	253:3	0	1000M	0	crypt	
sdc	8:32	0	8G	0	disk	
└─md0	9:0	0	8G	0	raid1	
└─┬─raidvg-newvar	253:2	0	1G	0	lvm	
└─┬─raidvg-newvar_crypt	253:3	0	1000M	0	crypt	
sr0	11:0	1	1024M	0	rom	

```
[root@localhost momid]# _
```

Ahora haremos la copia de /var en newvar. Lo primero será entrar en modo de mantenimiento como ya vimos anteriormente. De nuevo creamos la carpeta de montaje “*mkdir /mnt/newvar*” y ahora montamos

encriptada de donde estaba con “*umount /mnt/newvar*” y comprobamos con *lsblk* que se haya desmontado. Y finalmente hacemos un “*mount -a*” para que se relea *fstab* y hacemos un *lsblk* que nos quedará

```
[root@localhost ~]# lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda                                  8:0    0   8G  0 disk
├─sda1                              8:1    0    1G  0 part  /boot
├─sda2                              8:2    0    7G  0 part
│   └─cl-root                       253:0    0   6,2G  0 lvm    /
│       └─cl-swap                   253:1    0   820M  0 lvm    [SWAP]
sdb                                  8:16    0    8G  0 disk
├─md0                               9:0    0    8G  0 raid1
│   └─raidvg-newvar                 253:2    0    1G  0 lvm
│       └─raidvg-newvar_crypt       253:3    0  1000M  0 crypt  /var
sdc                                  8:32    0    8G  0 disk
├─md0                               9:0    0    8G  0 raid1
│   └─raidvg-newvar                 253:2    0    1G  0 lvm
│       └─raidvg-newvar_crypt       253:3    0  1000M  0 crypt  /var
sr0                                 11:0    1  1024M  0 rom
```

Al igual que en el sistema Ubuntu Server nos pedía la contraseña para abrir la partición entonces ahora tenemos que indicar en el archivo “*/etc/crypttab*” la partición que hay que desencriptar al iniciar el sistema. Para ello tendremos que crear este fichero pero lo haremos de la siguiente manera “*blkid | grep crypto >> /etc/crypttab*” de forma que a este fichero se va a mandar la linea de *blkid* que nos interesa. Tras esto vamos al fichero *crypttab* y nos tiene que quedar de la siguiente manera

```
[root@localhost ~]# cat /etc/crypttab
raidvg-newvar_crypt UUID=977820c7-71c5-449f-b143-99f4a0e26d41 none
[root@localhost ~]#
```

Al UUID le quitamos las comillas, sustituimos el TYPE por un none que indica que se tiene que pedir la contraseña siempre y dejamos solo el nombre de la partición. Tras esto reiniciamos y si todo va bien debería funcionar.

Tras reiniciar si arranca bien *lsblk* nos mostrará esto

```

CentOS Linux 8 (Core)
Kernel 4.18.0-193.el8.x86_64 on an x86_64

Activate the web console with: systemctl enable --now cockpit.socket

Hint: Num Lock on

localhost login: momid
Password:
Last login: Thu Mar 18 19:47:02 on tty1
[momid@localhost ~]$ lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda                                  8:0      0   8G  0 disk
├─sda1                              8:1      0   1G  0 part  /boot
├─sda2                              8:2      0   7G  0 part
│   └─cl-root                       253:0      0  6.2G  0 lvm    /
│   └─cl-swap                       253:1      0  820M  0 lvm    [SWAP]
sdb                                  8:16     0   8G  0 disk
├─md0                               9:0      0   8G  0 raid1
│   └─raidvg-newwar                 253:2      0   1G  0 lvm
│       └─raidvg-newwar_crypt       253:3      0 1000M  0 crypt  /var
sdc                                  8:32     0   8G  0 disk
├─md0                               9:0      0   8G  0 raid1
│   └─raidvg-newwar                 253:2      0   1G  0 lvm
│       └─raidvg-newwar_crypt       253:3      0 1000M  0 crypt  /var
sr0                                 11:0     1 1024M  0 rom
[momid@localhost ~]$ _

```

Si pasa más de un minuto y medio pensando significa que hay algún error y tras ese tiempo nos meterá en modo mantenimiento, con lo que tendremos que ir a revisar el fichero crypttab en busca de algún error.

Con esto tendríamos las tres lecciones hechas.