

Born2beroot

▼ Tabla de contenidos

[Tabla de contenidos](#)

[Parte Obligatoria](#)

[Parte Opcional](#)

[Conocimientos Generales](#)

[Desarrollo](#)

▼ Parte Obligatoria

Acciones

- He pegado la firma del disco virtual de mi máquina en el archivo `signature.txt`
- He entregado un archivo llamado `signature.txt` en la raíz de mi repositorio.
- Estoy seguro de que he instalado el número mínimo de servicios de mi servidor. No he incluido ninguna interfaz gráfica.
- Tengo la última versión estable de Debian.
- He creado al menos 2 particiones cifradas usando `LVM`

```
wil@wil:~$ lsblk
NAME           MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda            8:0    0   8G  0 disk 
└─sda1          8:1    0 487M  0 part /boot
└─sda2          8:2    0   1K  0 part 
└─sda5          8:5    0 7.5G  0 part 
  └─sda5_crypt 254:0   0 7.5G  0 crypt 
    ├─wil--vg-root 254:1   0 2.8G  0 lvm   /
    ├─wil--vg-swap_1 254:2   0 976M  0 lvm   [SWAP]
    └─wil--vg-home 254:3   0 3.8G  0 lvm   /home
sr0           11:0   1 1024M 0 rom 

wil@wil:~$ _
```

- Estoy seguro de que el servicio `SSH` solo se ejecuta sobre el puerto 4242
- Me he asegurado de que el root no puede conectarse a través de SSH
- He configurado `UFW` como cortafuegos y he dejado abierto exclusivamente el puerto 4242

- El Firewall está activo al ejecutar la máquina virtual
- El hostname de mi máquina es mi login terminado en 42 (fsanchez42)

▼ Implementar una política de contraseñas fuerte

- He instalado y configurado `sudo`
- El usuario root existe
- Existe un usuario con mi login (fsanchez)
- He configurado el usuario fsanchez para que pertenezca a los grupos `user42` y `sudo`
- Las contraseñas deben tener como mínimo 10 caracteres de longitud y contener una mayúscula y un número. No debe contener más de 3 veces consecutivas el mismo carácter.
- La contraseña no debe contener el nombre del usuario
- La contraseña debe tener al menos 7 caracteres que no sean parte de la antigua contraseña.
Excepto en el caso del root
- La contraseña del root debe seguir esta política.
- La contraseña debe expirar cada 30 días
- El número mínimo de días permitido antes de modificar una contraseña debe ser de 2 días
- El usuario debe recibir un mensaje de aviso 7 días antes de que su contraseña caduque
- Me he asegurado de cambiar las contraseñas de todas las cuentas presentes en la máquina virtual

▼ Implementar una política de contraseñas fuerte para el grupo sudo

- He establecido un máximo de 3 contraseñas incorrectas consecutivas para este grupo
- Me he asegurado de que tanto los inputs como los outputs ejecutados con sudo se archivan en `/var/log/sudo/`
- He activado el modo TTY (por seguridad)
- He restringido los directorios que utiliza sudo:
 - /usr/local/sbin
 - /usr/local/bin
 - /usr/sbin
 - /usr/bin
 - /sbin
 - /bin
 - /snap/bin

▼ Creación del script `monitoring.sh`

- Eh comprobado que el script se ejecuta al iniciar el servidor y cada 10 minutos
- Estoy seguro de que el script no muestra ningún error
- El script muestra:
 - La arquitectura de mi sistema operativo y mi versión de kernel
 - El número de núcleos físicos
 - El número de núcleos virtuales
 - La memoria RAM disponible actualmente en mi servidor y su porcentaje de uso
 - El espacio en disco actualmente en mi servidor y su porcentaje de uso
 - El porcentaje actual de uso de los núcleos
 - La fecha y hora del último reinicio
 - Si LVM está o no activo
 - El número de conexiones activas
 - El número de usuarios del servidor
 - La dirección IPv4 y la MAC de tu servidor
 - El número de comandos ejecutados en `sudo`

```
Broadcast message from root@wil (tty1) (Sun Apr 25 15:45:00 2021):  
  
#Architecture: Linux wil 4.19.0-16-amd64 #1 SMP Debian 4.19.181-1 (2021-03-19) x86_64 GNU/Linux  
#CPU physical : 1  
#vCPU : 1  
#Memory Usage: 74/987MB (7.50 %)  
#Disk Usage: 1009/2Gb (39%)  
#CPU load: 6.7%  
#Last boot: 2021-04-25 14:45  
#LVM use: yes  
#Connexions TCP : 1 ESTABLISHED  
#User log: 1  
#Network: IP 10.0.2.15 (08:00:27:51:9b:a5)  
#Sudo : 42 cmd
```

▼ Parte Opcional

- He configurado correctamente las particiones según se ve en la imagen inferior

# lsblk						
NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	30.8G	0	disk	
└─sda1	8:1	0	500M	0	part	/boot
└─sda2	8:2	0	1K	0	part	
└─sda5	8:5	0	30.3G	0	part	
└─sda5_crypt	254:0	0	30.3G	0	crypt	
└─LVMGroup-root	254:1	0	10G	0	lvm	/
└─LVMGroup-swap	254:2	0	2.3G	0	lvm	[SWAP]
└─LVMGroup-home	254:3	0	5G	0	lvm	/home
└─LVMGroup-var	254:4	0	3G	0	lvm	/var
└─LVMGroup-srv	254:5	0	3G	0	lvm	/srv
└─LVMGroup-tmp	254:6	0	3G	0	lvm	/tmp
└─LVMGroup-var--log	254:7	0	4G	0	lvm	/var/log
sr0	11:0	1	1024M	0	rom	

- He configurado un sitio WordPress funcional con los servicios
 - lighttpd
 - MariaDB
 - PHP
- He configurado un servicio que considere útil (excepto NGINX y Apache2)

▼ Conocimientos Generales

- Conozco las diferencias entre [Debian](#) y [Centos](#)
- Sé explicar qué es LVM
- Conozco las diferencias entre [aptitude](#) y [apt](#)
- Conozco las diferencias entre [SELinux](#) y [AppArmor](#)
- Sé qué es [SSH](#) y cómo funciona

Los servidores Linux suelen administrarse remotamente usando SSH mediante la conexión con un servidor [OpenSSH](#), que es el software de servidor SSH predeterminado usado en Ubuntu, Debian, CentOS, y la mayoría de los otros sistemas basados en Linux/BSD.

El servidor OpenSSH es la parte de servidor de SSH, también conocida como daemon SSH o [sshd](#). Puede conectar con un servidor OpenSSH usando el cliente OpenSSH: el comando [ssh](#). Puede obtener más información sobre el modelo cliente-servidor SSH en [Puntos esenciales de SSH: Trabajar con](#)

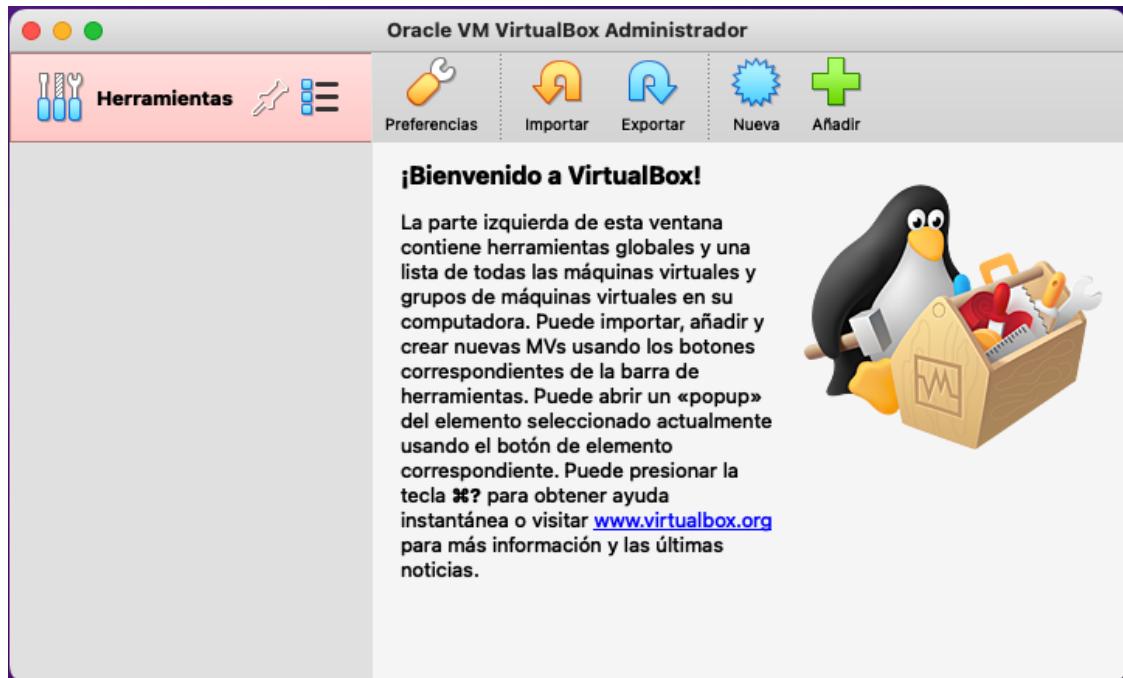
servidores, clientes y claves SSH. Proteger de forma adecuada su servidor OpenSSH es muy importante, ya que actúa como la puerta principal o de entrada a su servidor.

- Sé crear un nuevo usuario
- Sé como cambiar el nombre del hostame
- Sé cómo asignar un grupo a un usuario (`usermod`)
- Entiendo lo que es el modo `TTY` de una contraseña fuerte
- Sé que es `wall` y cómo funciona
- Entiendo el funcionamiento del script `monitoring.sh` y soy capaz de interrumpirlo sin modificarlo
- Soy capaz de defender por qué he elegido el servicio adicional para el apartado de bonus

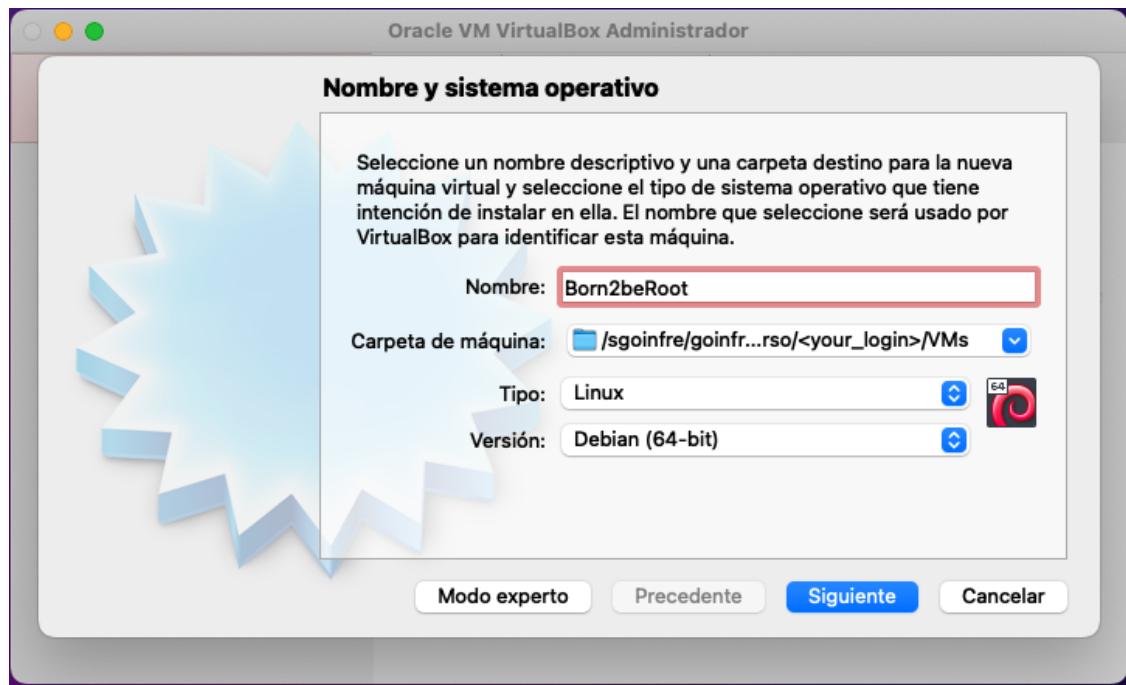
▼ Desarrollo

▼ Creación de la máquina virtual

1. En el Administrador de VirtualBox pulsamos sobre 'Nueva'



2. Completamos los campos '**Nombre**' y '**Carpeta de máquina**' y pulsamos '**Siguiente**'



3. Dejamos 1024 MB como el tamaño de memoria deseado y pulsamos 'Siguiente'



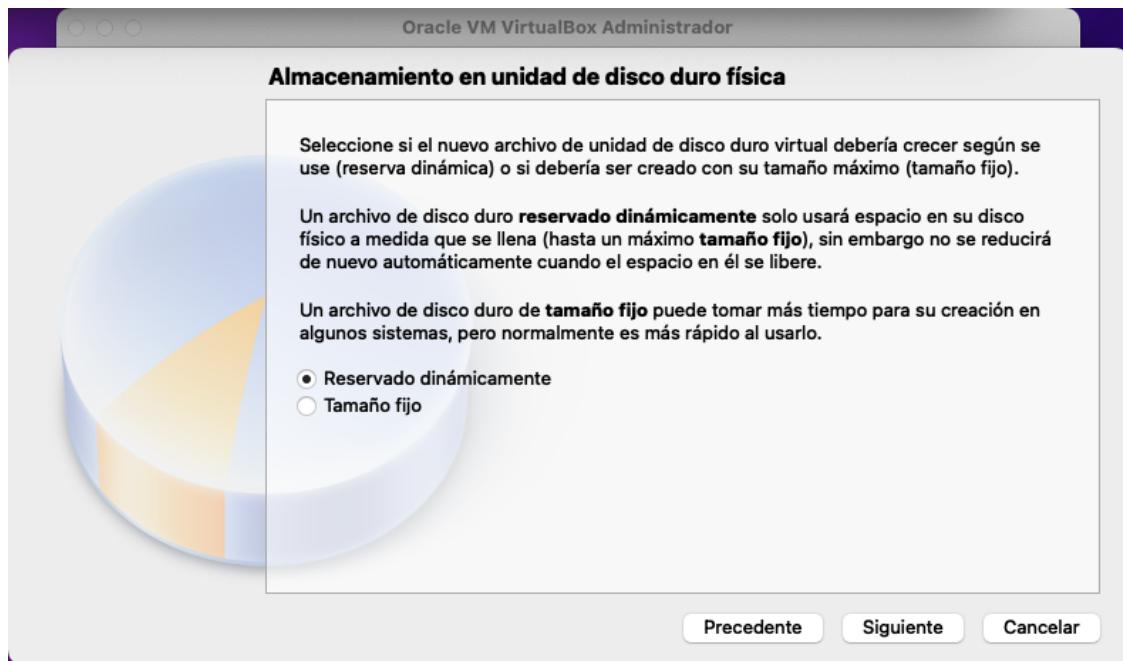
4. Marcamos la opción 'Crear un disco duro virtual ahora' y luego pulsamos 'Crear'



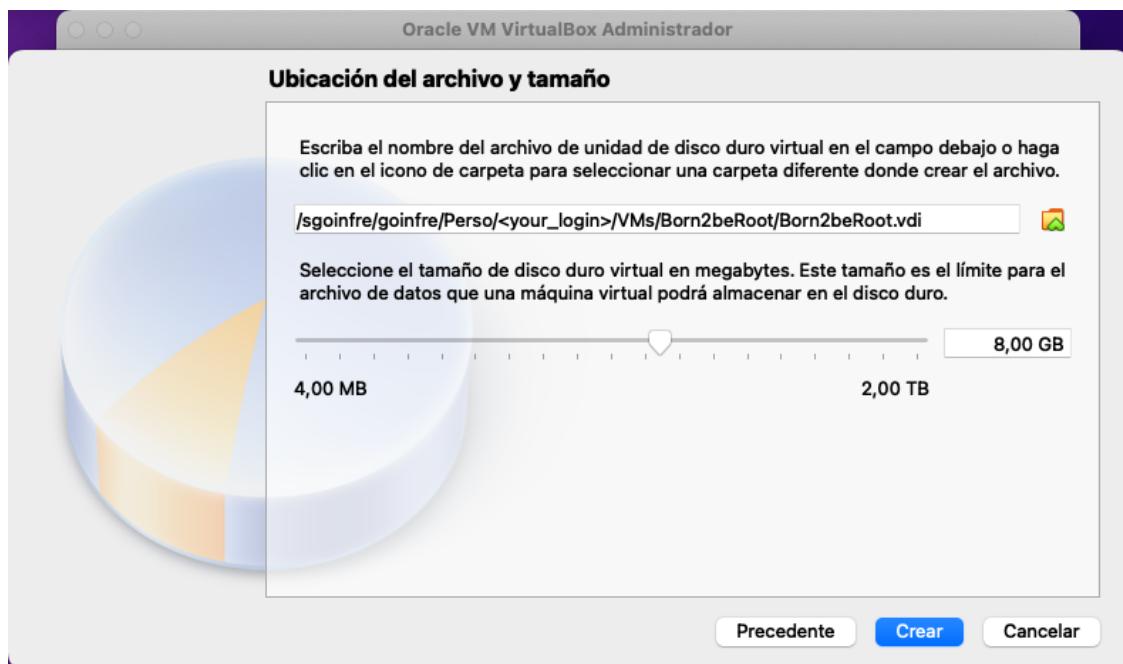
5. Seleccionamos la opción '**VDI (VirtualBox Disk Image)**' y pulsamos '**Siguiente**'



6. Marcamos la opción '**Reservado dinámicamente**' y pulsamos '**Siguiente**'



7. Comprobamos la ruta de ubicación del disco virtual, dejamos 8,00 GB como tamaño del mismo y pulsamos '[Crear](#)'

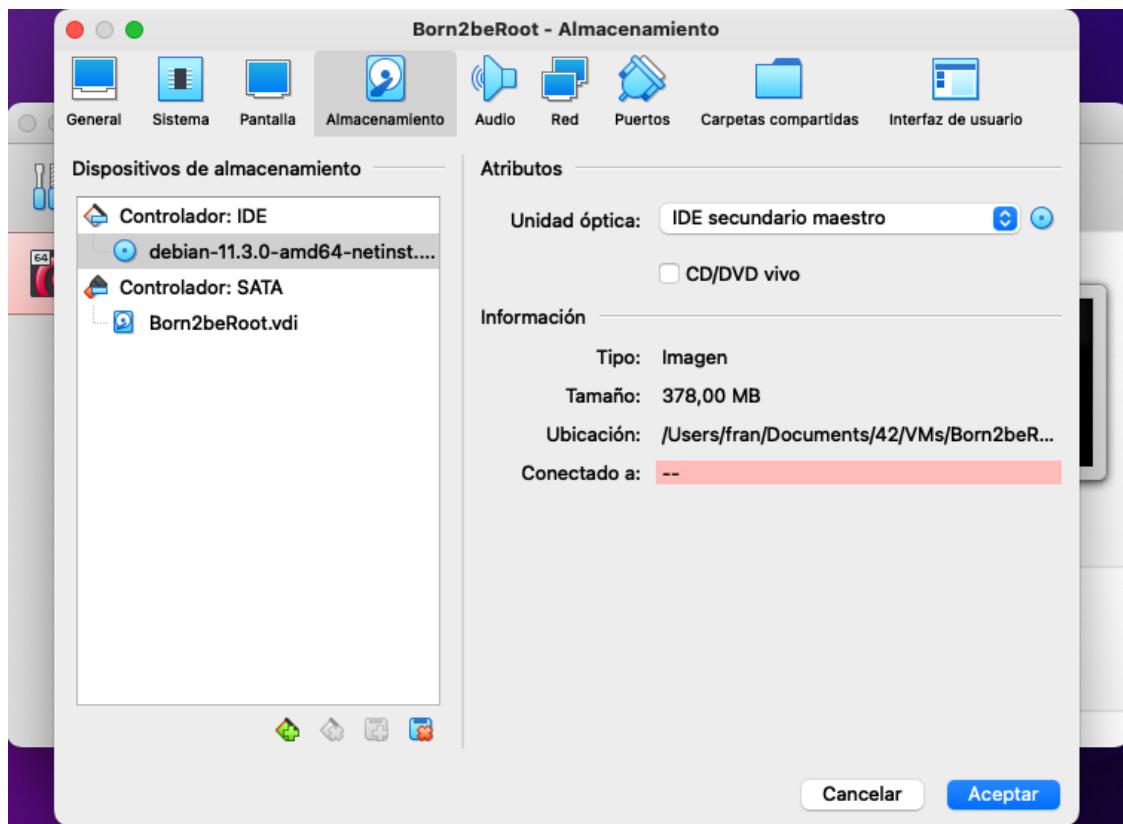


▼ Configuración de la máquina virtual

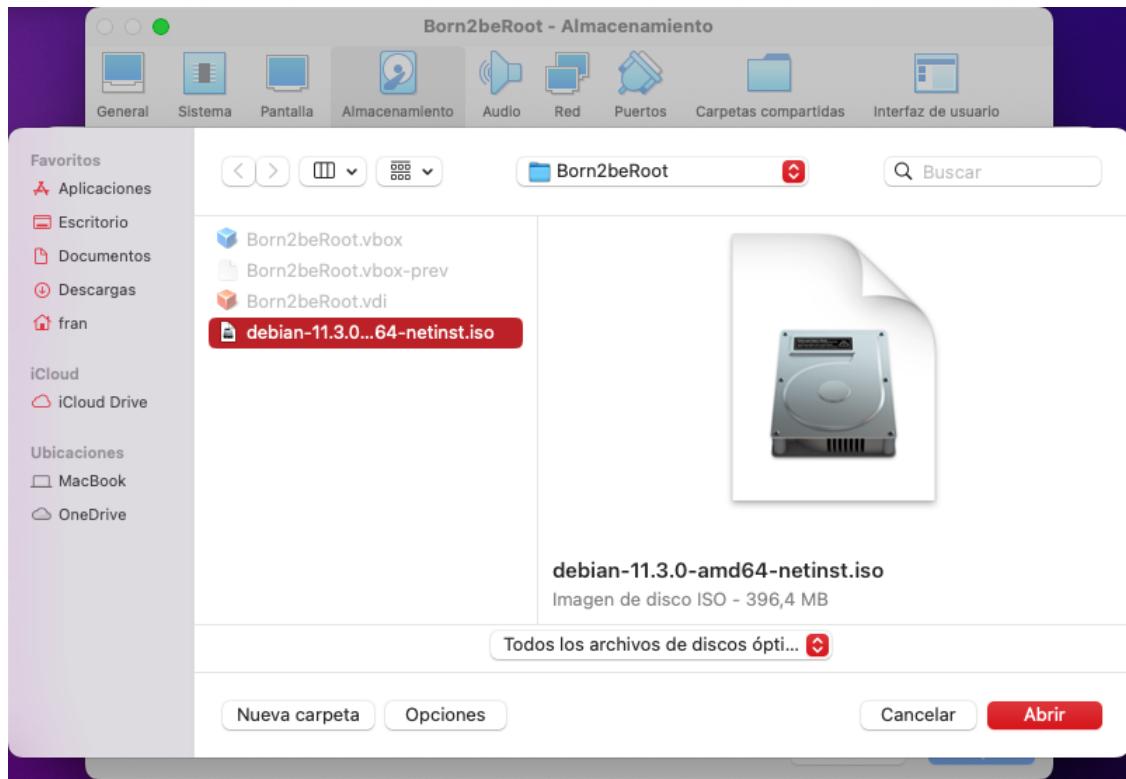
1. Una vez de vuelta al Administrador de VirtualBox, con la máquina virtual creada, pulsamos en '[Configuración](#)'



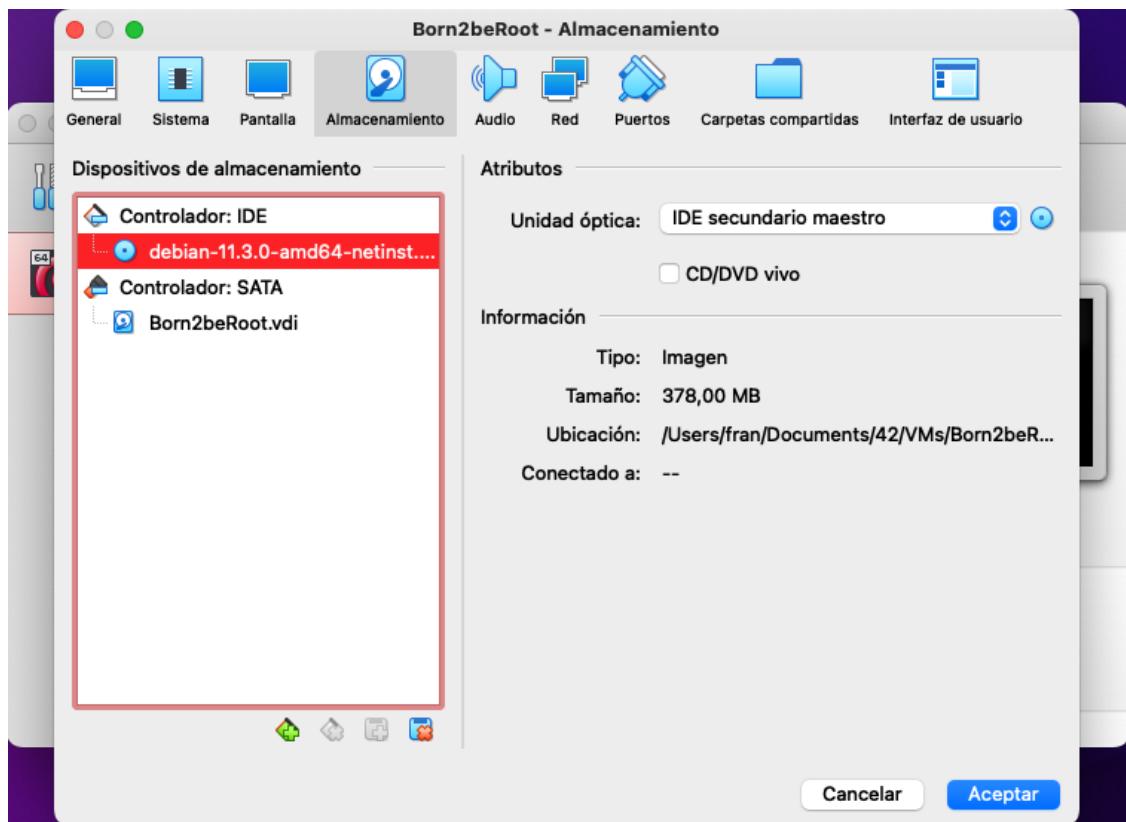
2. En la pestaña '**Almacenamiento**' seleccionamos el medio '**Vacio**' de la unidad **Controlador: IDE**. Pulsamos sobre el icono de CD junto al desplegable **Unidad óptica** y escogemos '[Seleccionar un archivo de disco...](#)'



3. En la nueva ventana navegamos hasta la ruta en la que se encuentra la imagen .iso de Debian y después de seleccionarla, pulsamos en '**Abrir**'

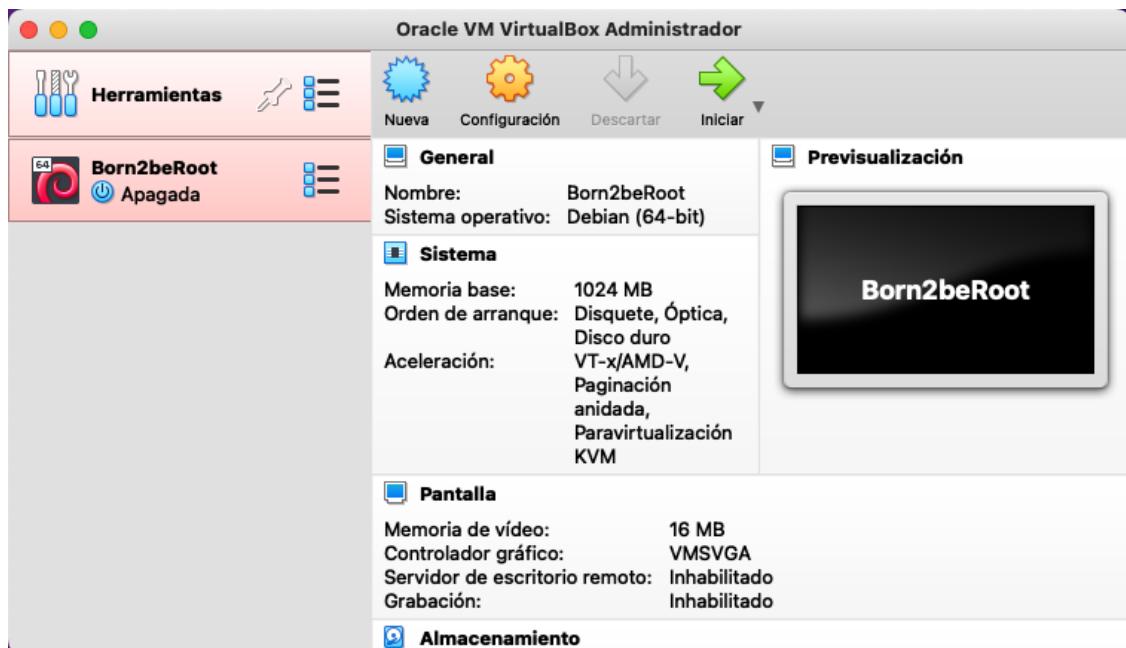


4. De vuelta a la pantalla anterior (Almacenamiento), pulsamos sobre '**Aceptar**'

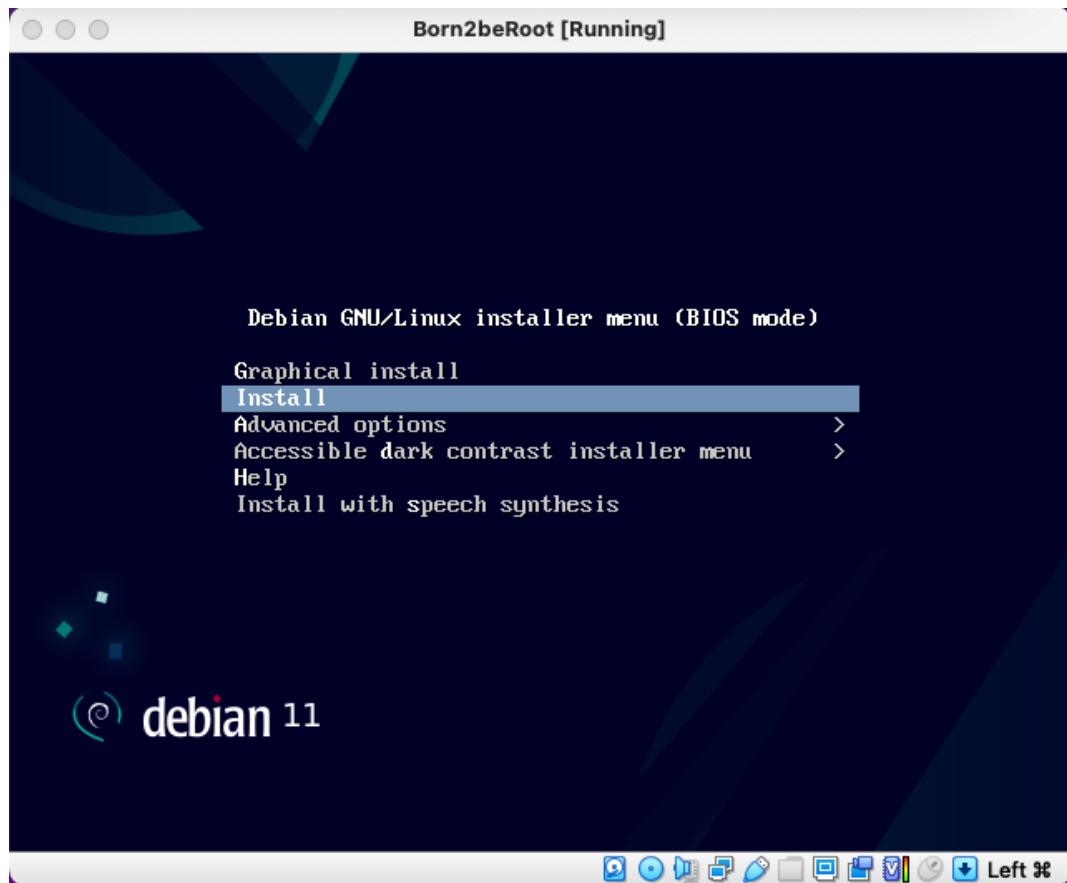


▼ Instalación de Debian

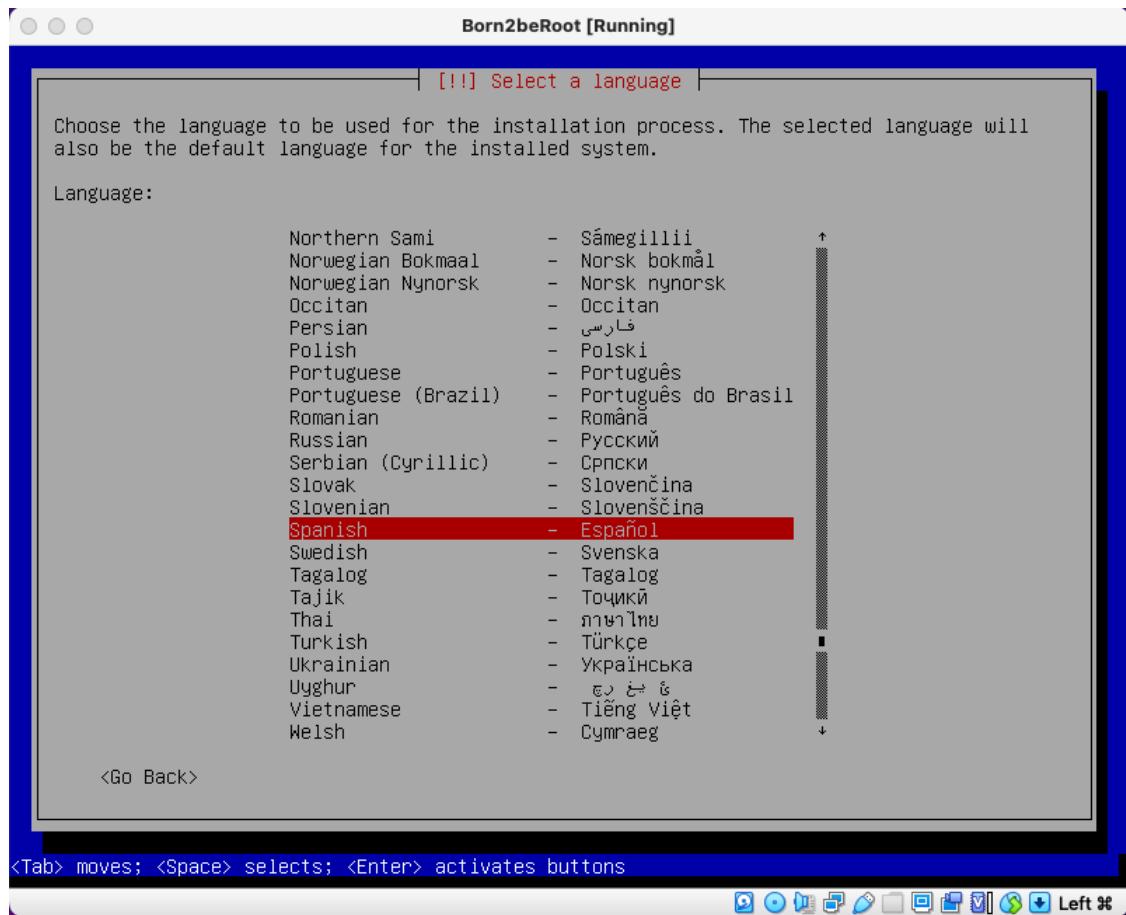
1. En el Administrador de VirtualBox, pulsamos sobre 'Iniciar'



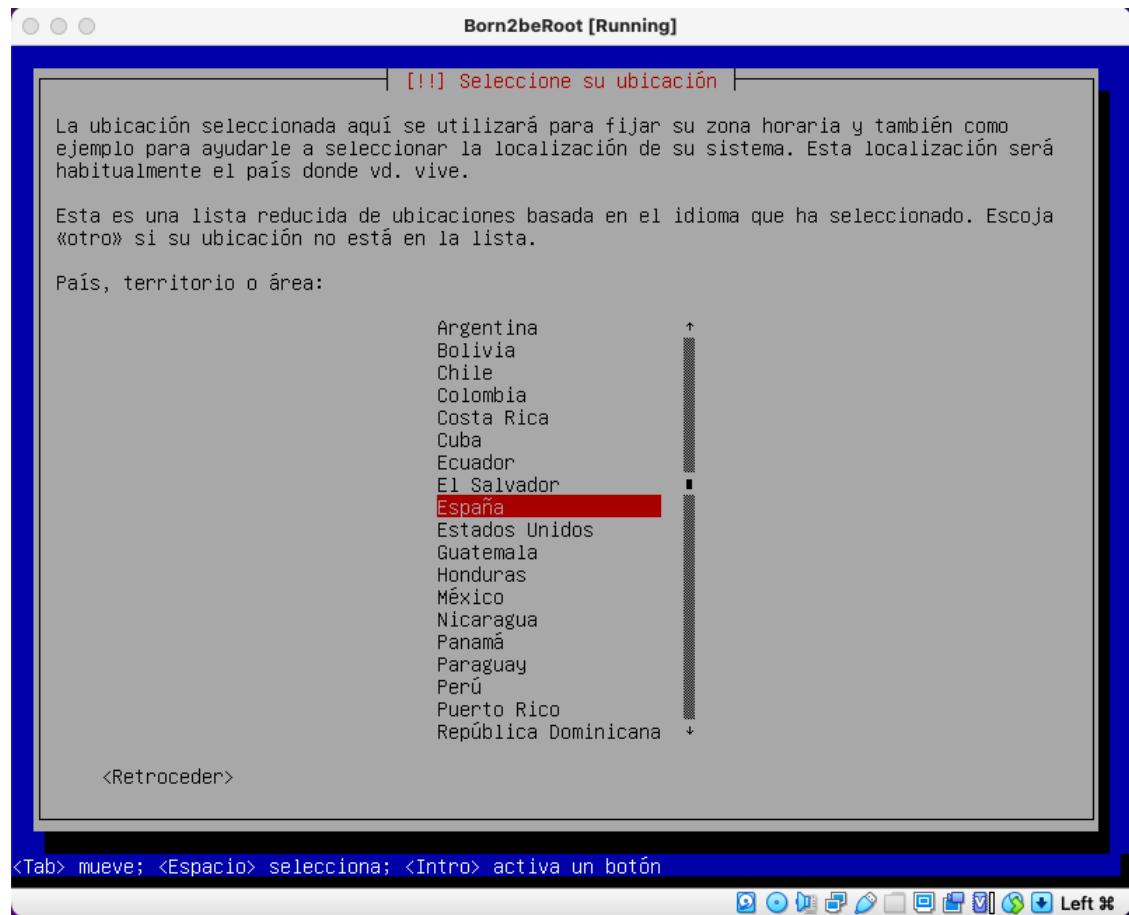
2. Esto iniciará la instalación del sistema. En la primera pantalla, escogemos 'Install'



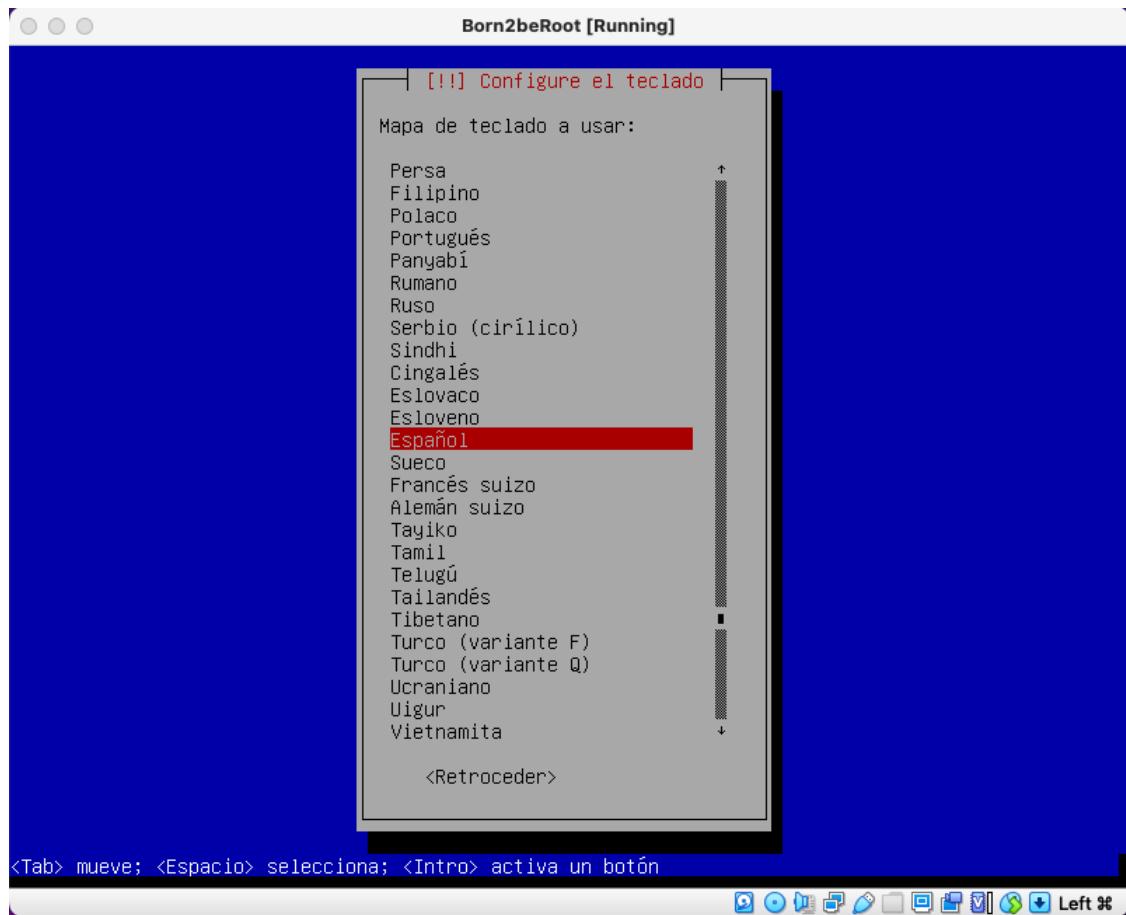
3. En la pantalla ‘Seleccione un idioma’, escogemos el deseado y pulsamos <Enter>



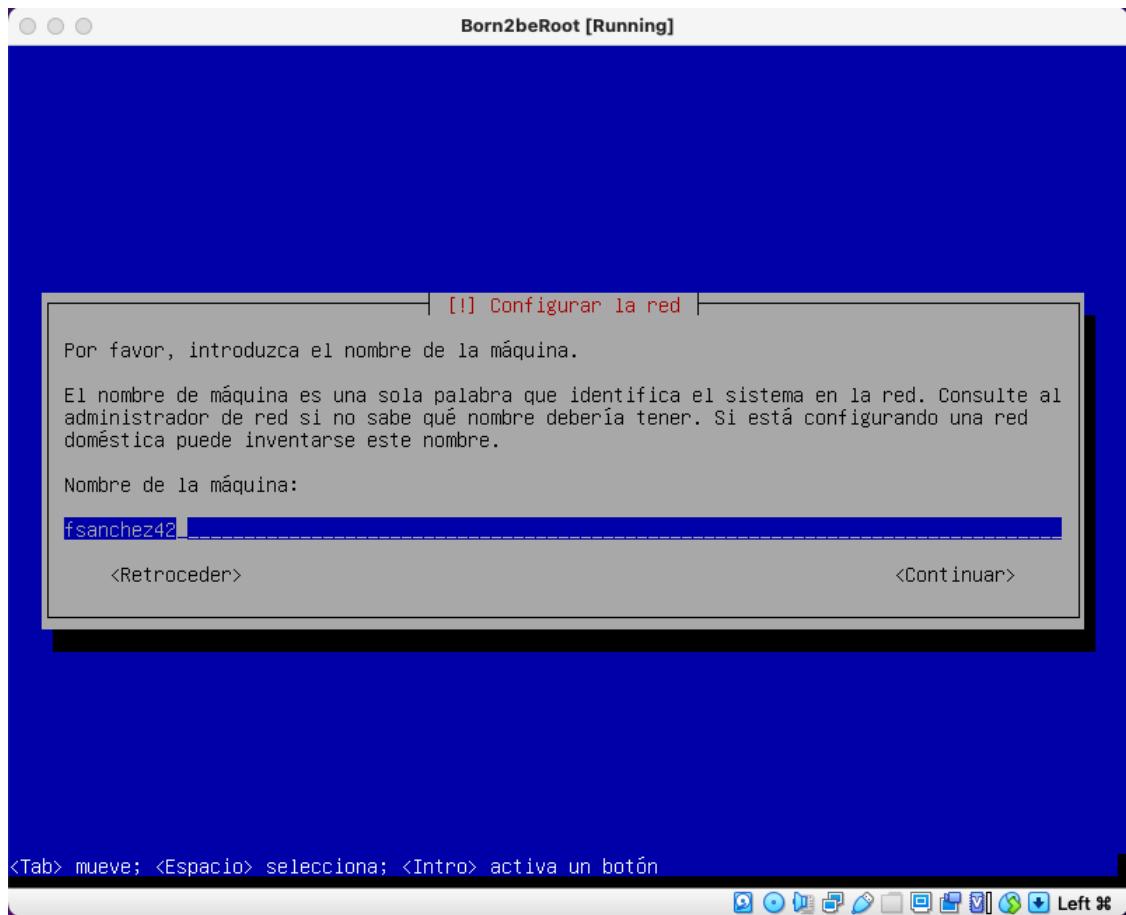
4. En la pantalla '**Seleccione su ubicación**', escogemos la deseada y pulsamos **<Enter>**



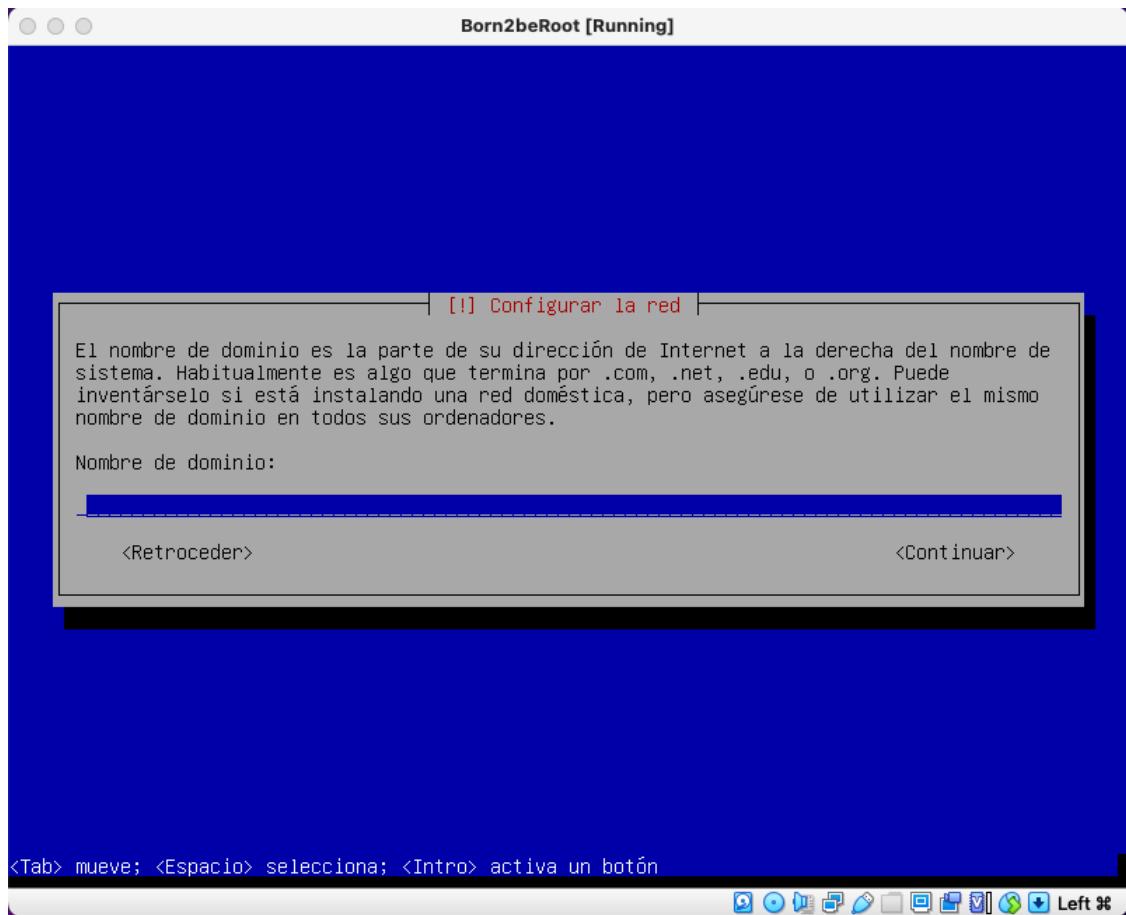
5. En la pantalla de 'Configure el teclado', elegimos la disposición deseada y pulsamos <Enter>



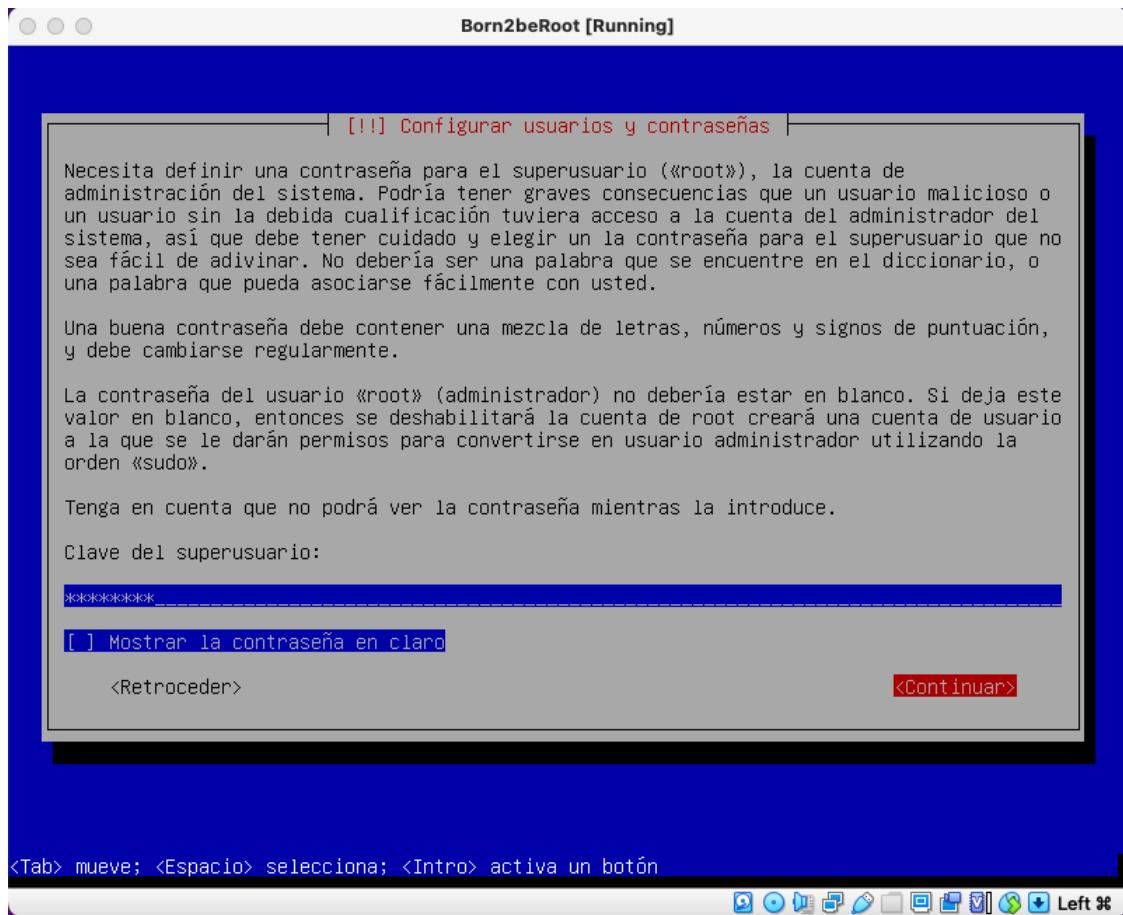
6. En la pantalla '**Configurar la red**' introducimos el nombre de la máquina (Hostname) tal y como nos indica el ejercicio '**your_login42**' y luego pulsamos Enter sobre la opción [**<Continuar>**](#)



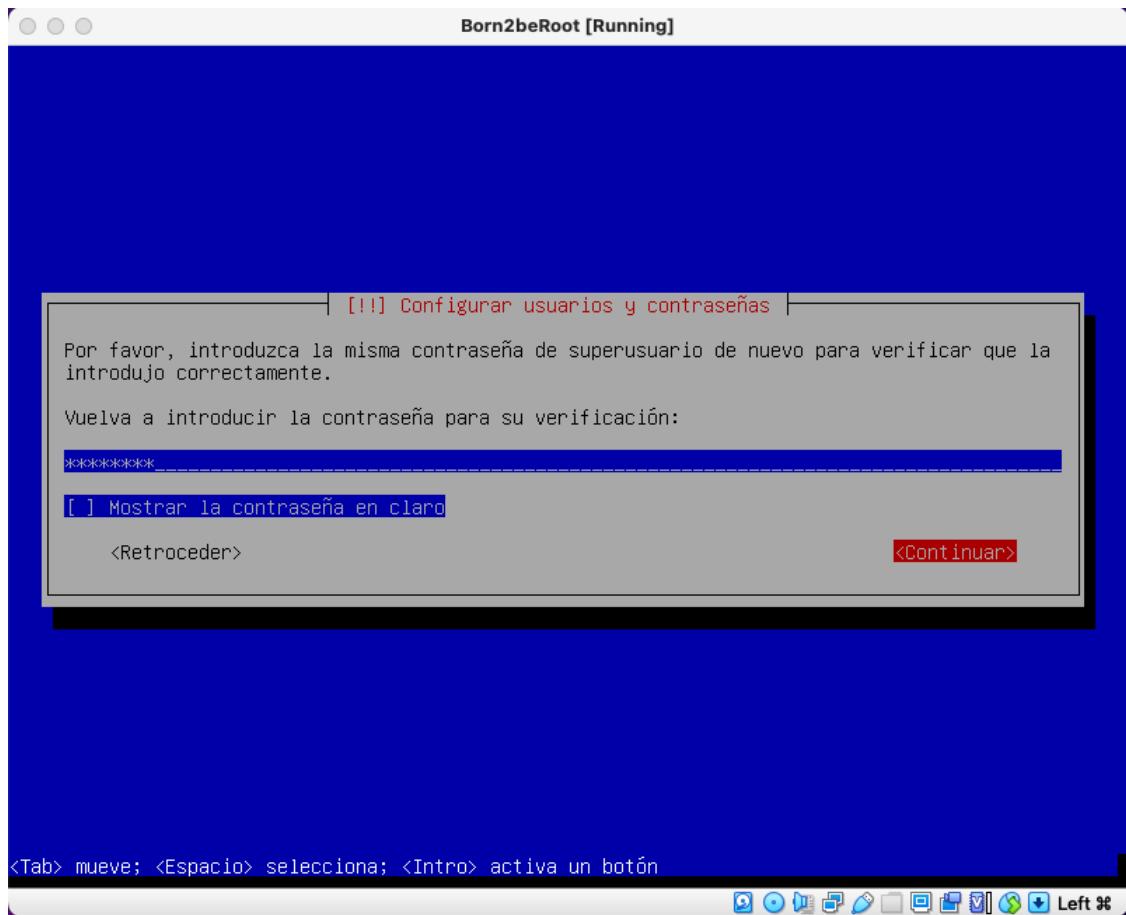
7. En la segunda pantalla '**Configurar la red**' dejamos en blanco el Nombre del dominio y seleccionamos **<Continuar>**



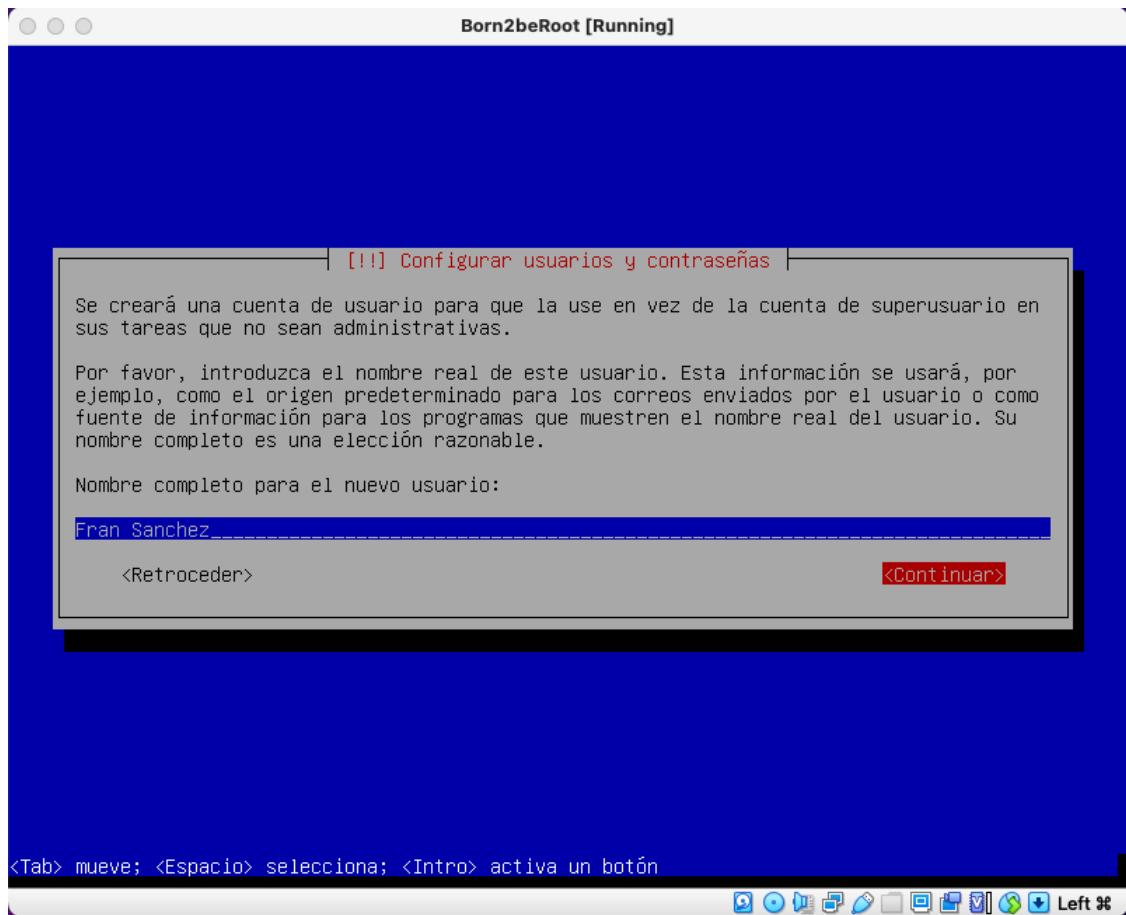
8. En la pantalla '**Configurar usuarios y contraseñas**' introducimos la contraseña deseada para el usuario root (P@ssw0rD) y ejecutamos la opción <**Continuar**>



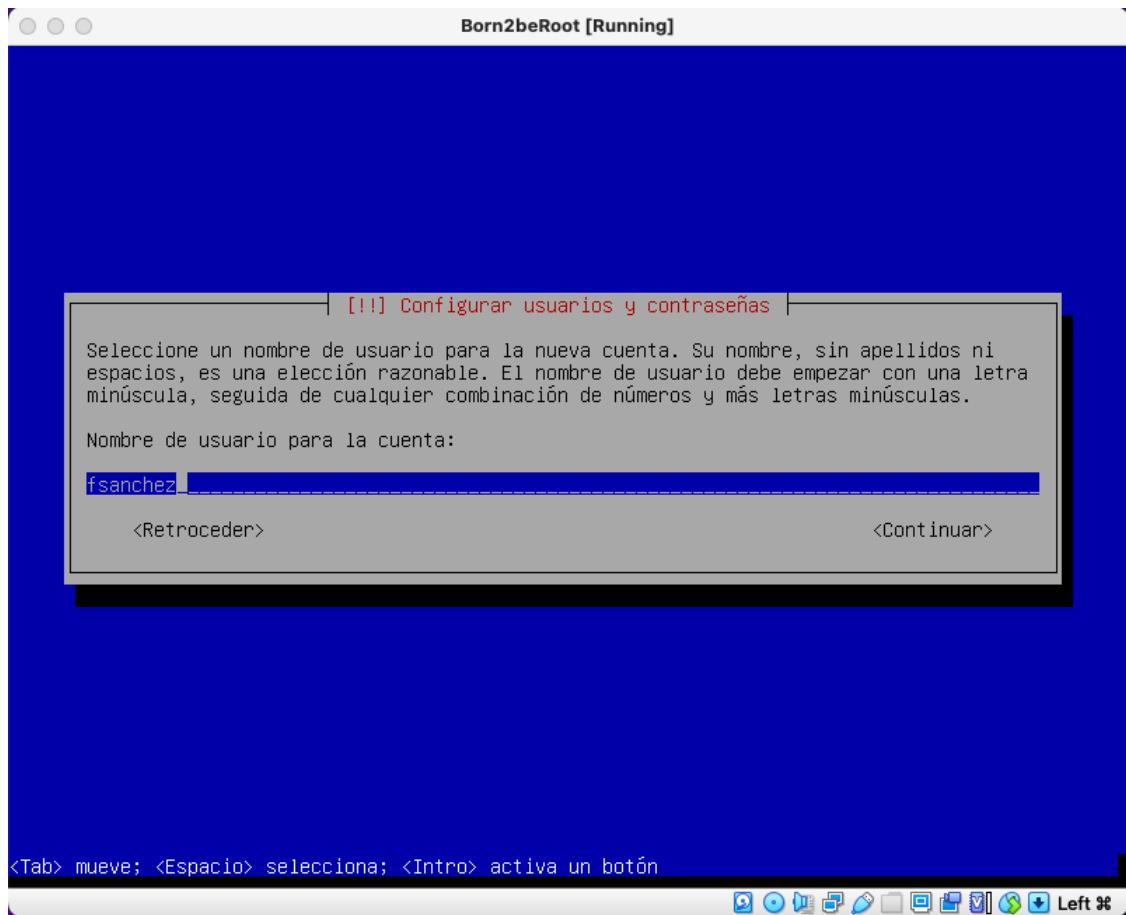
9. En la siguiente pantalla, volvemos a escribir la contraseña del root y ejecutamos [<Continuar>](#)



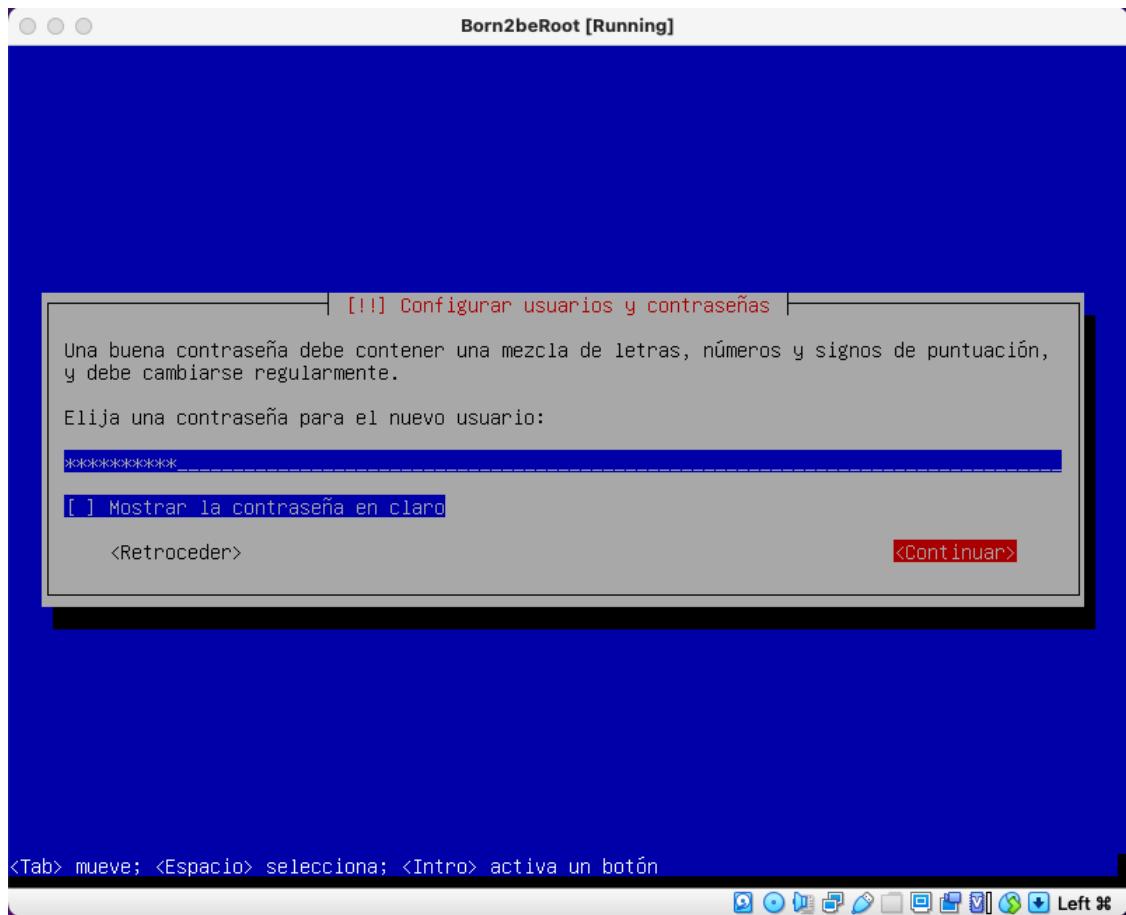
10. Escribimos a continuación el nombre completo de nuestro usuario y pulsamos <Continuar>



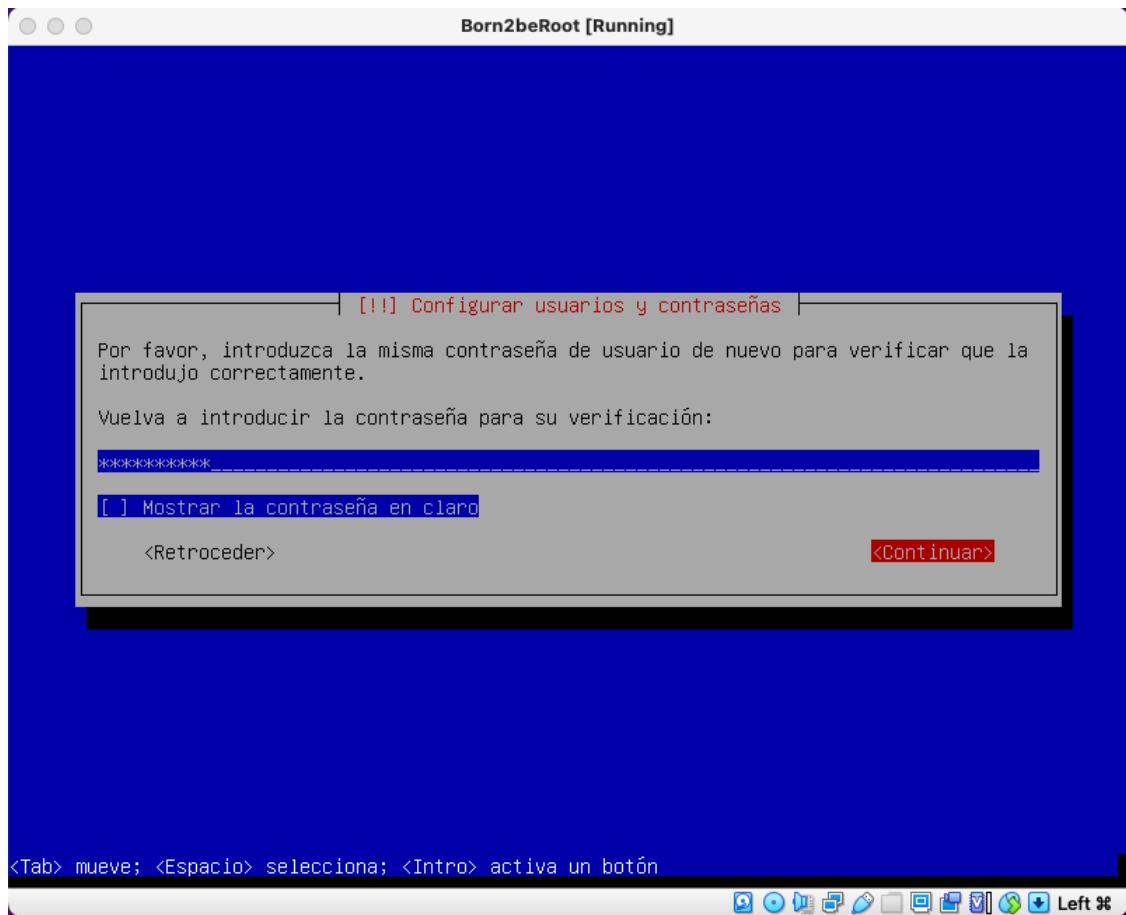
11. Introducimos nuestro login en 42 y escogemos la opción **<Continuar>**



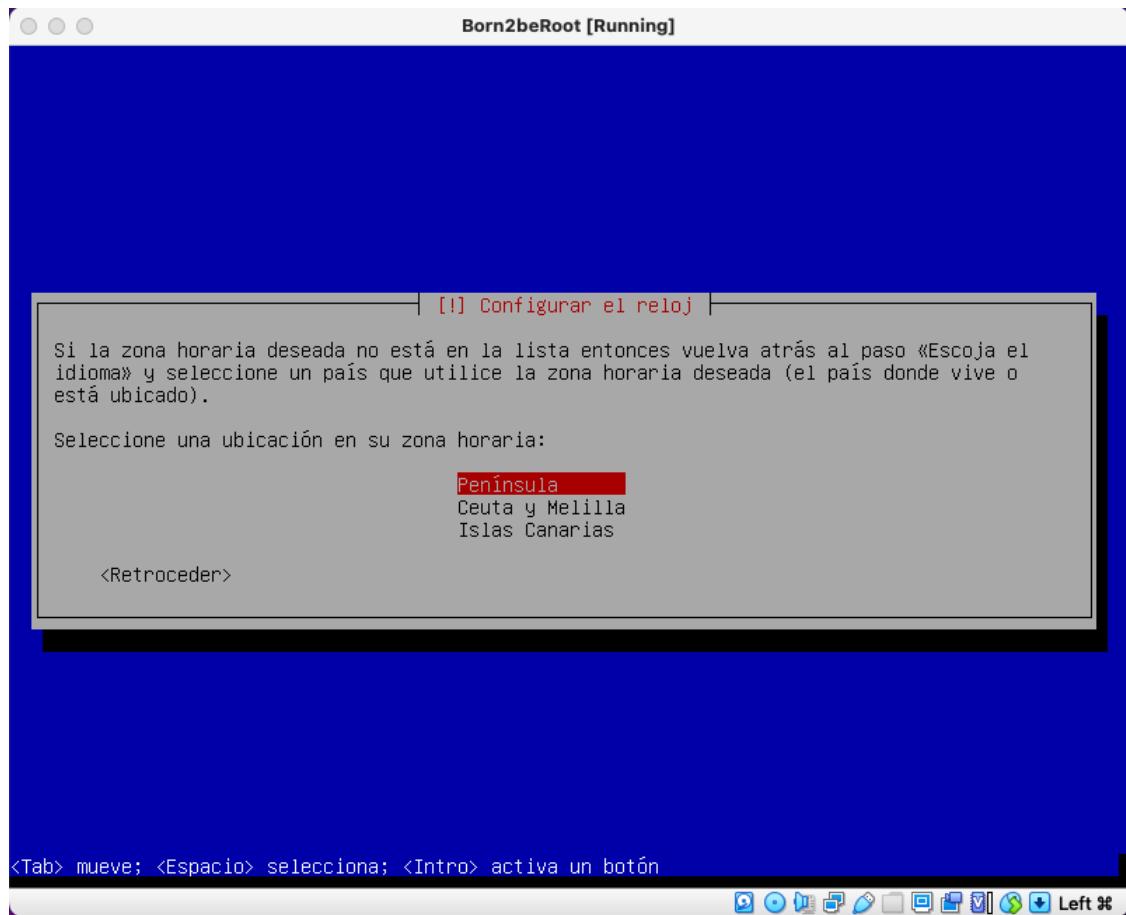
12. Elegimos una contraseña (P@sw0rDfs) para nuestro usuario y damos a [**<Continuar>**](#)



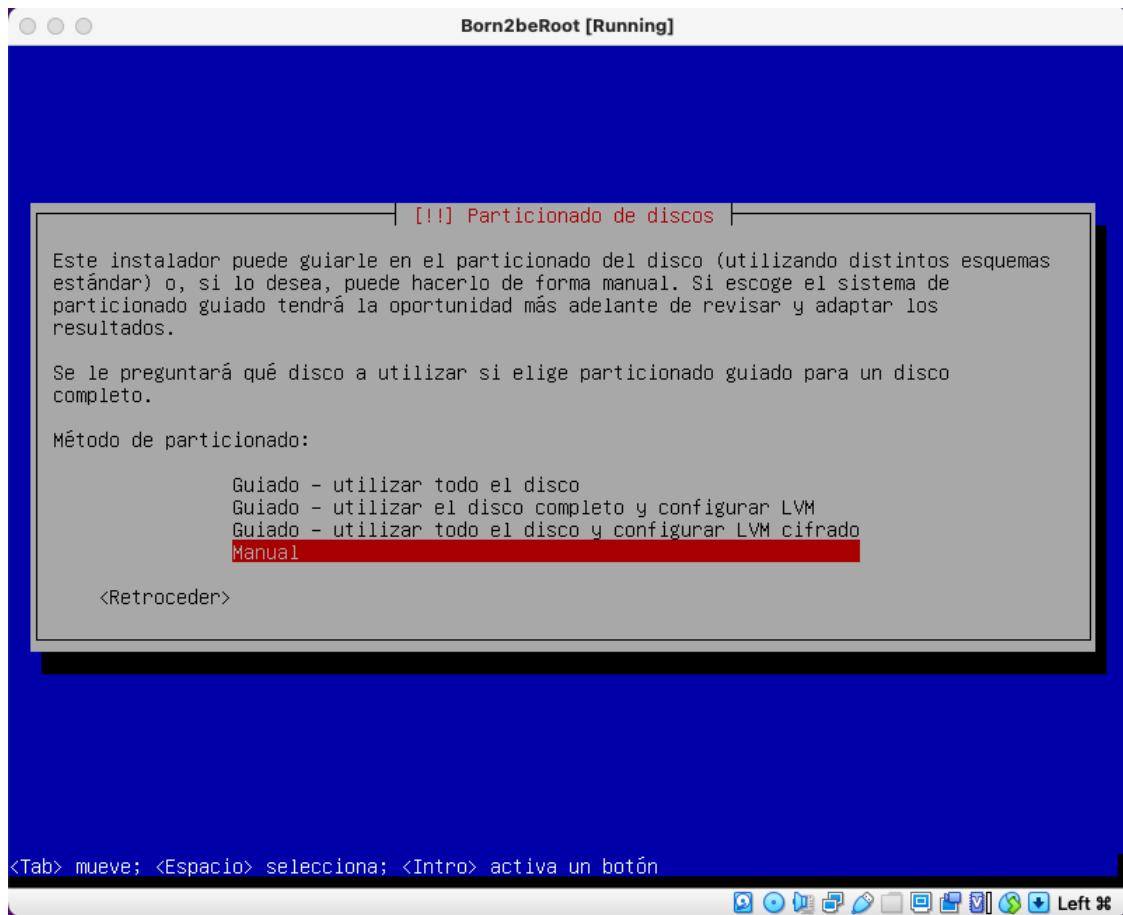
13. Volvemos a introducir la contraseña elegida para nuestro usuario y damos a <Continuar>



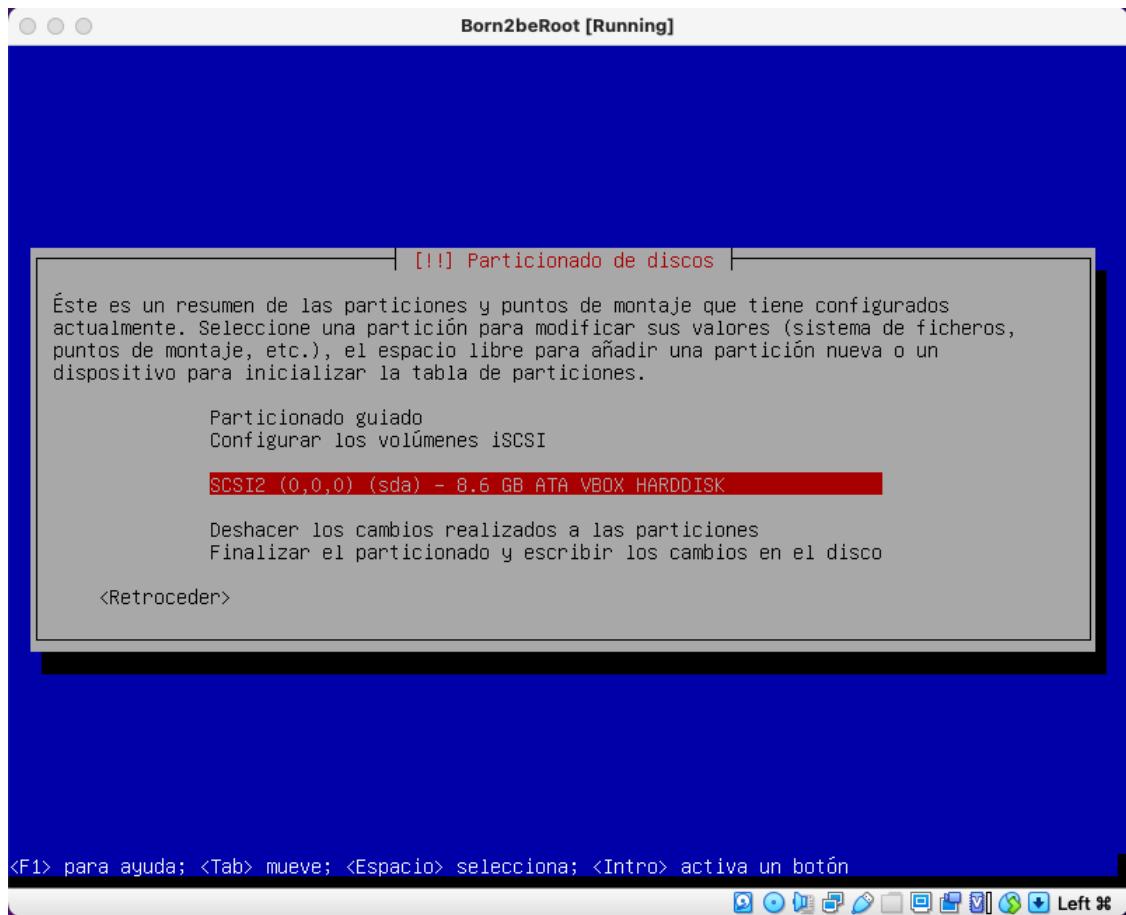
14. Si nos aparece la pantalla '**Configurar el reloj**', seleccionamos la opción adecuada y pulsamos **<Enter>**



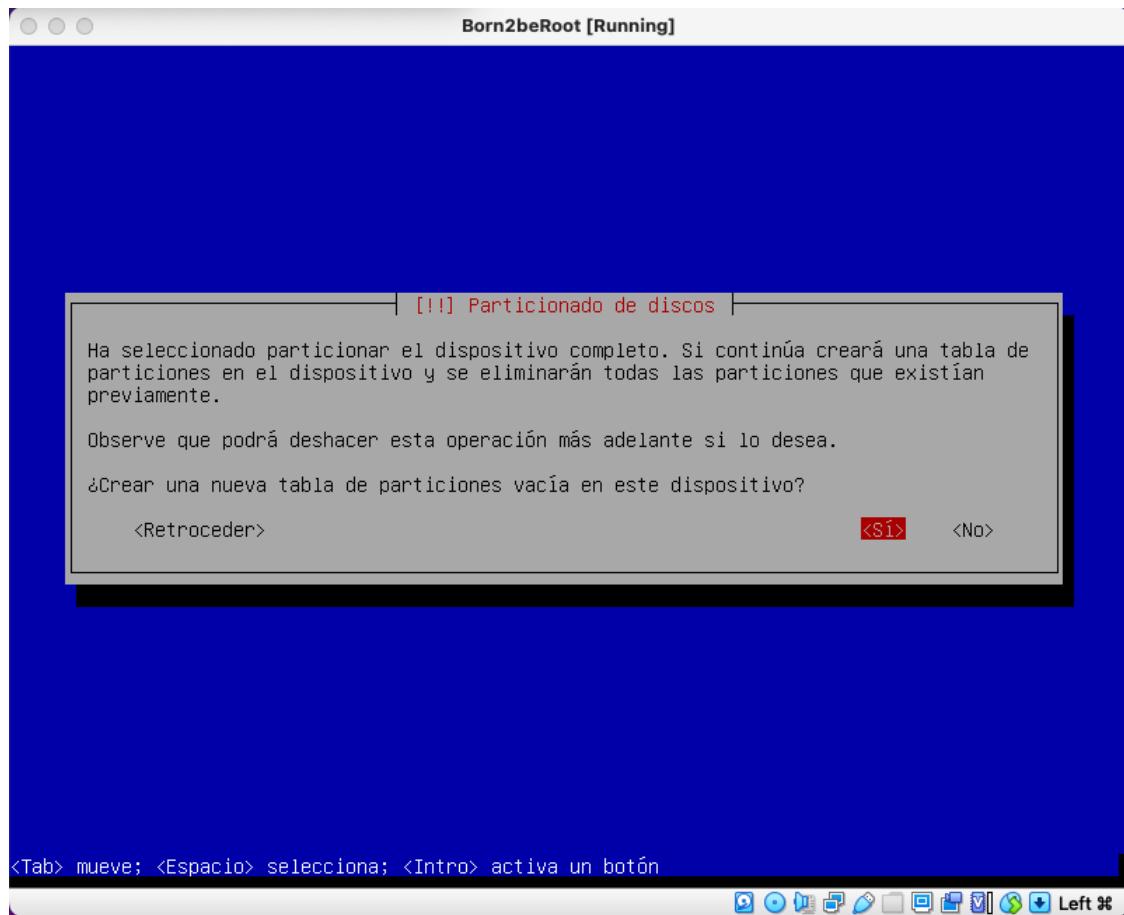
15. En la pantalla '**Particionado de discos**' seleccionamos '**Manual**' y pulsamos **<Enter>**



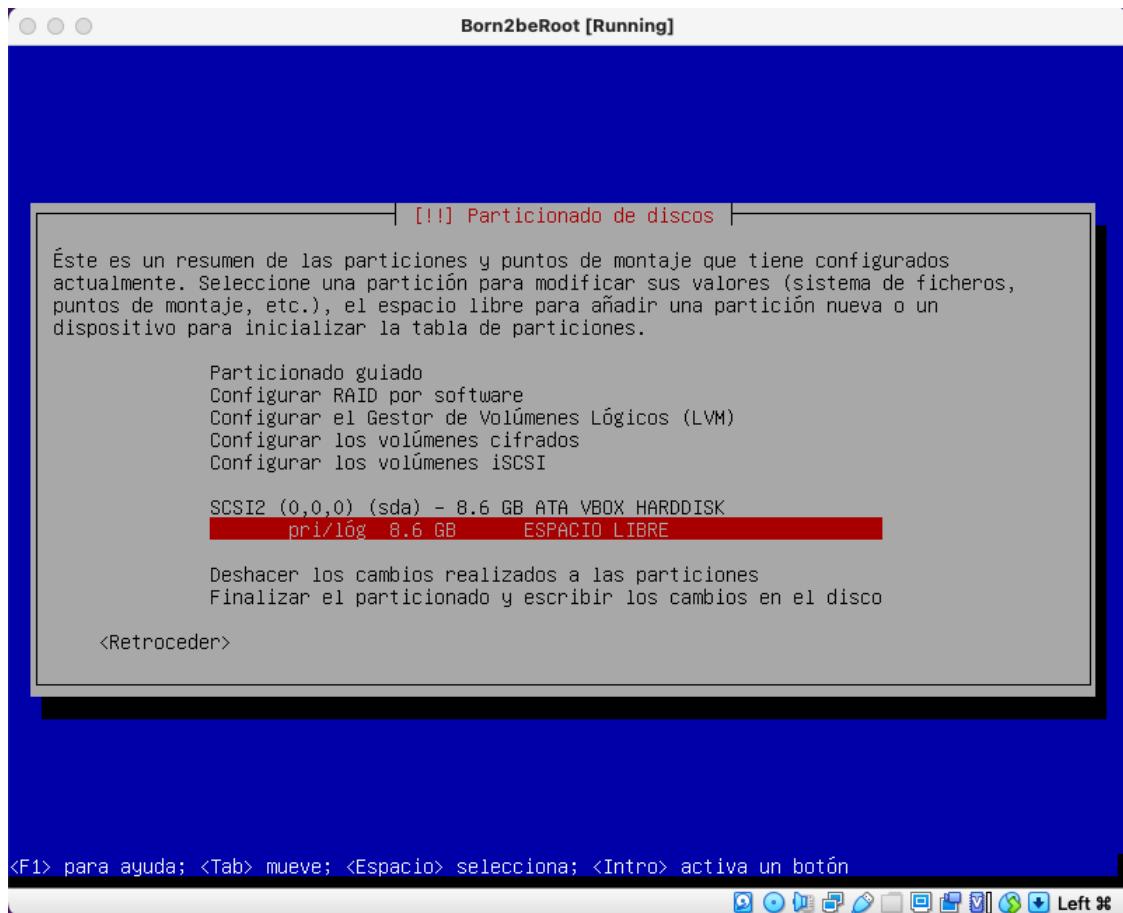
16. En la siguiente pantalla elegimos la entrada '**SCSI2 (0,0,0)...**' y pulsamos **<Enter>**



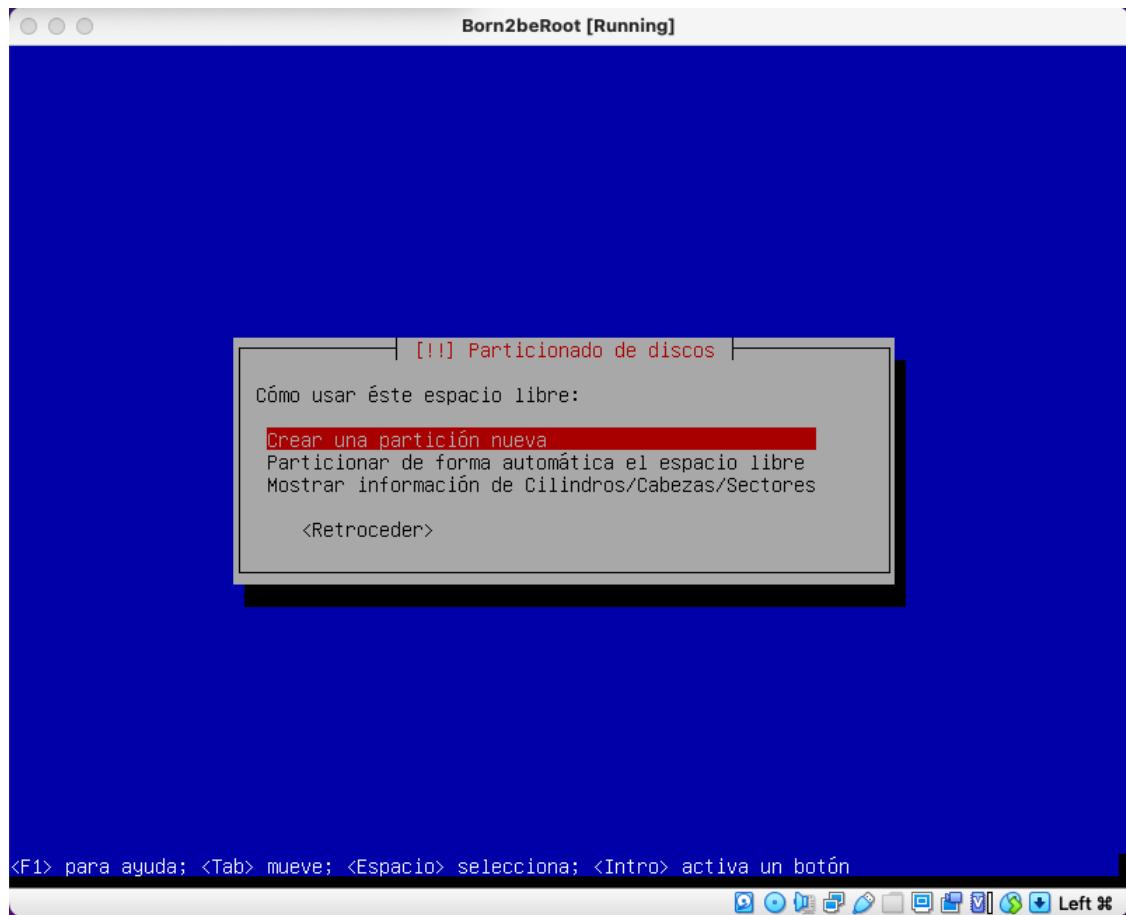
17. Elegimos **<Sí>** crear una nueva tabla de particiones vacía en el dispositivo y pulsamos **<Enter>**



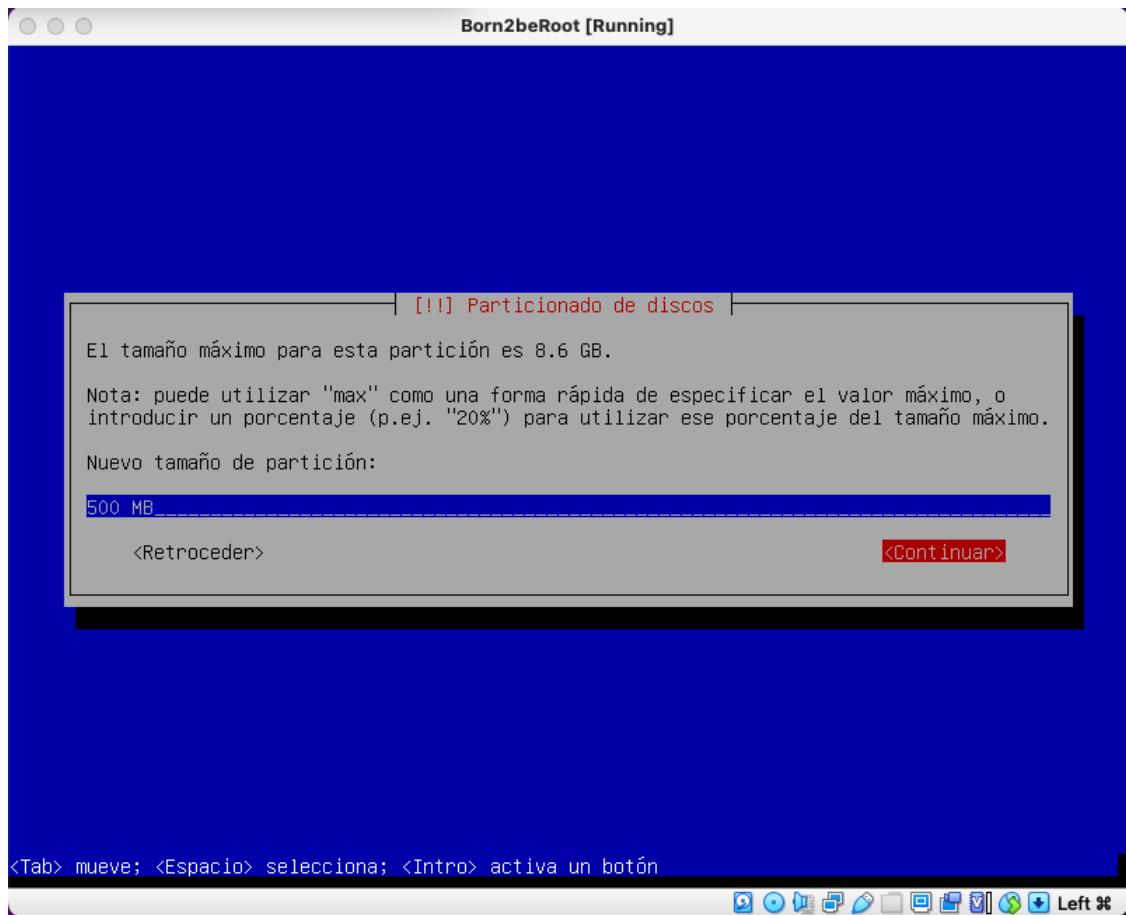
18. A continuación elegimos la entrada '**pri|lób 8.6 GB... ESPACIO LIBRE**' y pulsamos **<Enter>**



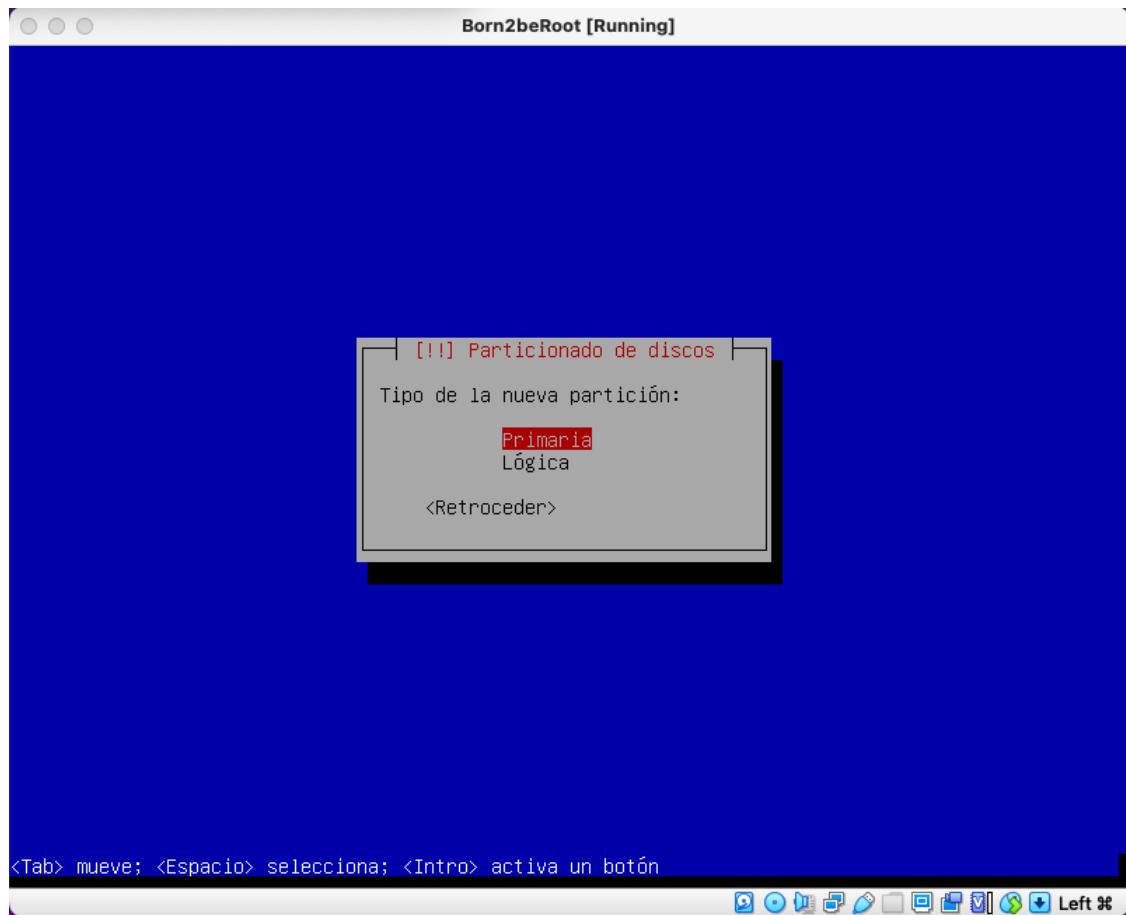
19. Elegimos 'Crear una partición nueva' y pulsamos <Enter>



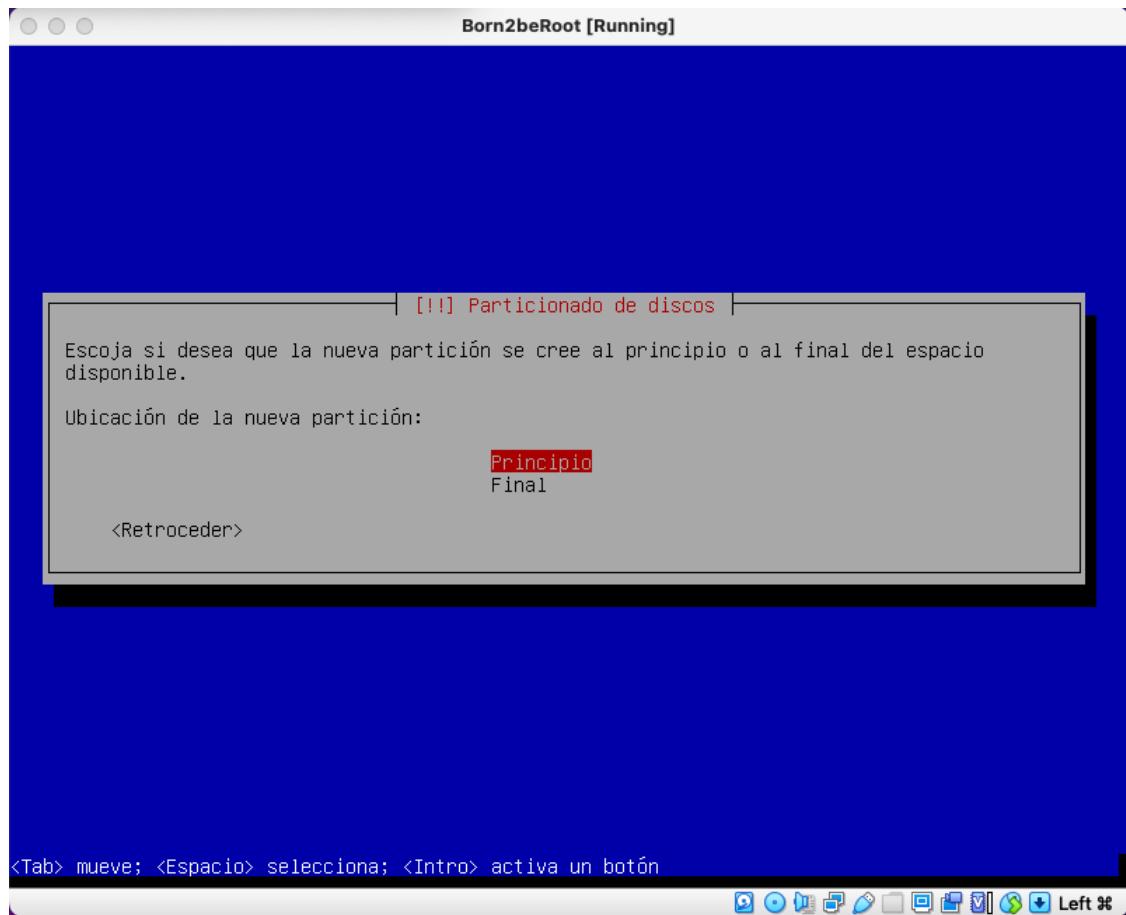
20. Establecemos el Nuevo tamaño de partición en '**500 MB**' y pulsamos **<Continuar>**



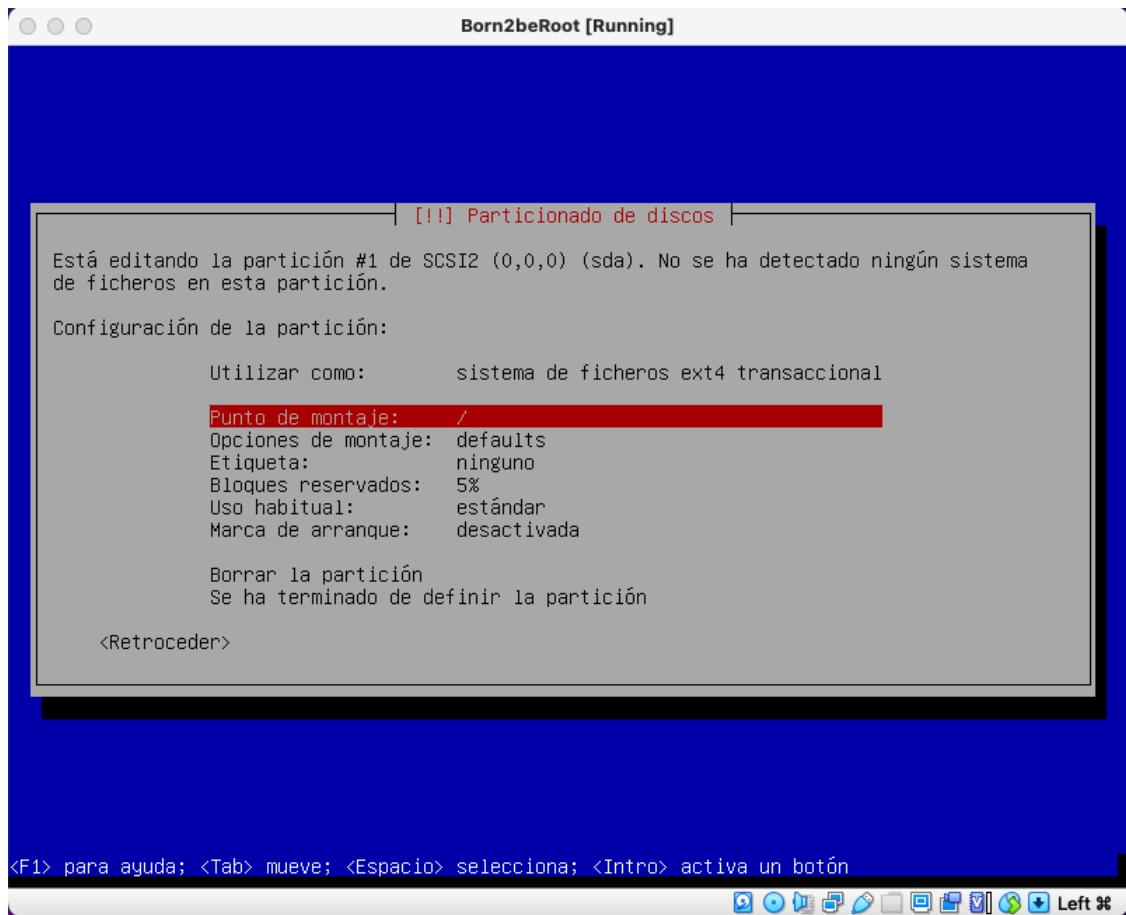
21. El tipo de la nueva partición será '**Primaria**' y pulsamos **<Enter>**



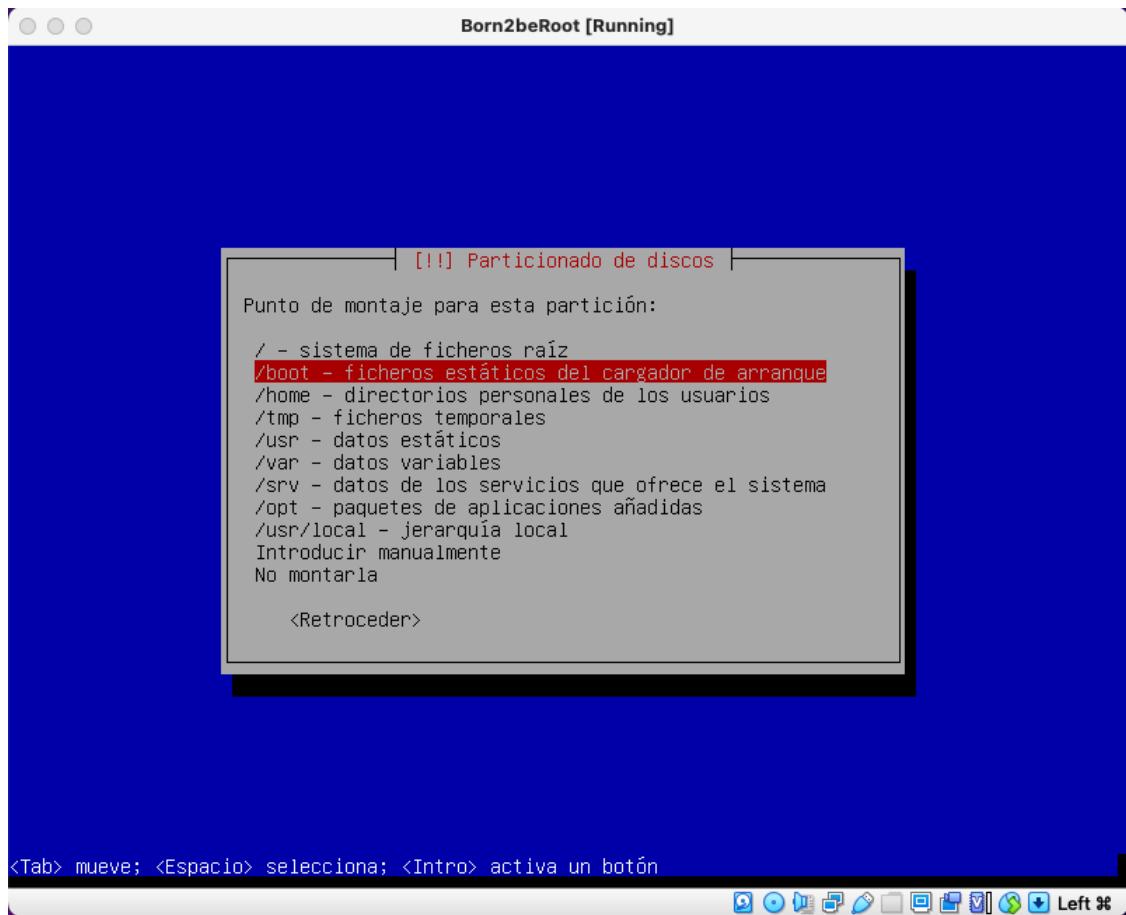
22. Ubicaremos la nueva partición al '**Principio**' y pulsaremos **<Enter>**



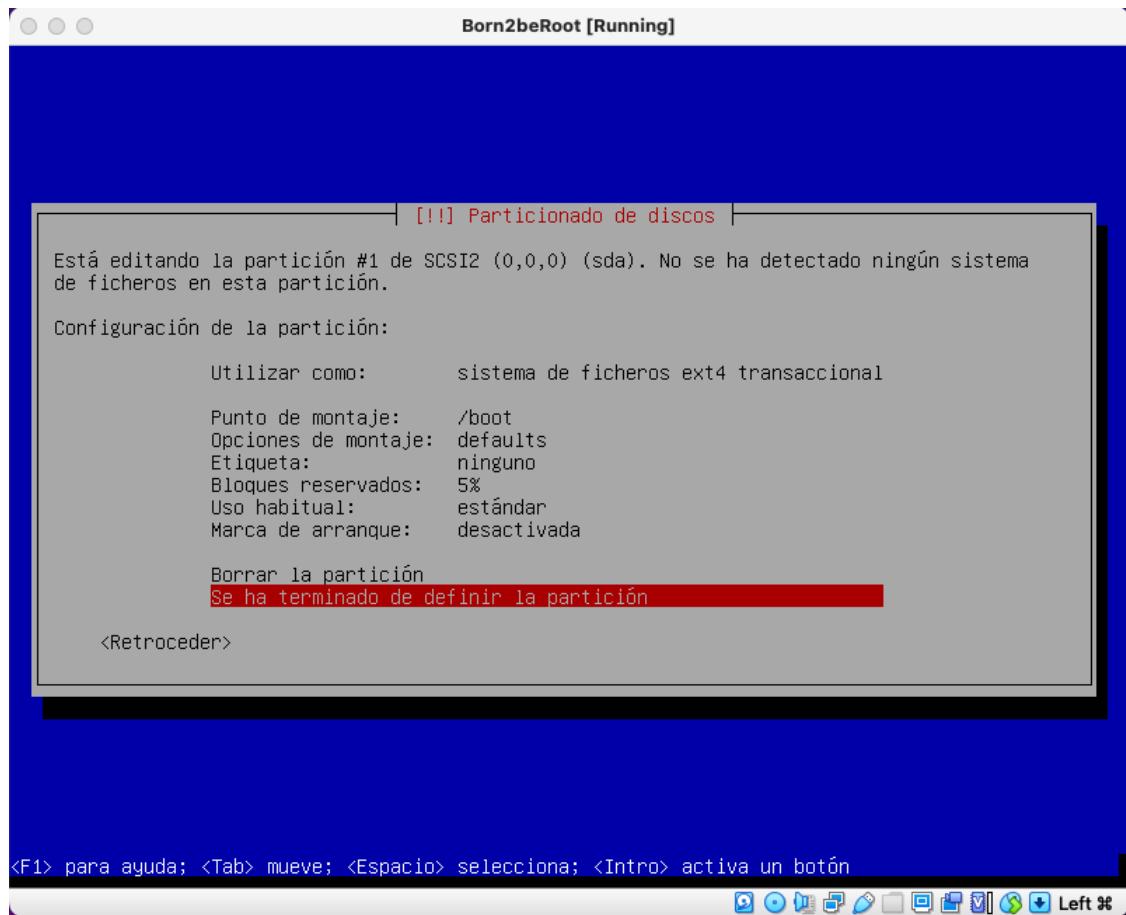
23. Nos colocamos sobre el campo '**Punto de montaje**' y pulsamos **<Enter>**



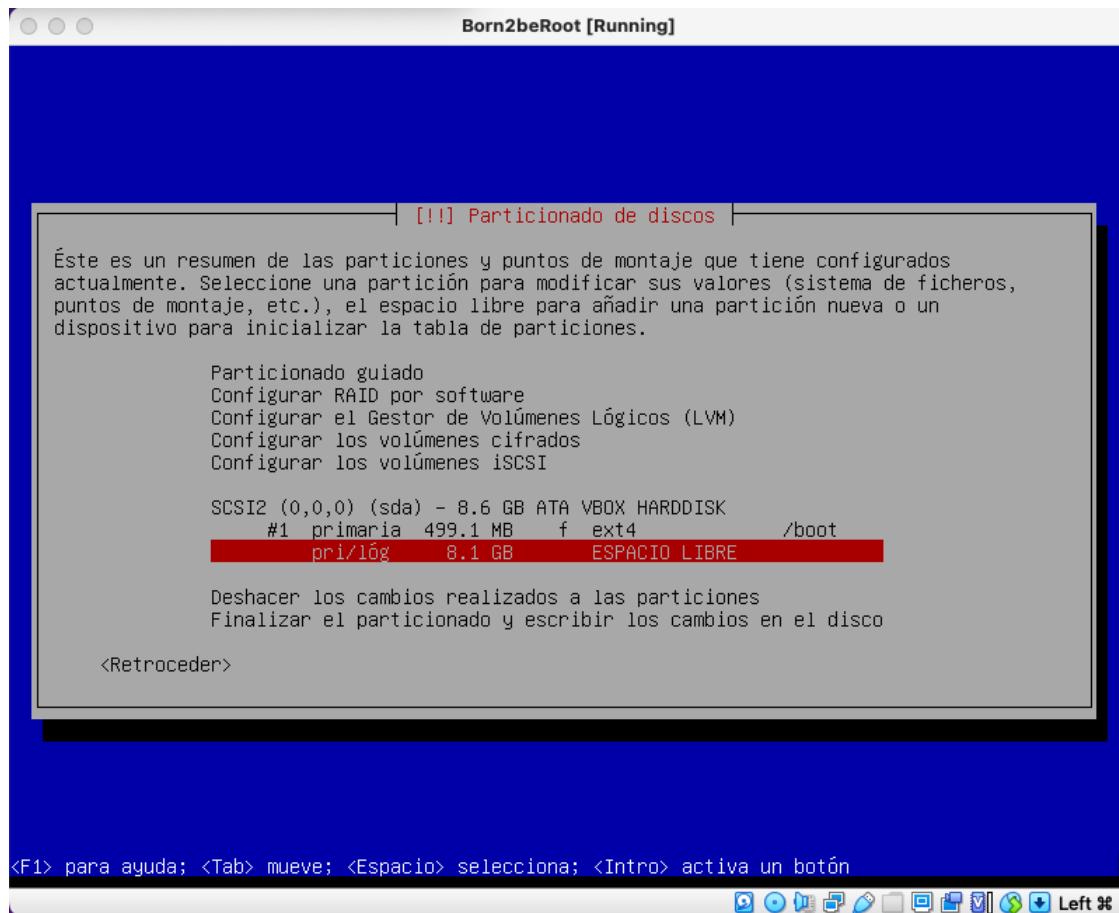
24. Marcamos '**/boot - ficheros estáticos del cargador de arranque**' y pulsamos **<Enter>**



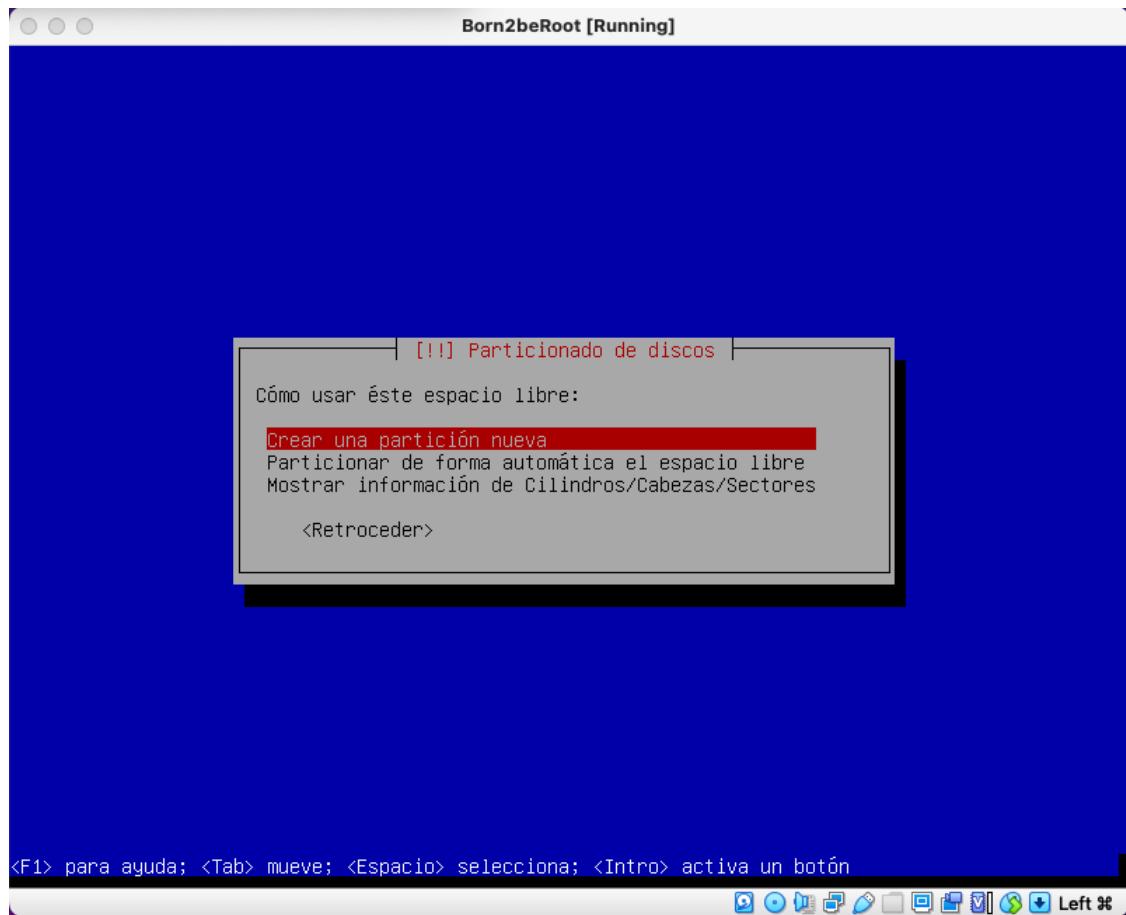
25. De vuelta a la pantalla anterior, seleccionamos '**Se ha terminado de definir la partición**' y pulsamos **<Enter>**



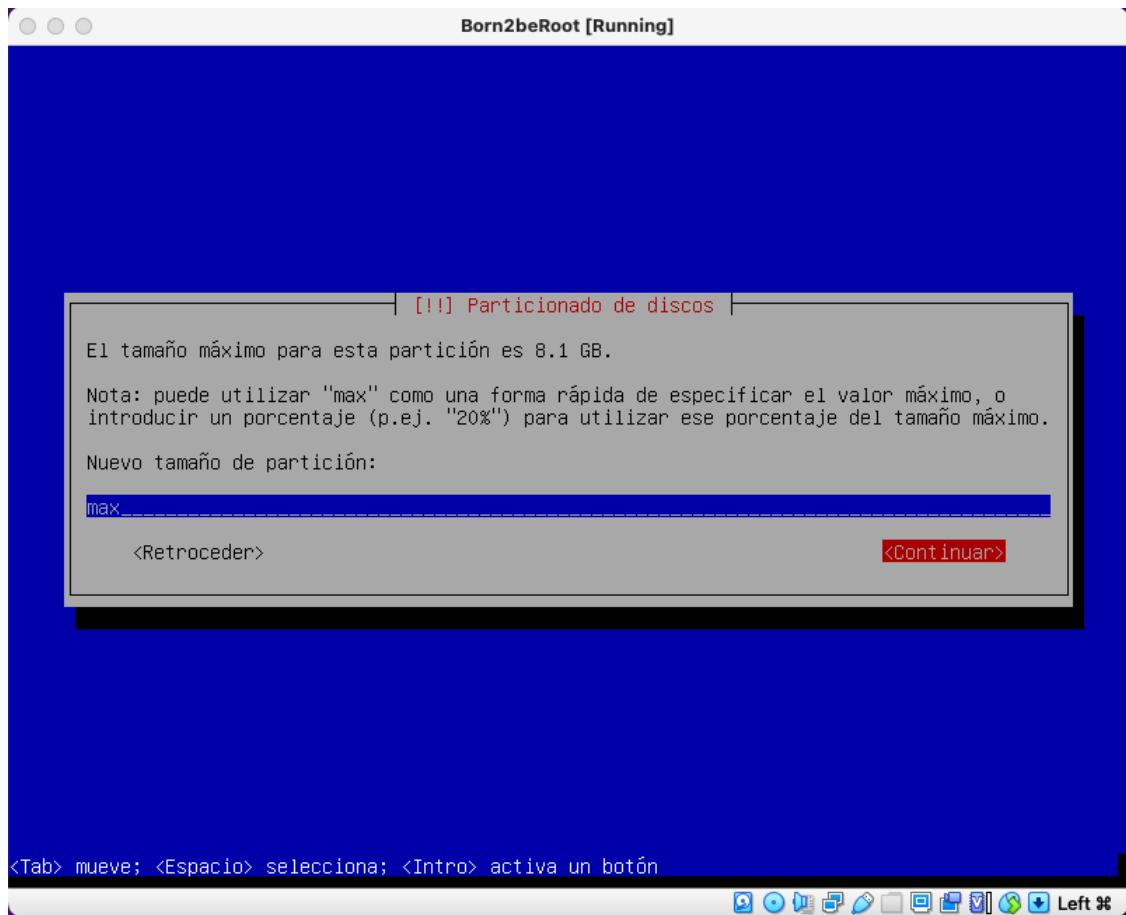
26. Volvemos a la pantalla de asignación de espacio de disco y de nuevo escogemos la entrada de **ESPACIO LIBRE**, seguido de **<Enter>**



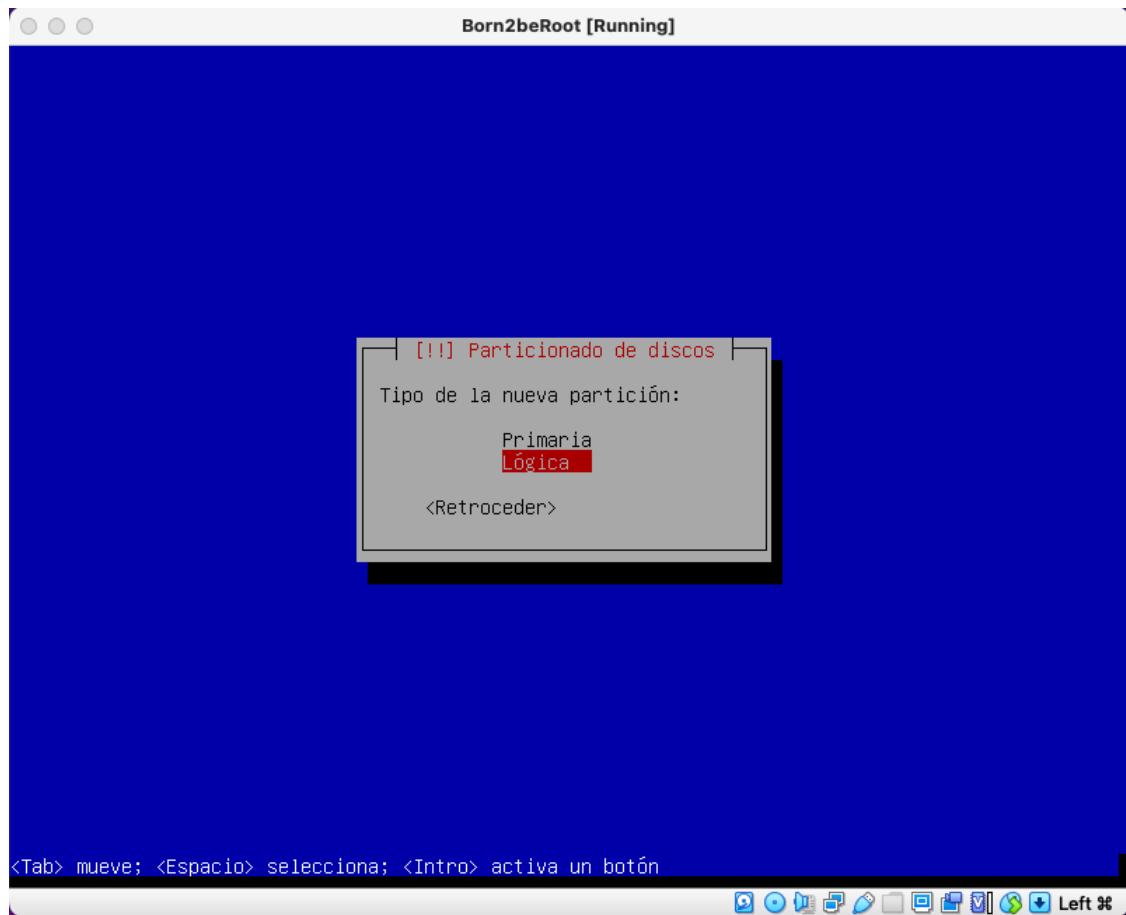
27. Creamos una partición nueva.



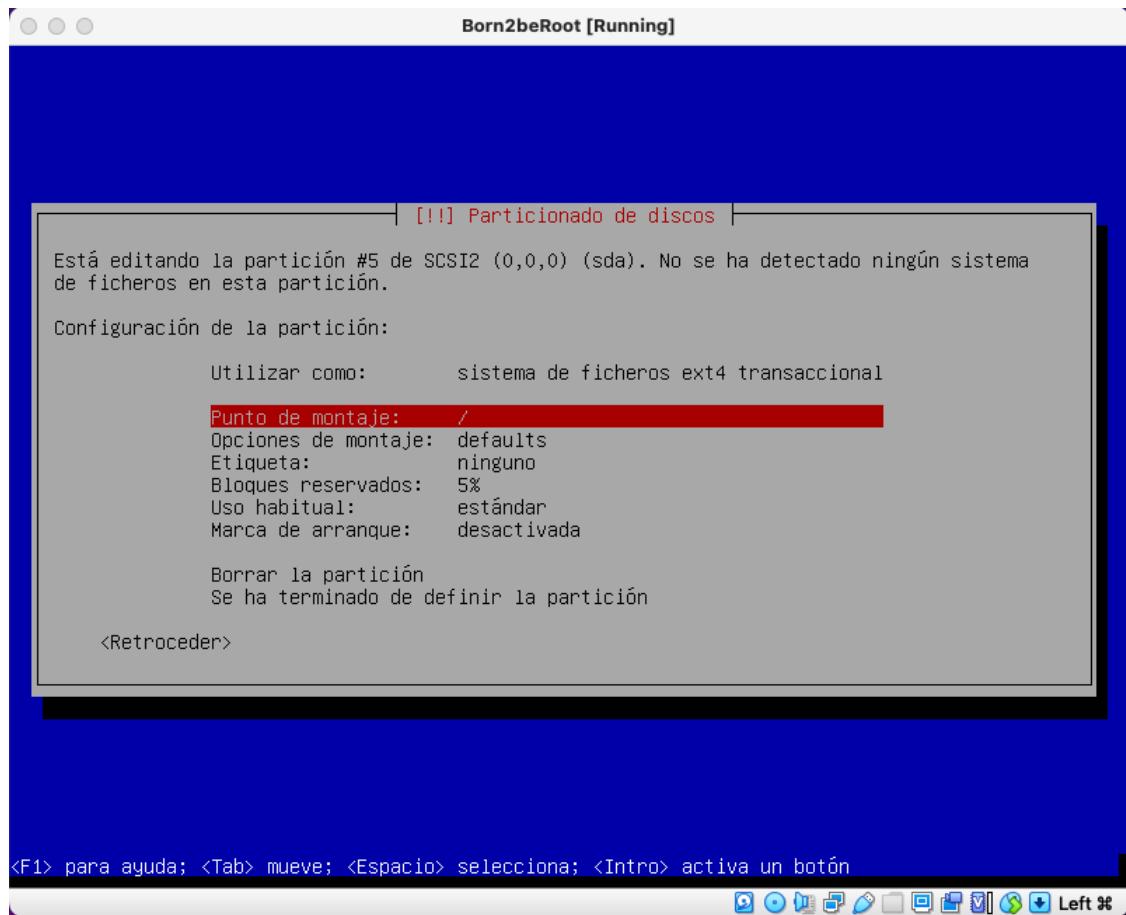
28. Y esta vez escribimos '**max**' en el apartado '**Nuevo tamaño de partición**', para utilizar todo el espacio disponible. Luego pulsamos en **<Continuar>**

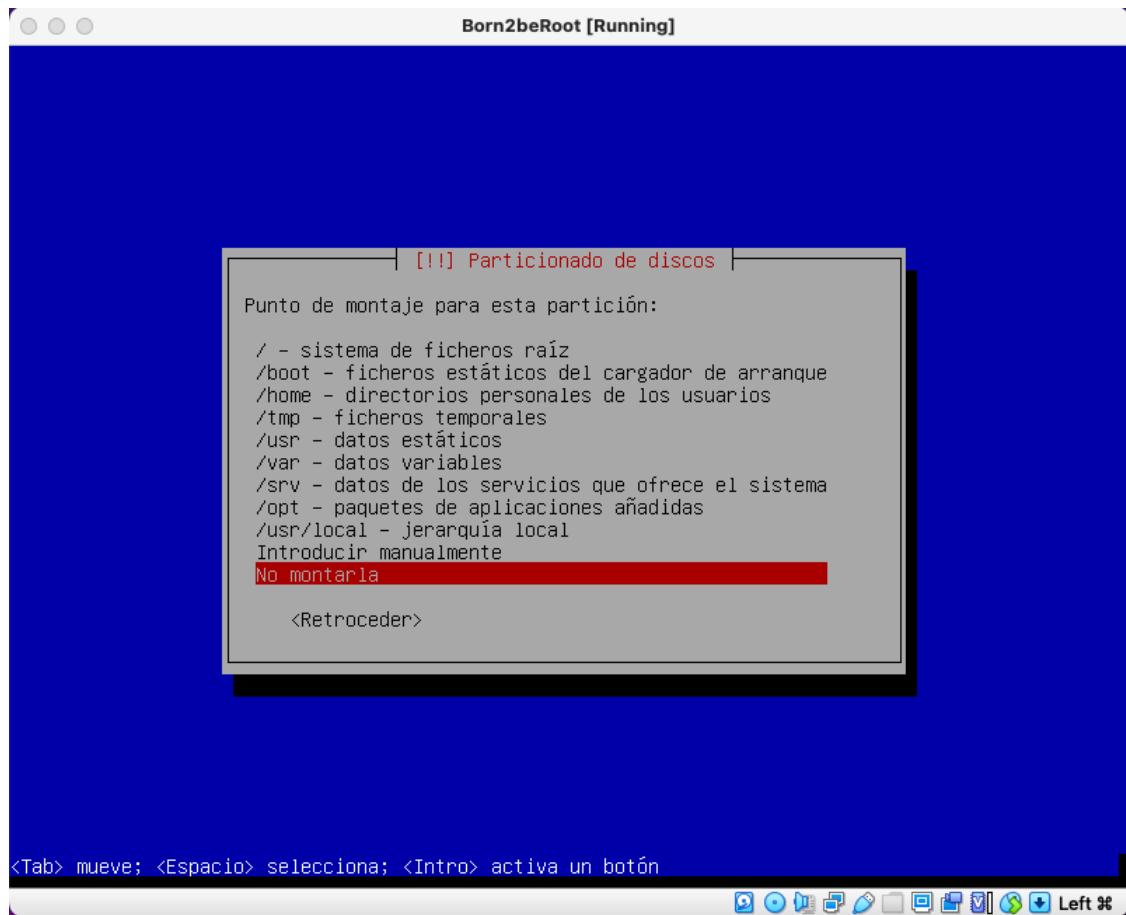


29. En este caso el tipo de la nueva partición será '**LÓGICA**'. Pulsamos **<Enter>**

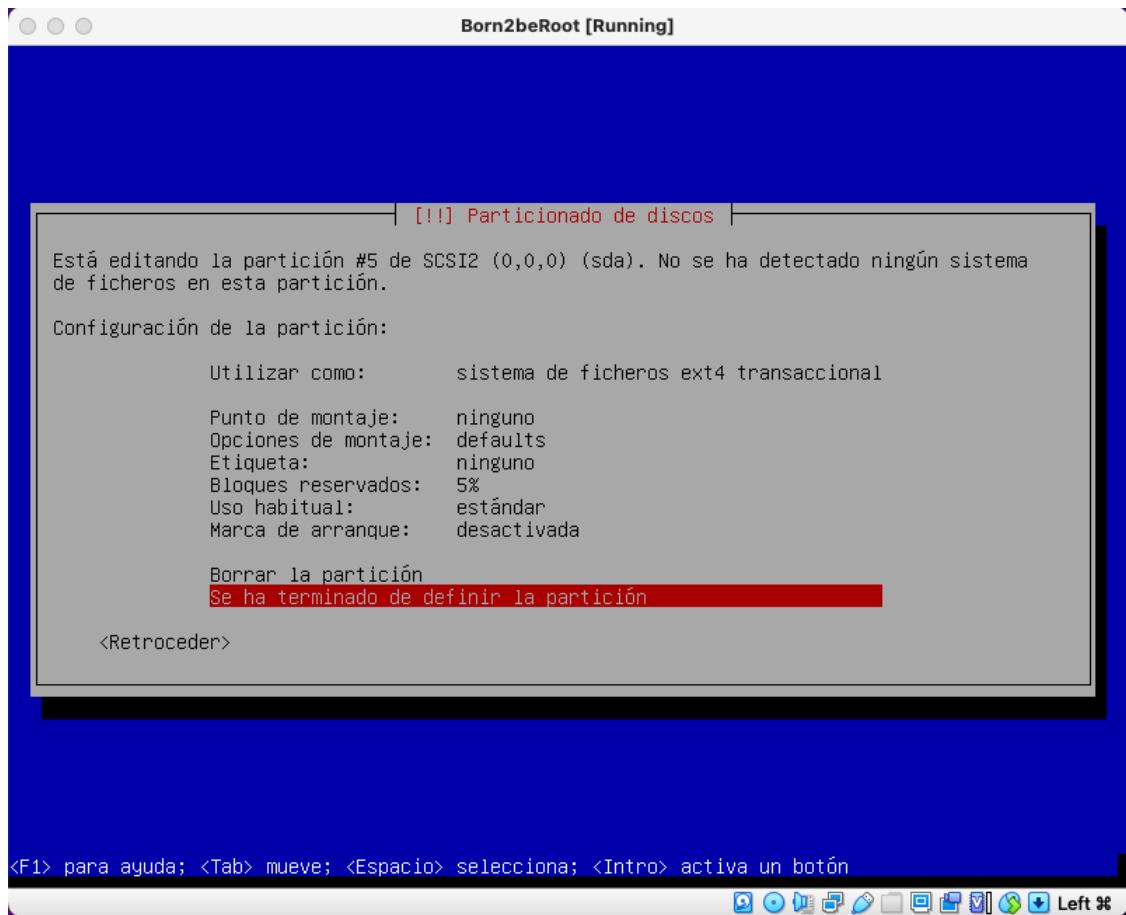


30. Cambiamos el punto de montaje y seleccionamos '**No montarla**'

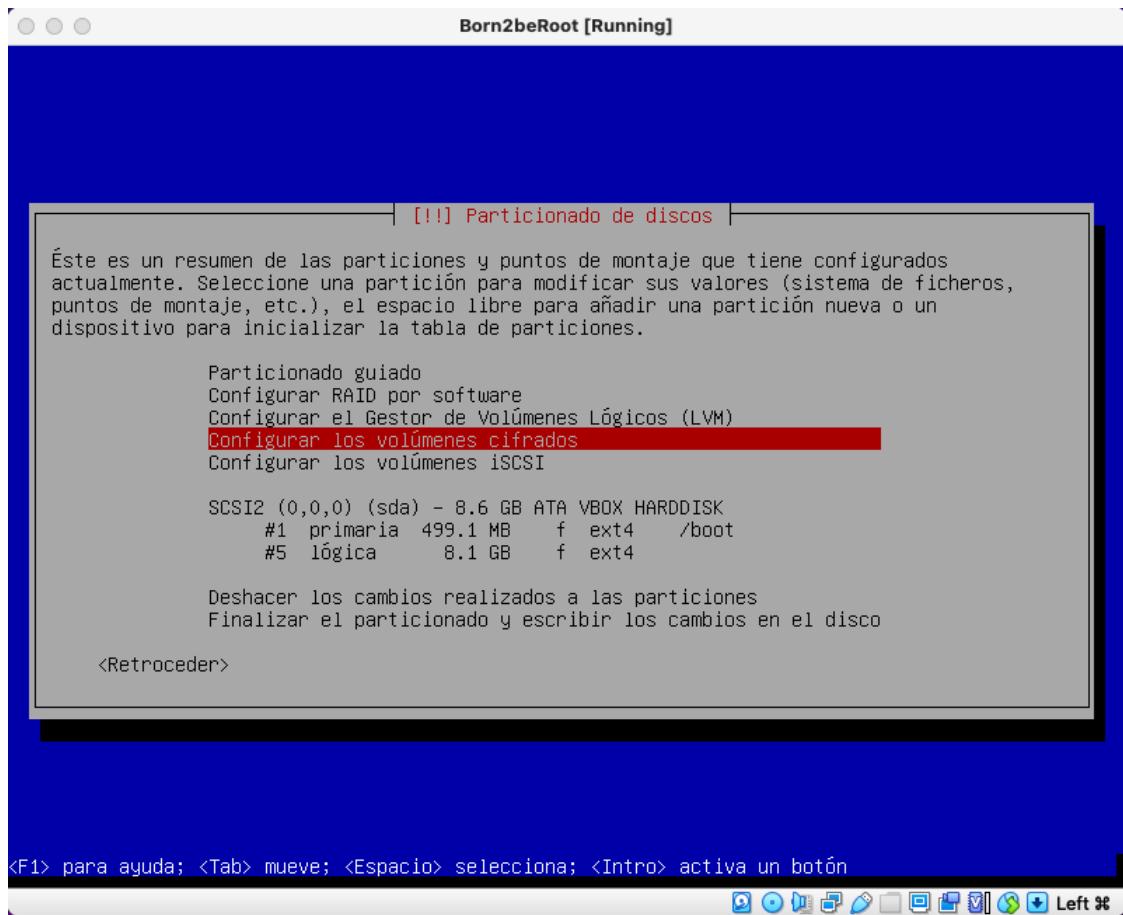




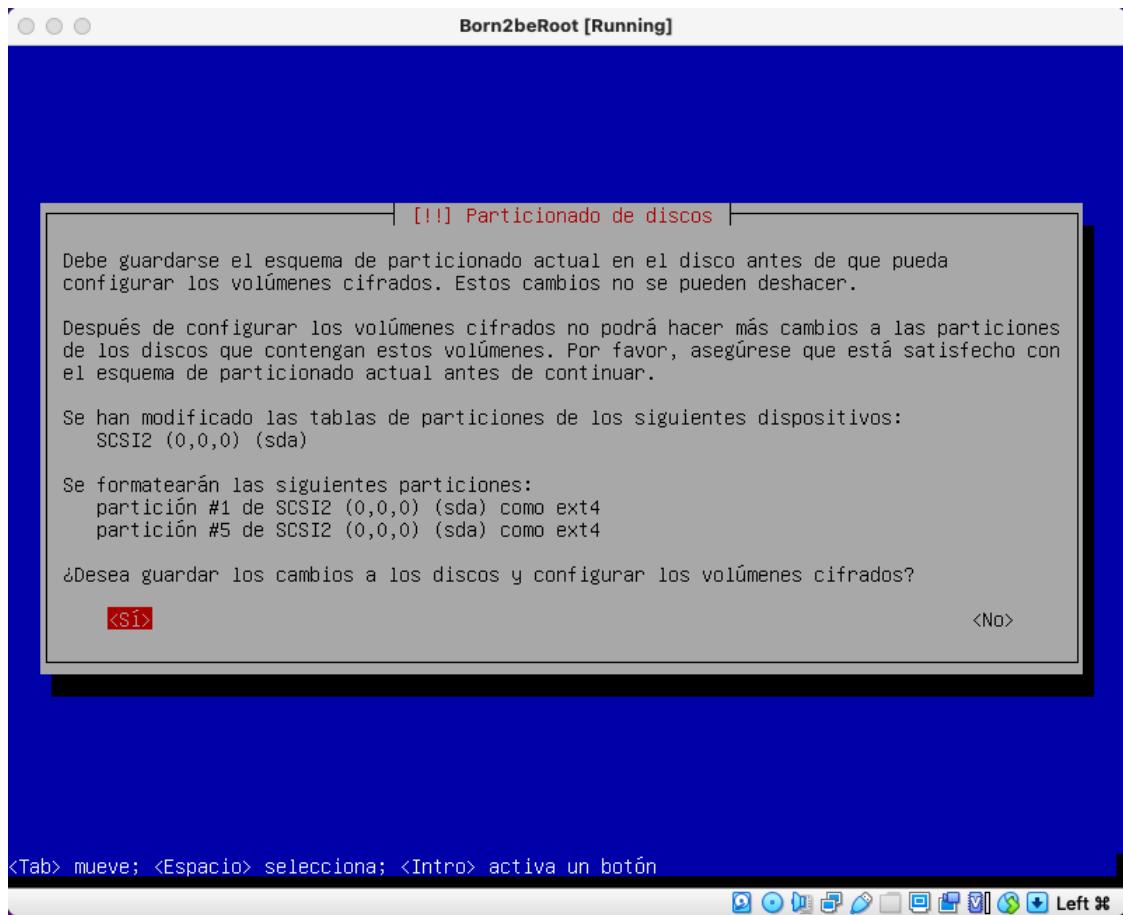
31. De regreso, escogemos '**Se ha terminado de definir la partición**' y pulsamos **<Enter>**



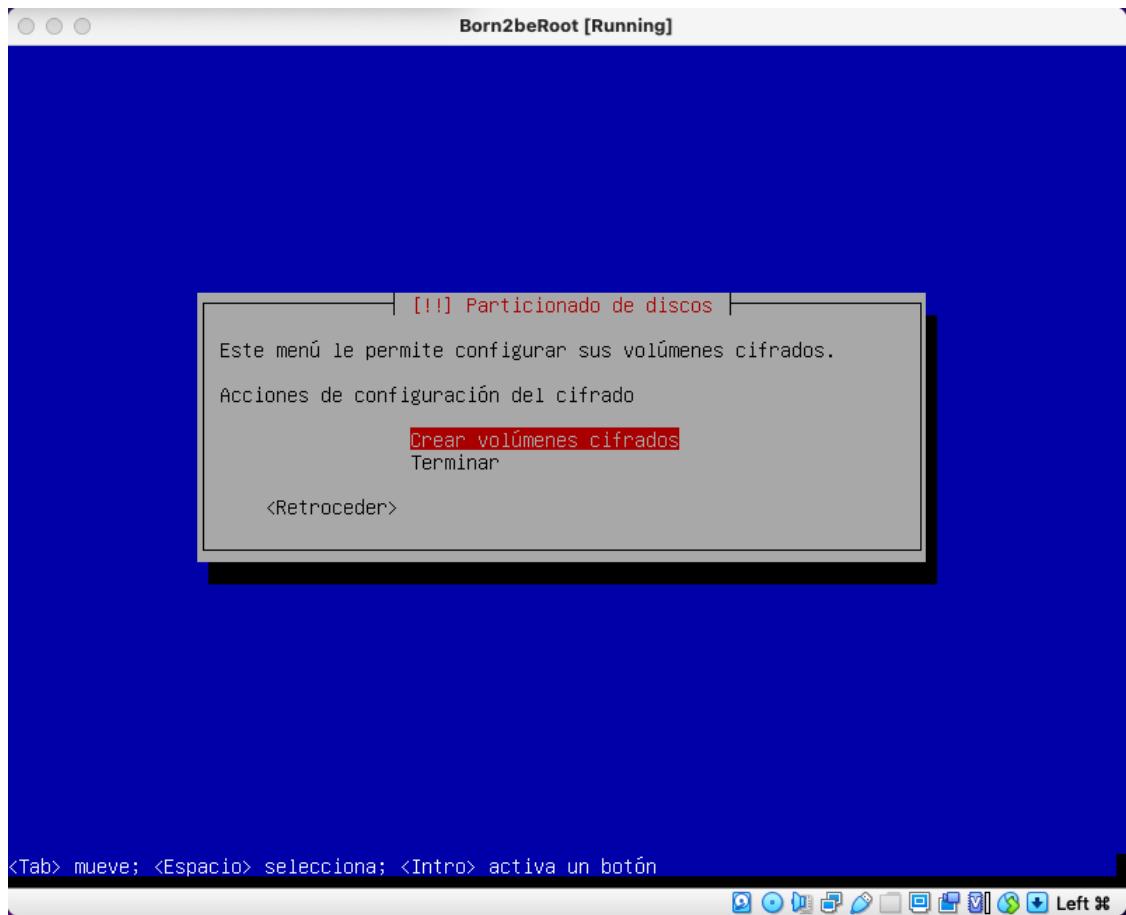
32. De vuelta al '**Particionado de discos**', seleccionamos la opción '**Configurar los volúmenes cifrados**' y pulsamos **<Enter>**



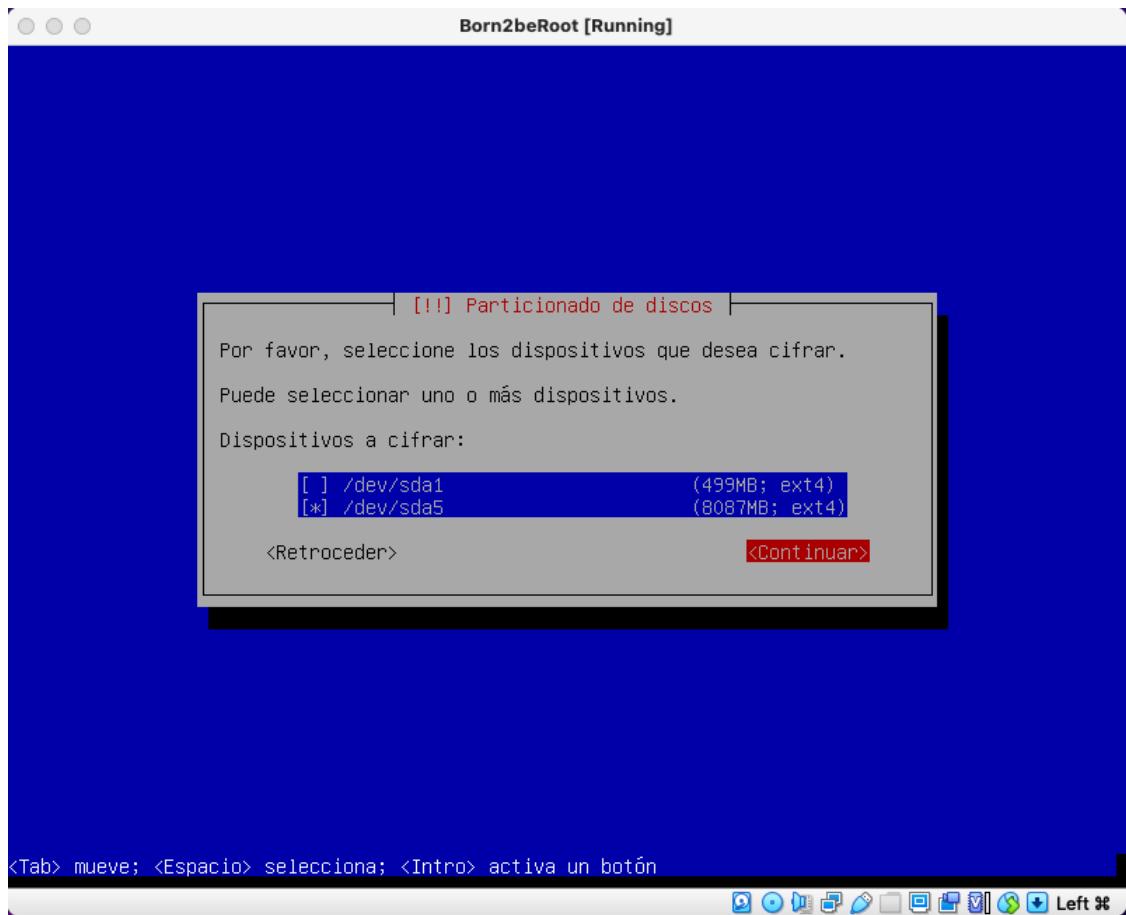
33. En la siguiente pantalla siguiente escogemos **<Sí>** para guardar los cambios realizados en los discos y configurar los volúmenes cifrados



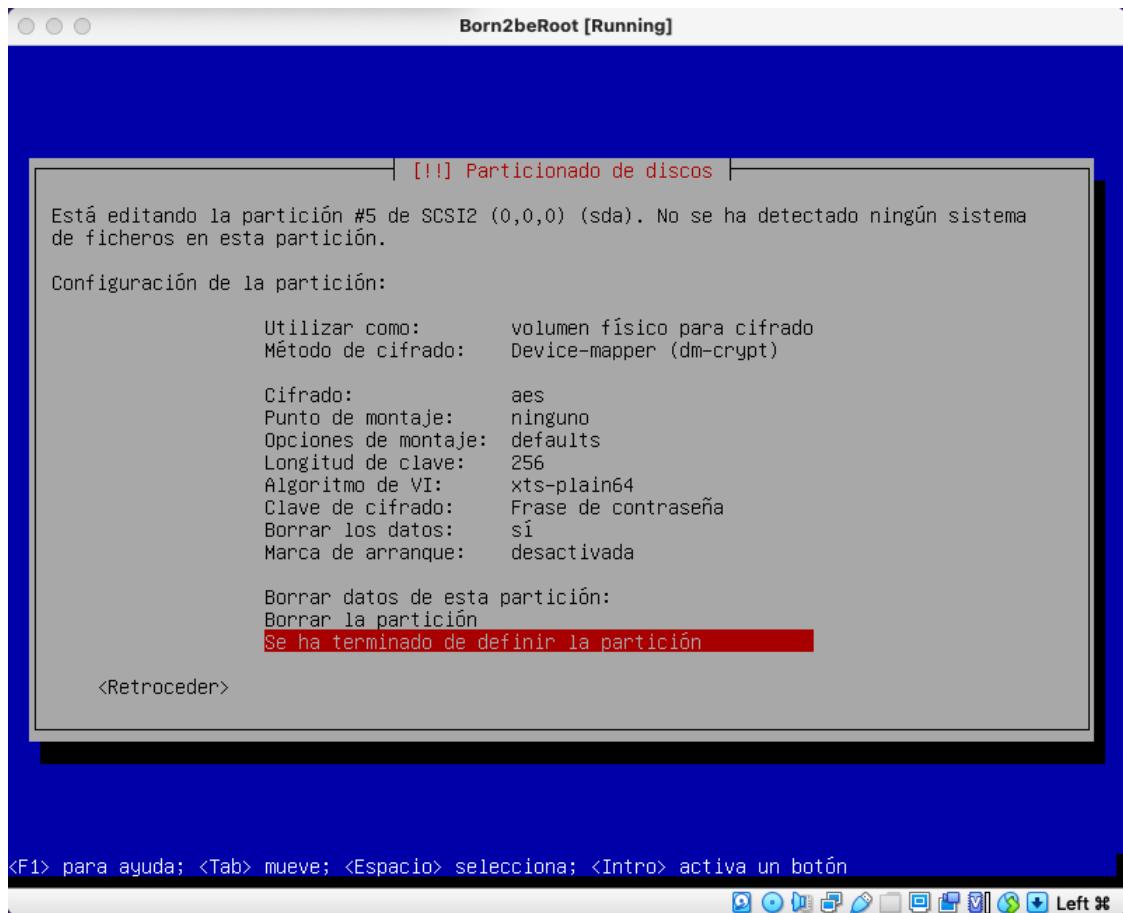
34. Seleccionamos '**Crear volúmenes cifrados**' y pulsamos **<Enter>**



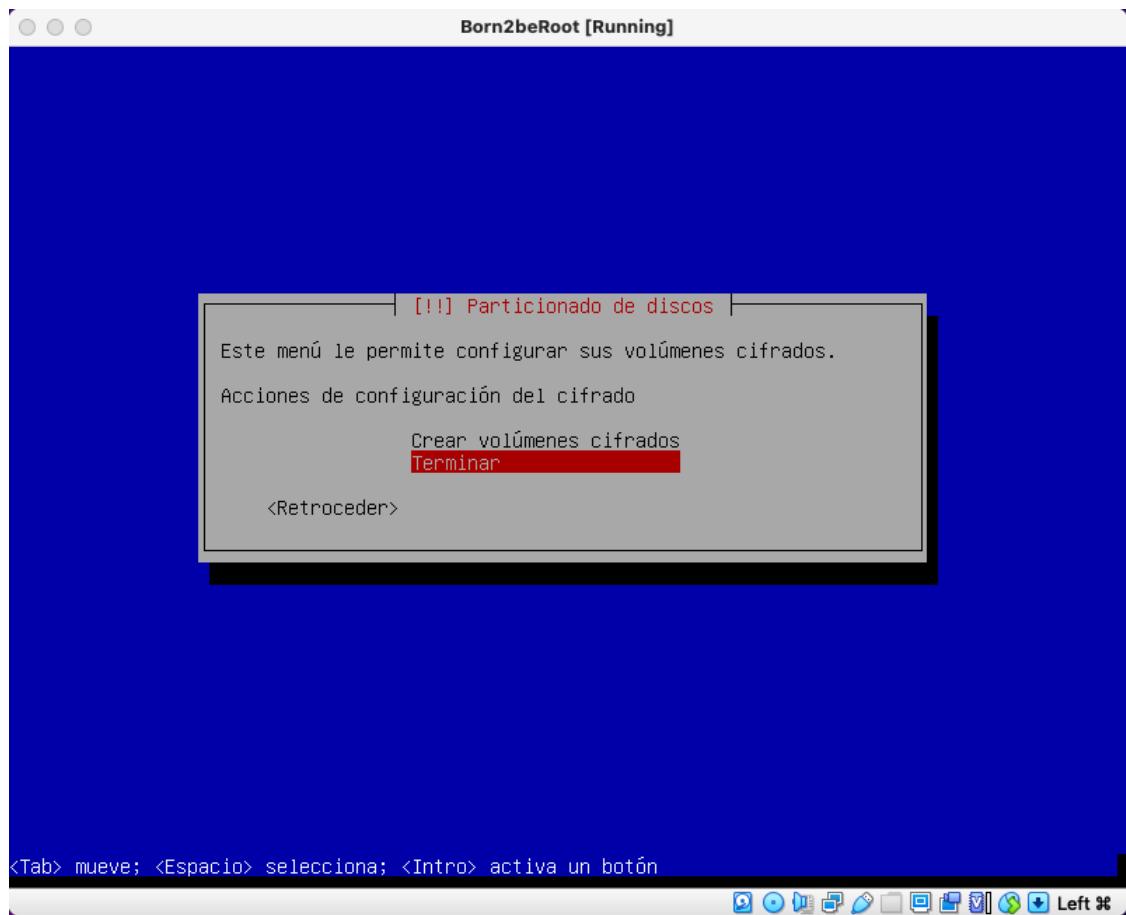
35. Seleccionamos con <Espacio> la entrada '**/dev/sda5**' (de 8087MB) y pulsamos sobre <Continuar>



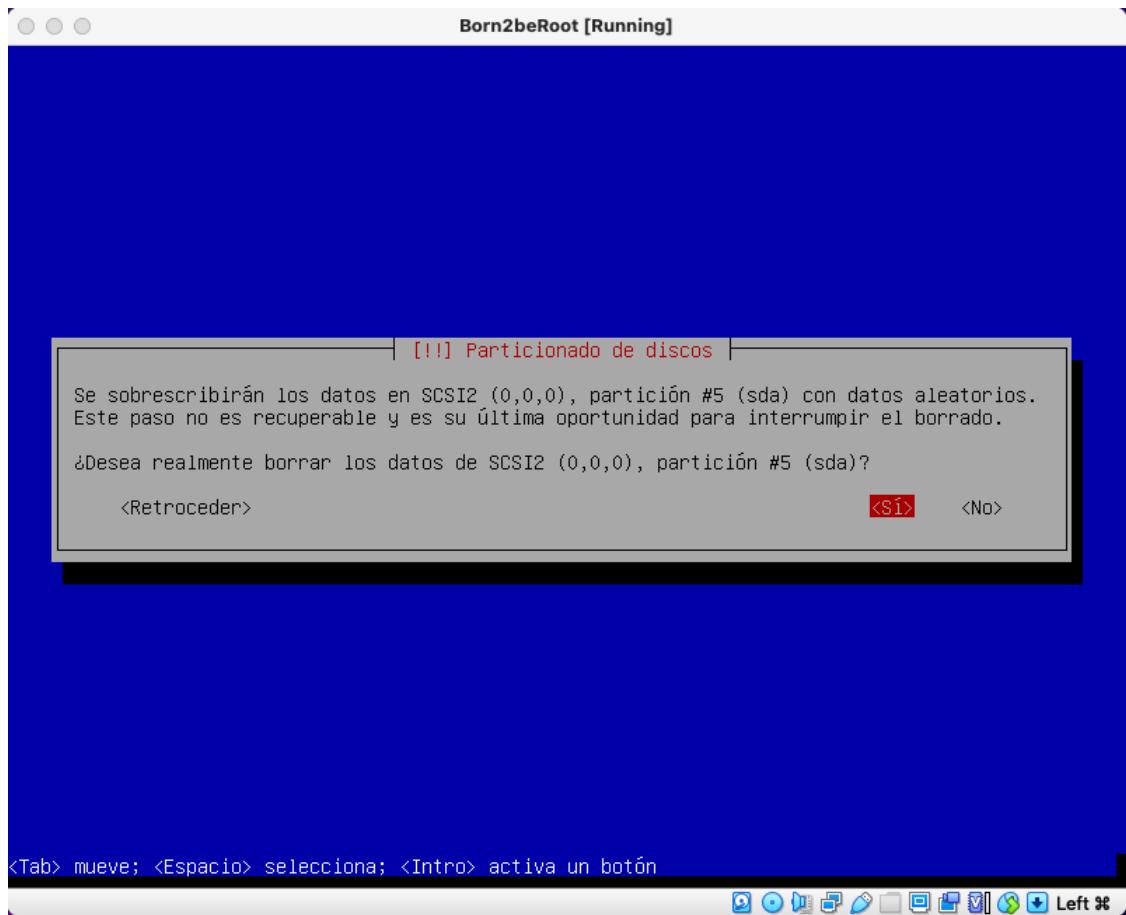
36. Marcamos '**Se ha terminado de definir la partición**' y pulsamos **<Enter>**



37. Seleccionamos '**Terminar**', luego **<Enter>**



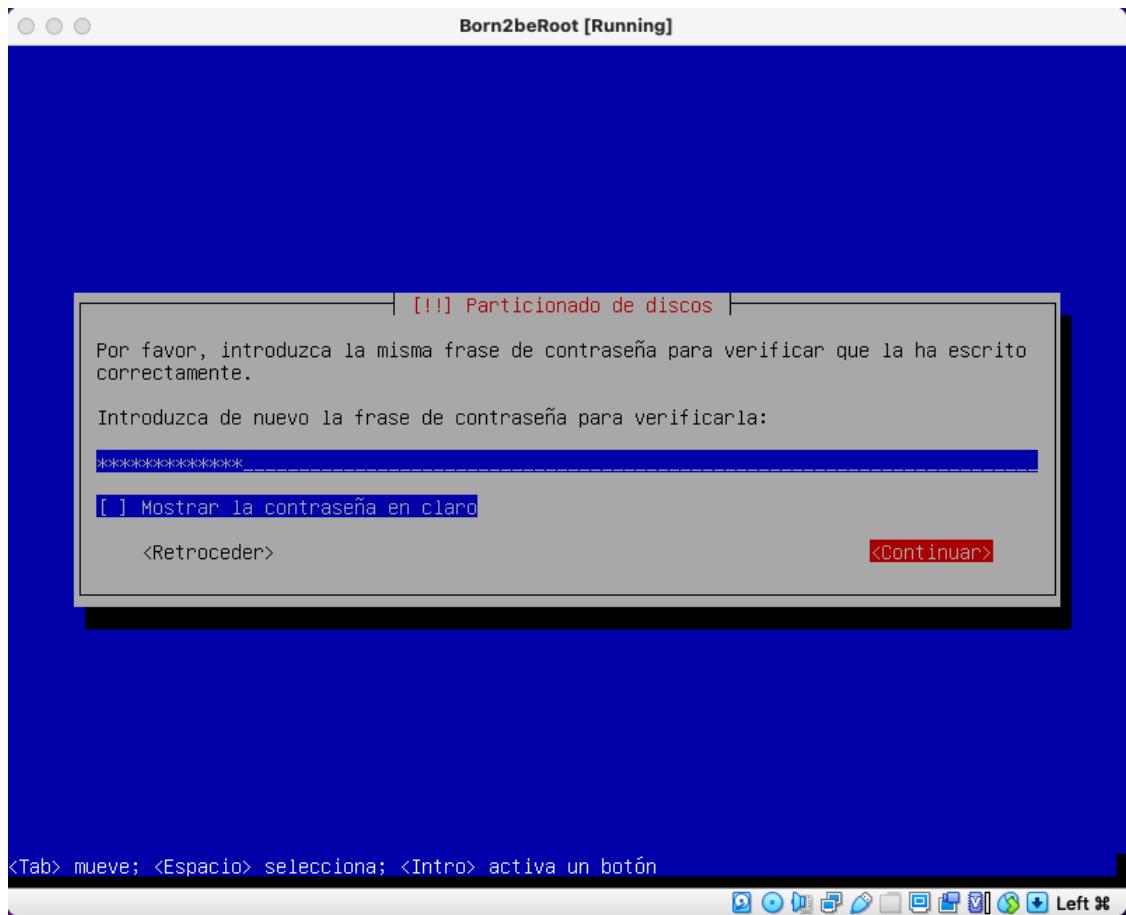
38. Cuando nos pregunte si deseamos borrar los datos de SCSI2, seleccionamos <Sí> y esperamos a que finalice el proceso



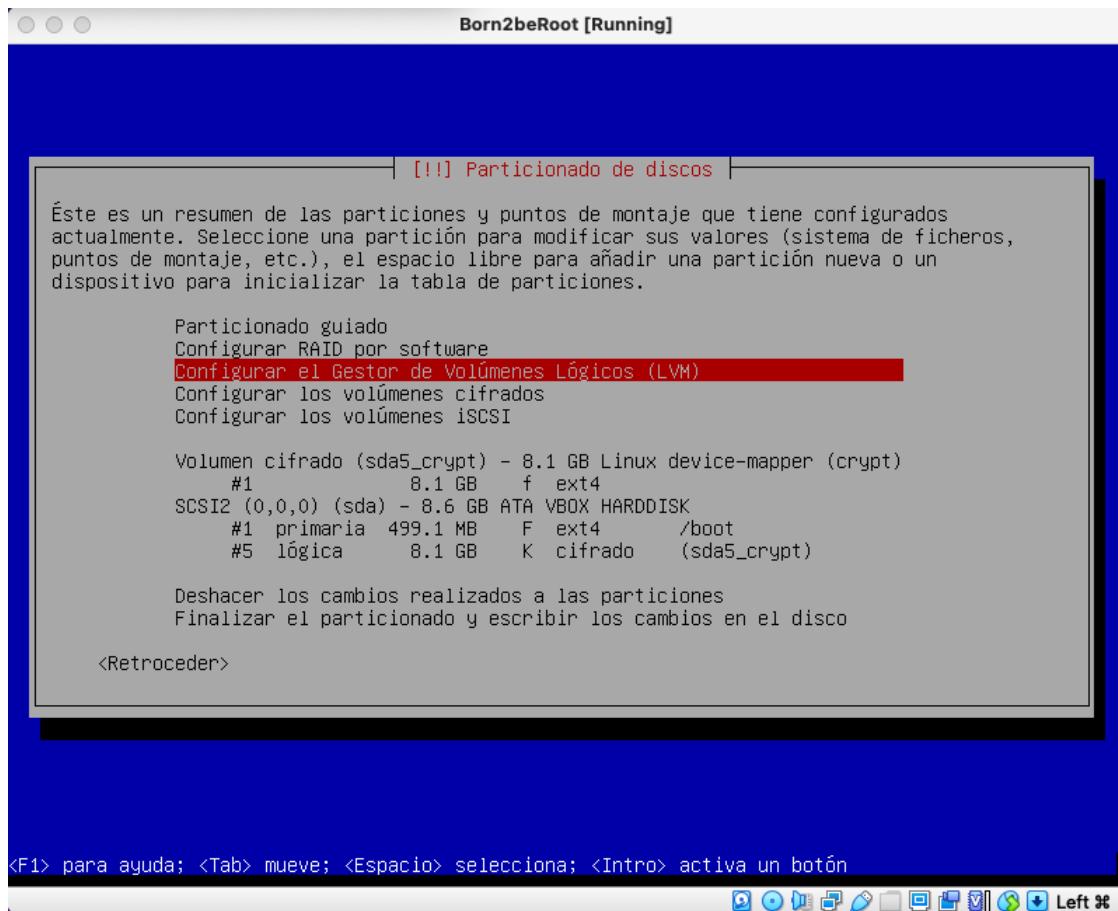
39. Escogemos la Frase de contraseña de cifrado (lsdscrsdlvrdd) y pulsamos sobre <[Continuar](#)>



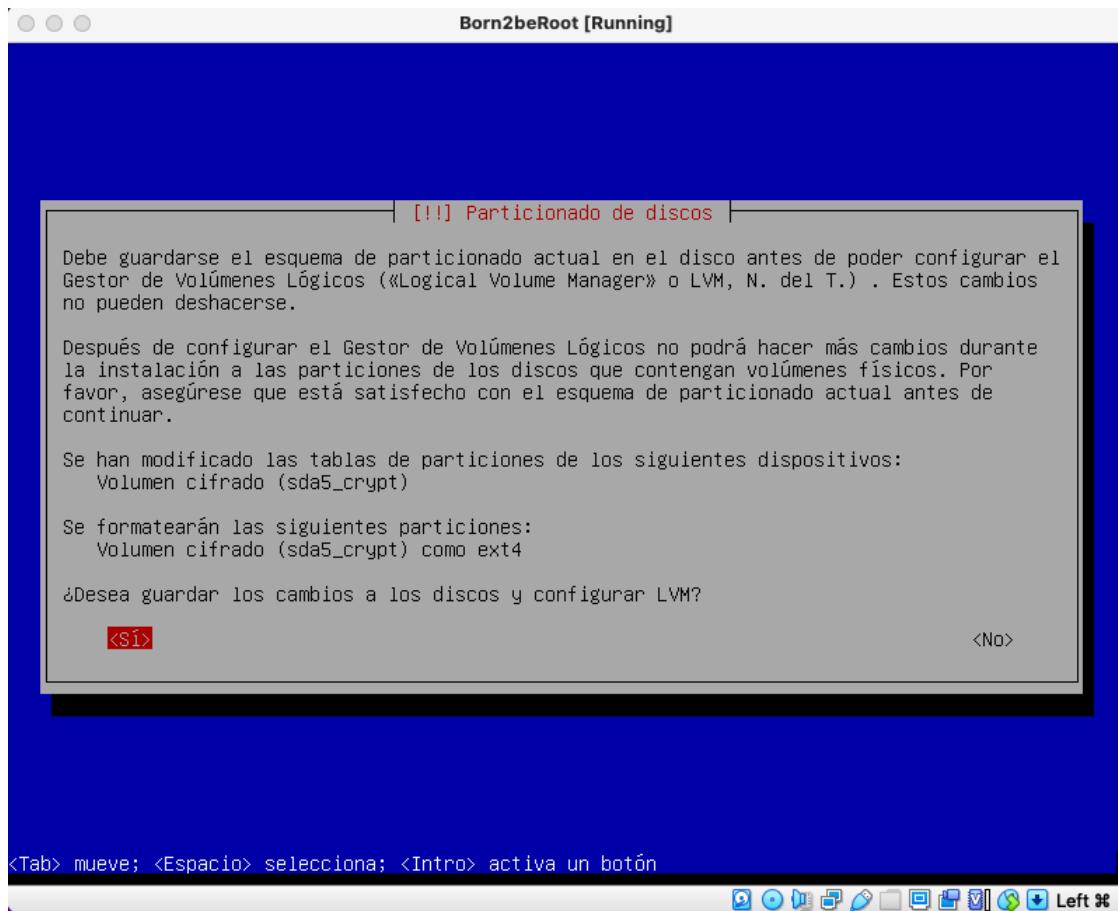
40. Volvemos a introducir la frase de cifrado y de nuevo <Continuar>



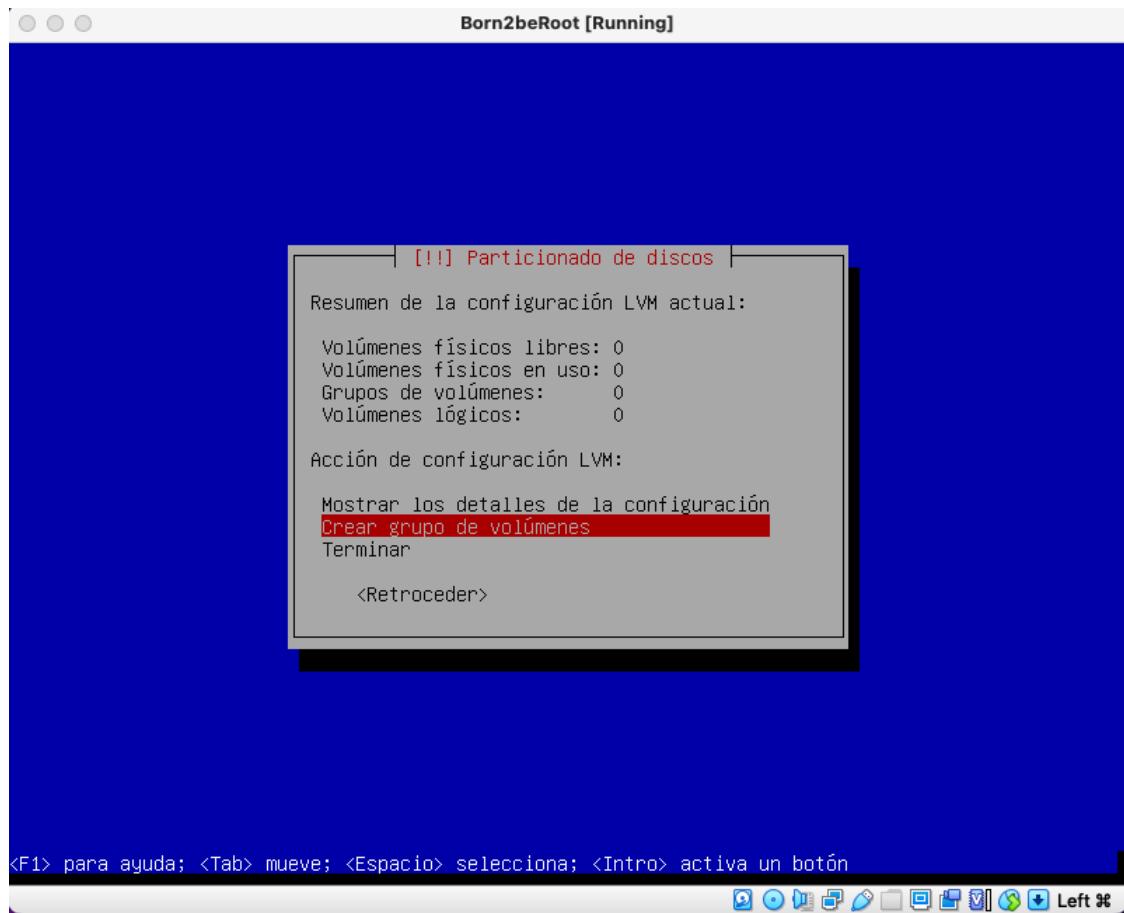
41. Otra vez en la pantalla de '**Particionado de discos**' seleccionamos '**Configurar el Gestor de Volúmenes Lógicos (LVM)**'



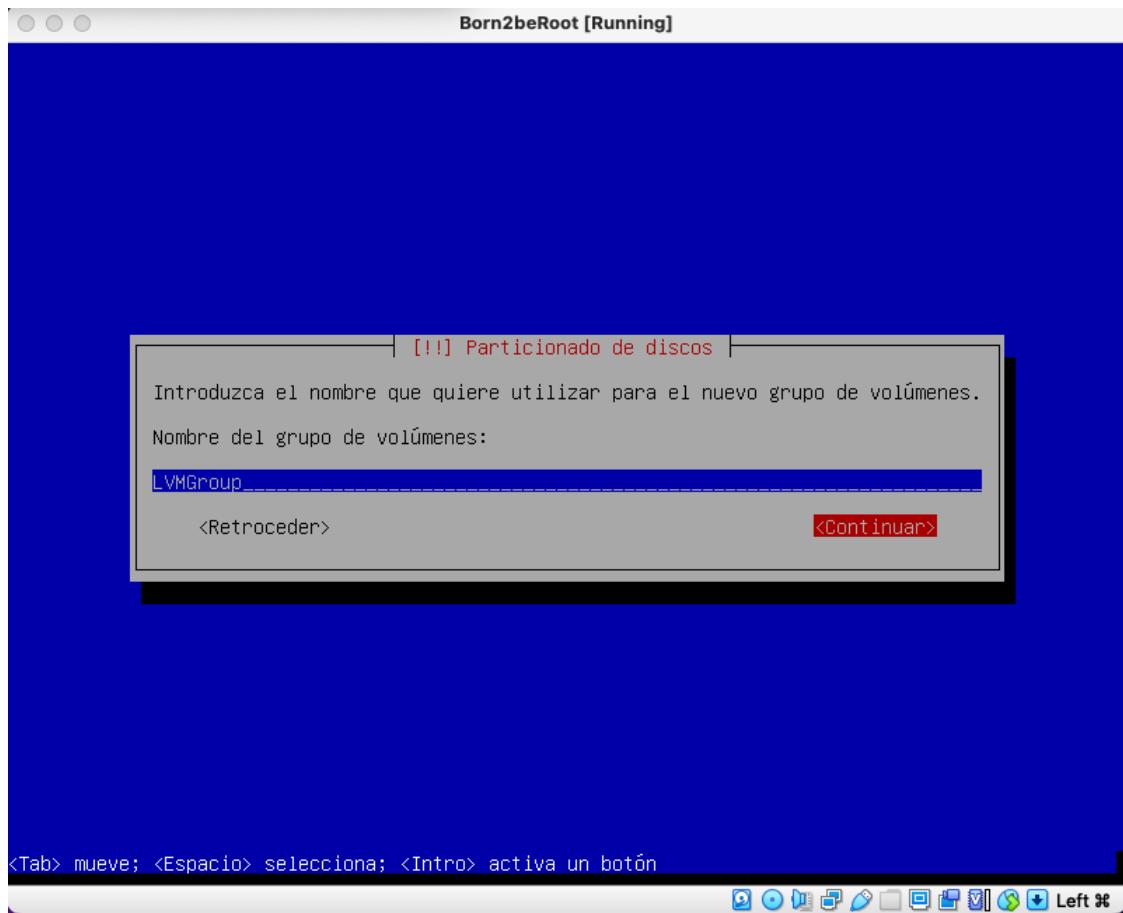
42. Escogemos **<Sí>** para guardar los cambios a los discos y configurar LVM



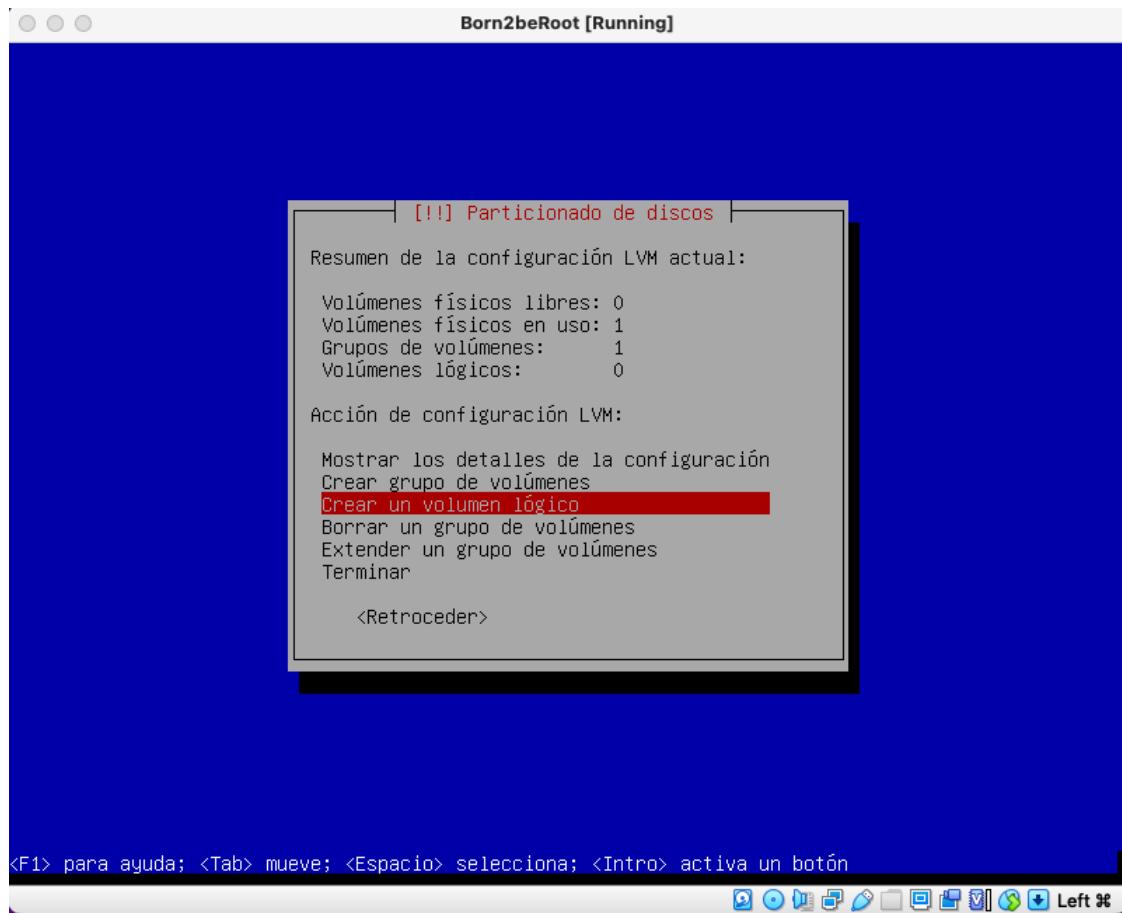
43. Seleccionamos '**Crear grupo de volúmenes**' y pulsamos **<Enter>**



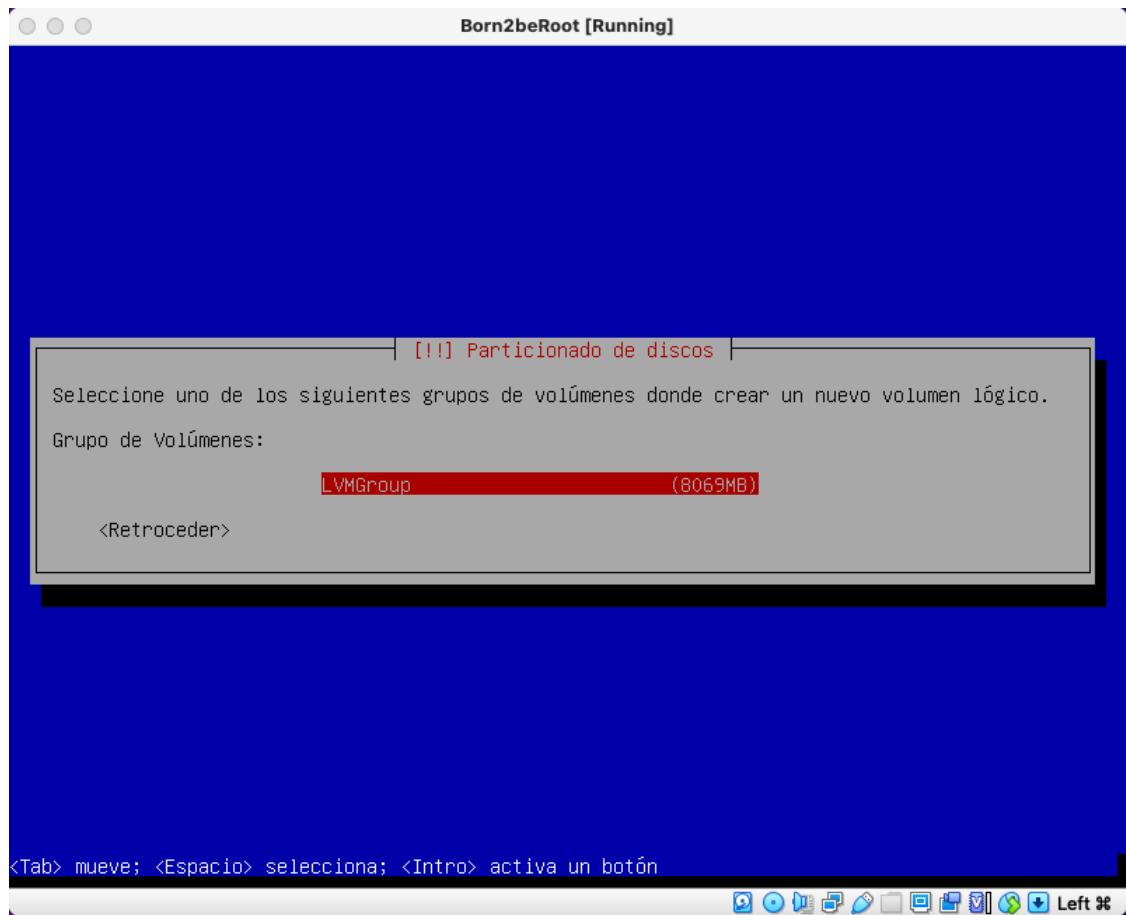
44. Asignamos el nombre **LVMGroup** al grupo de volúmenes y seleccionamos **<Continuar>**



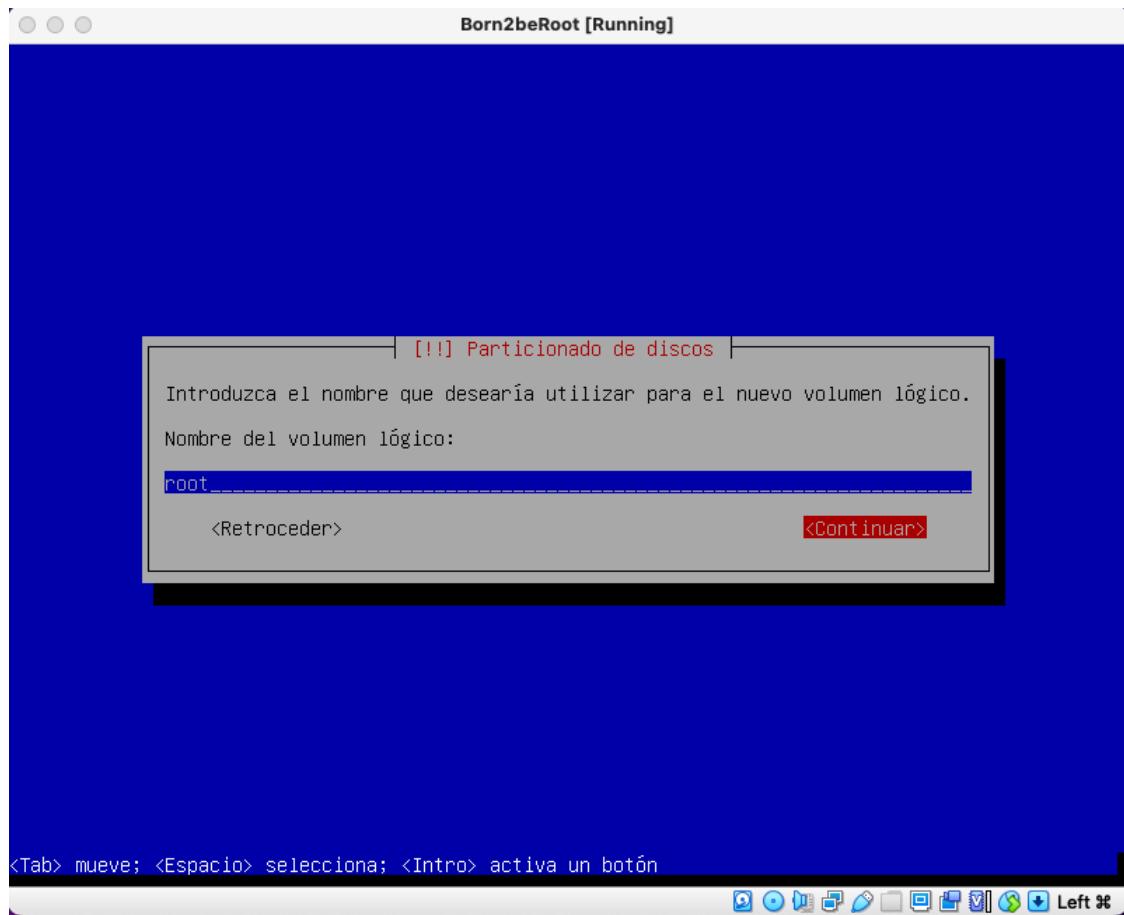
45. Seleccionamos el dispositivo '**/dev/mapper/sda5_crypt**' (de 8070MB) y pulsamos **<Continuar>**
46. Vamos a '**Crear un volumen lógico**' con la opción correspondiente y pulsamos **<Enter>**



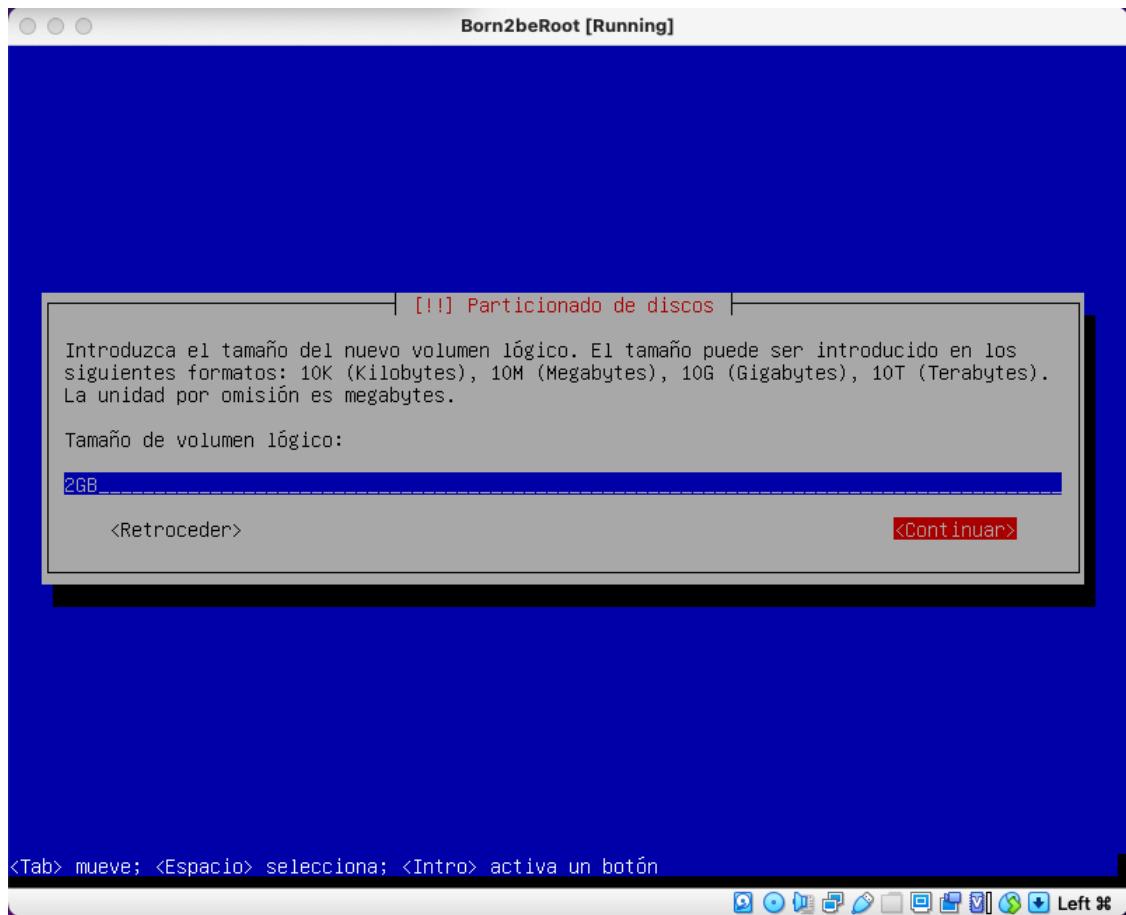
47. Pulsamos **<Enter>** en la pantalla donde se muestra el Grupo de Volúmenes



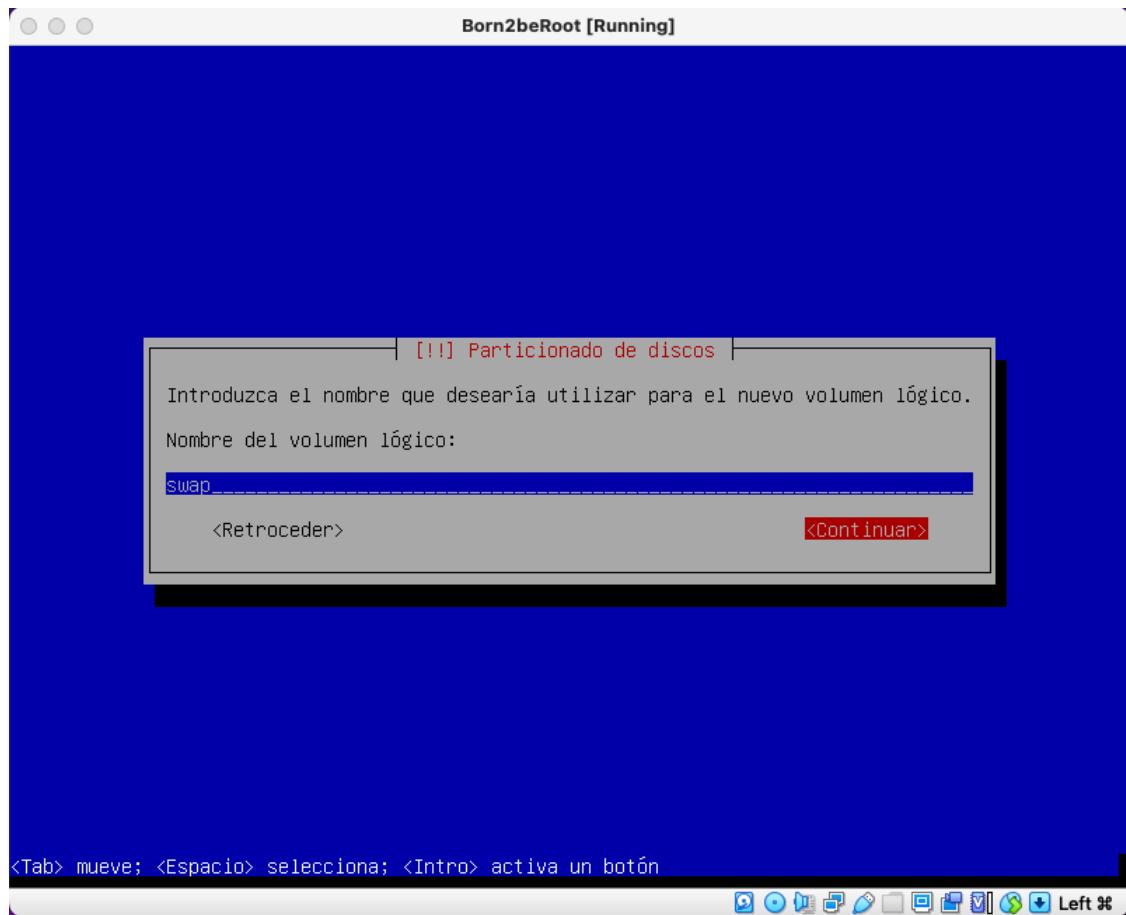
48. Le damos el nombre '**root**'

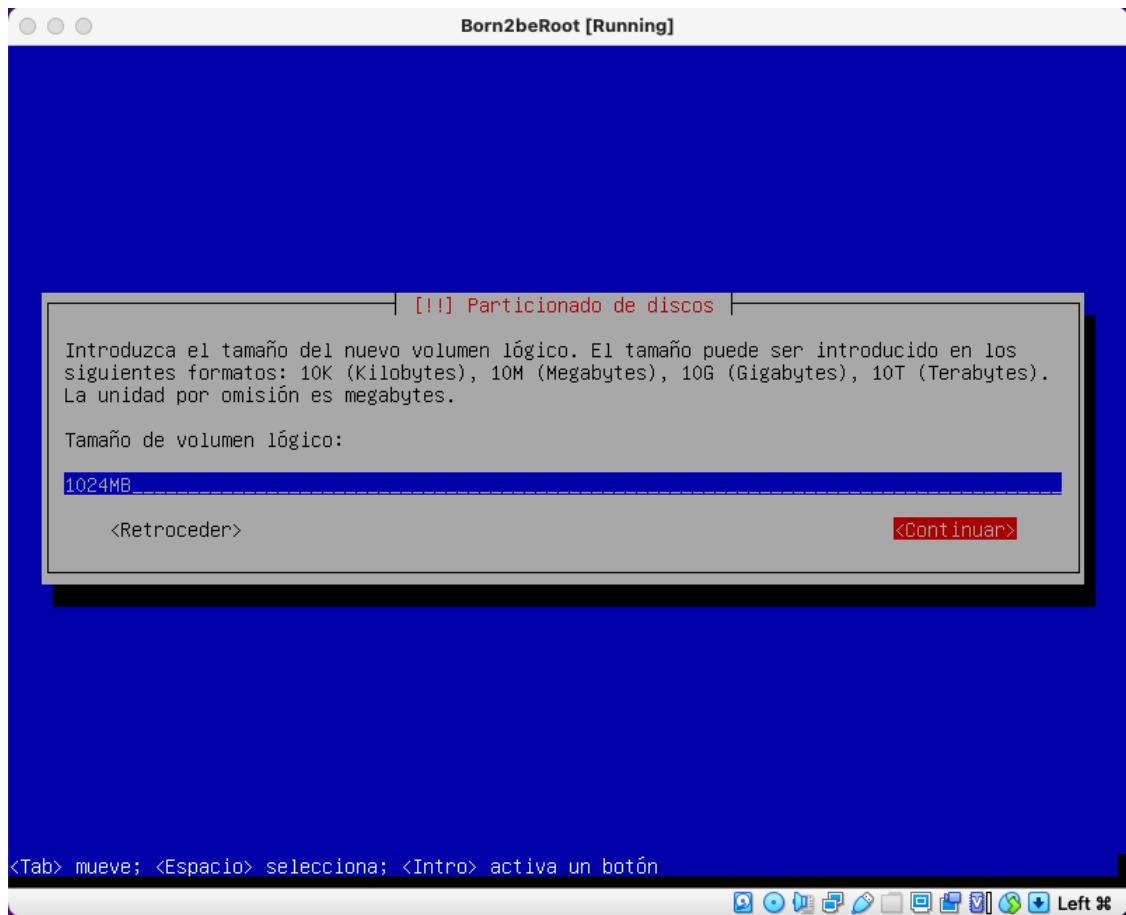


49. Asignamos 2GB y pulsamos [**<Continuar>**](#)

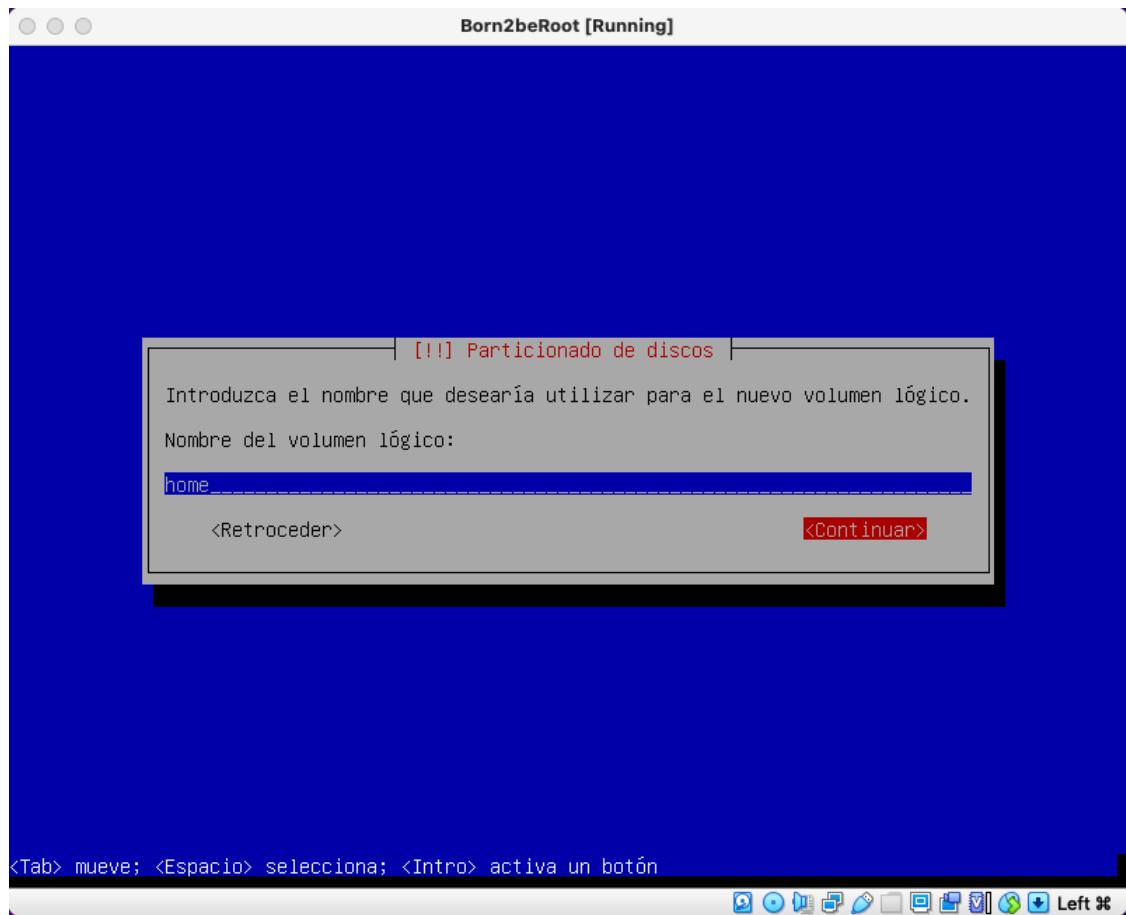


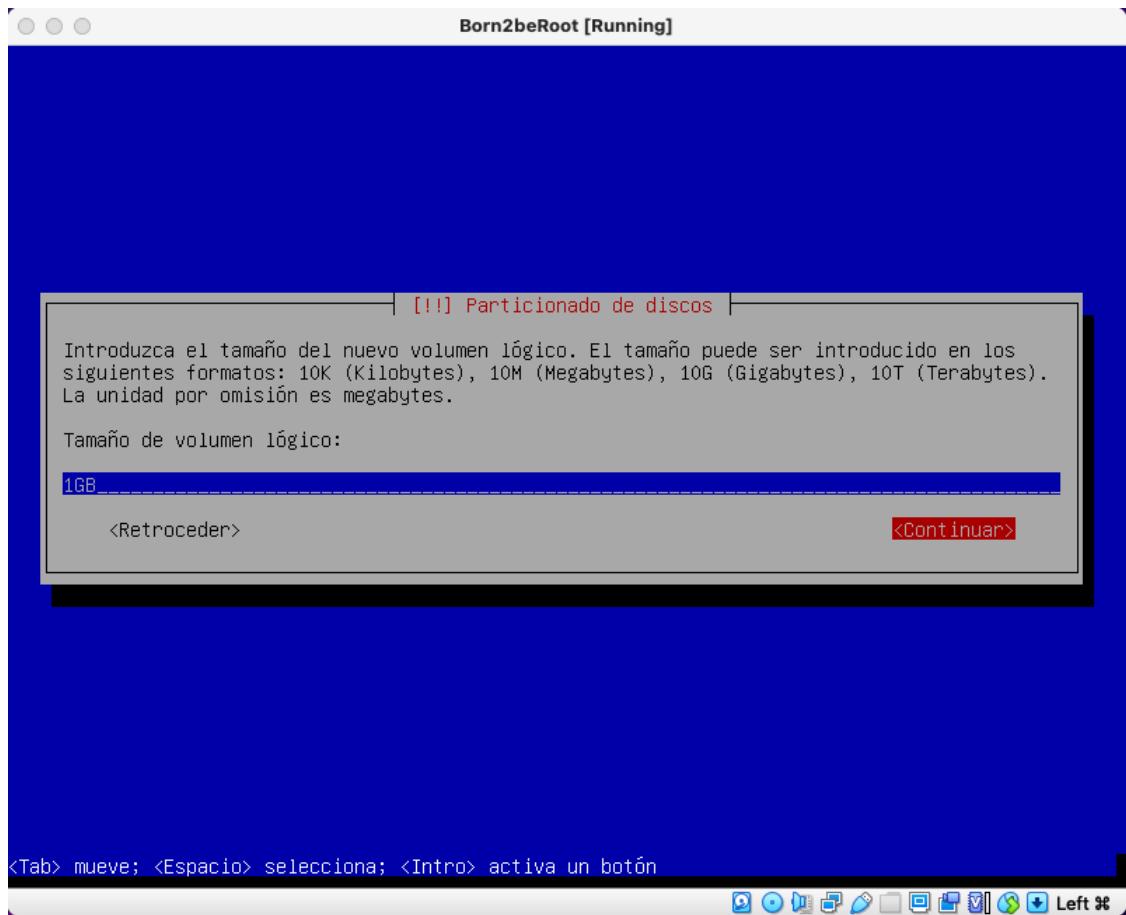
50. Repetimos la creación de otro volumen lógico, ahora de nombre '**swap**' y '**1024MB**'



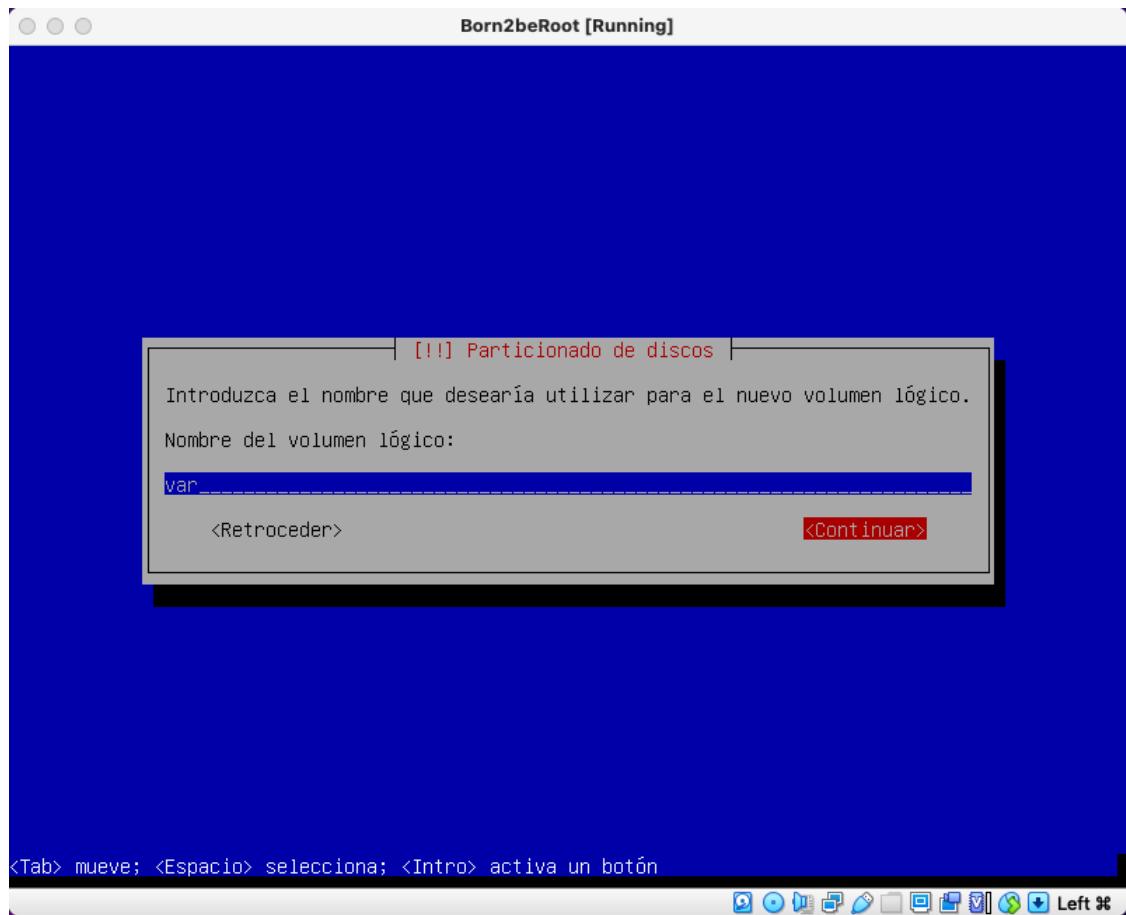


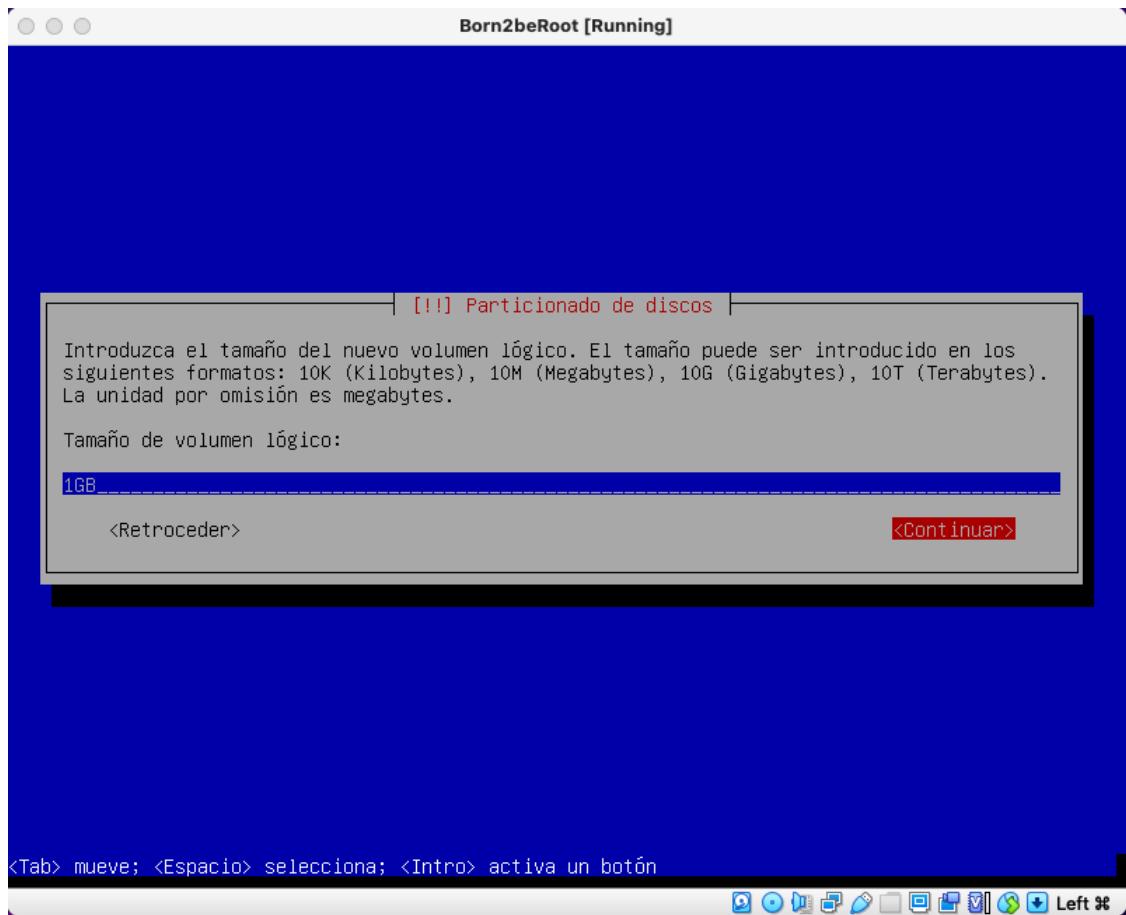
51. Otro más de nombre '**home**' y '**1GB**' de tamaño



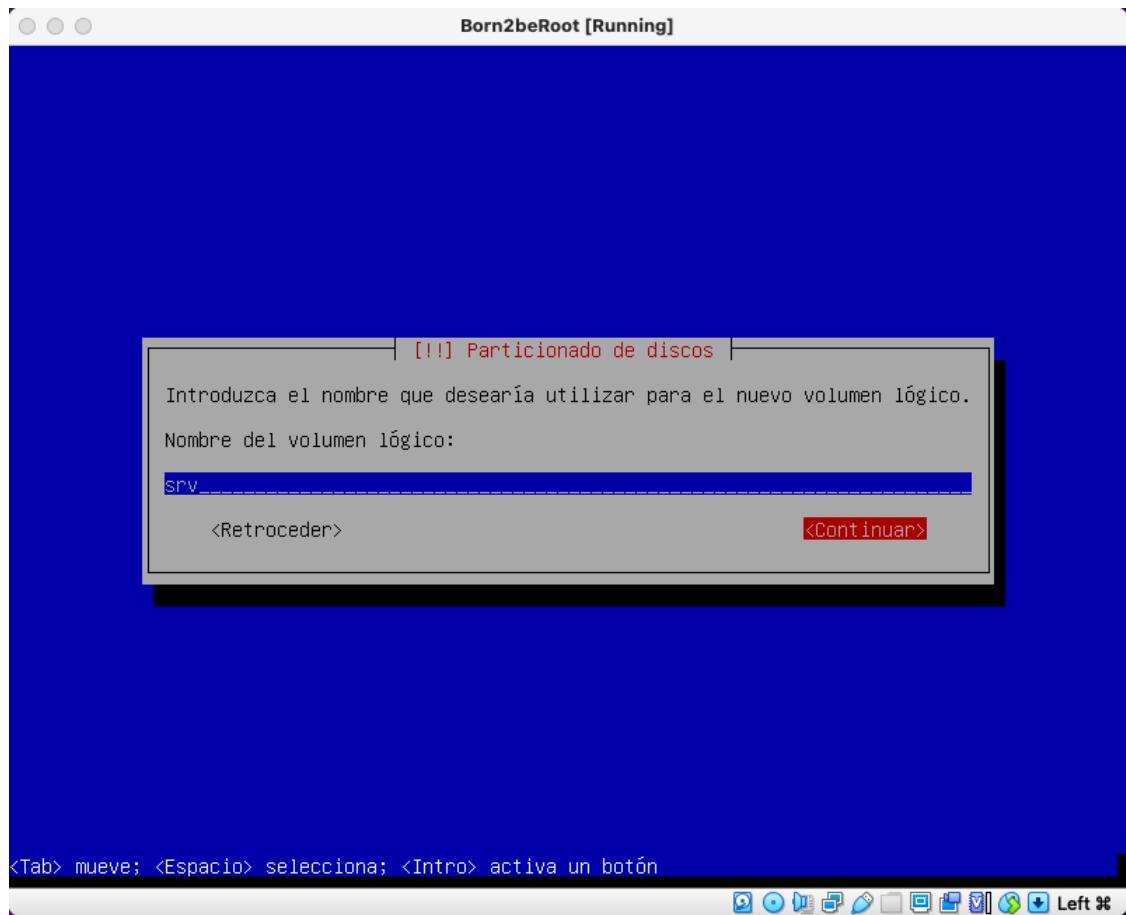


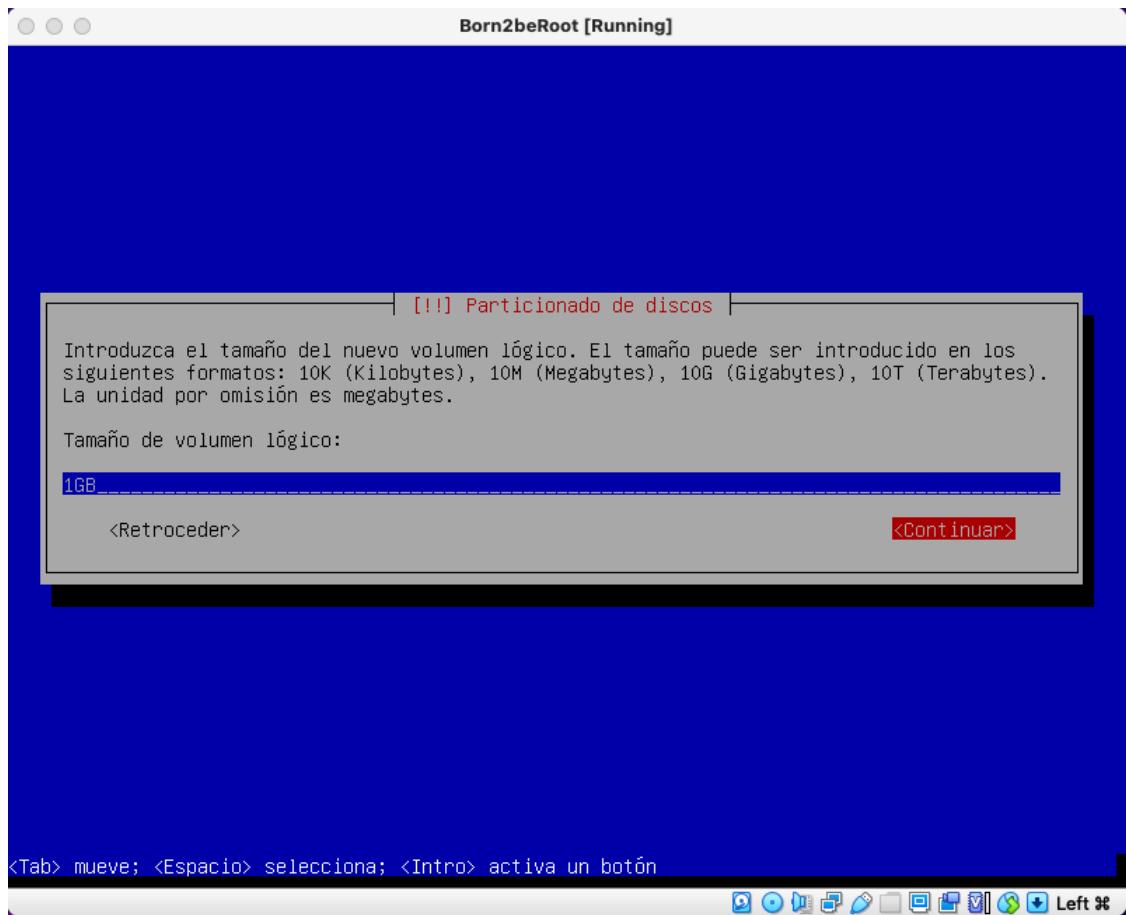
52. Otro de nombre '**var**' y tamaño '**1GB**'



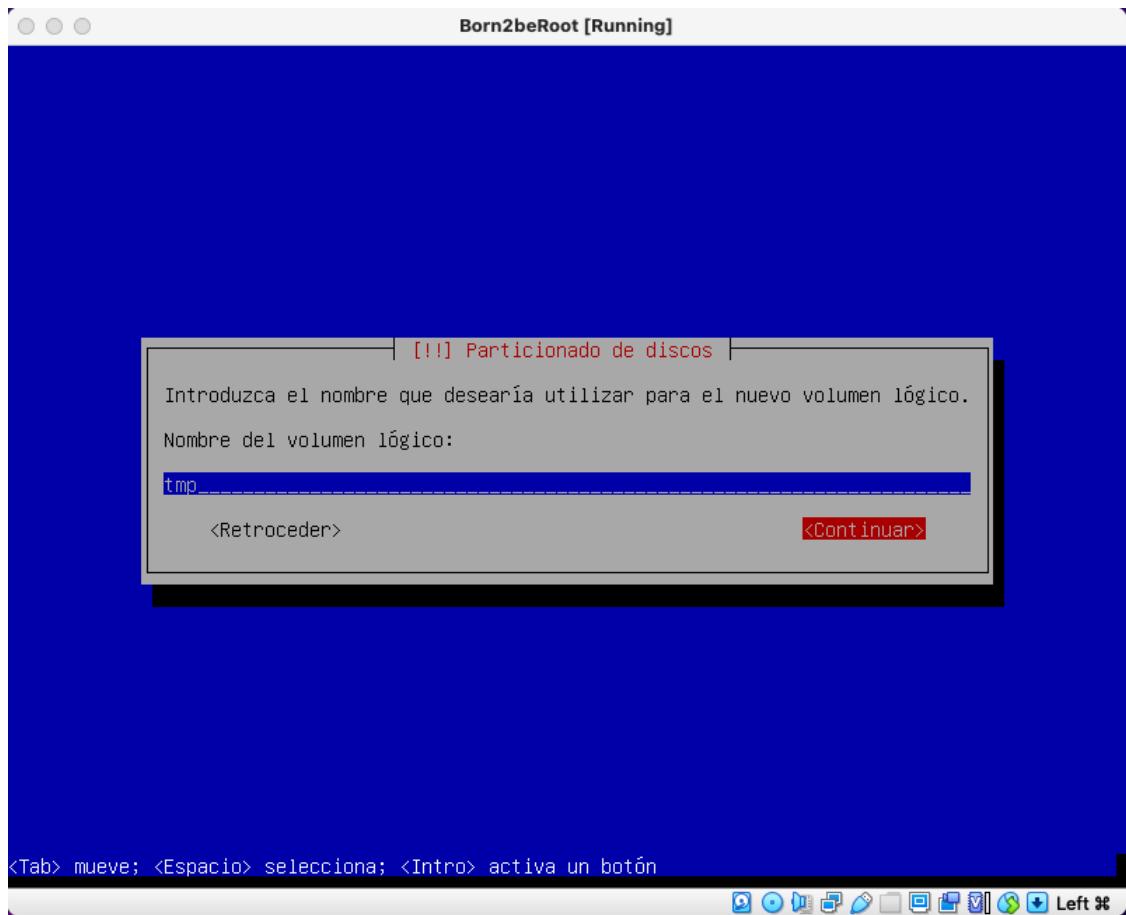


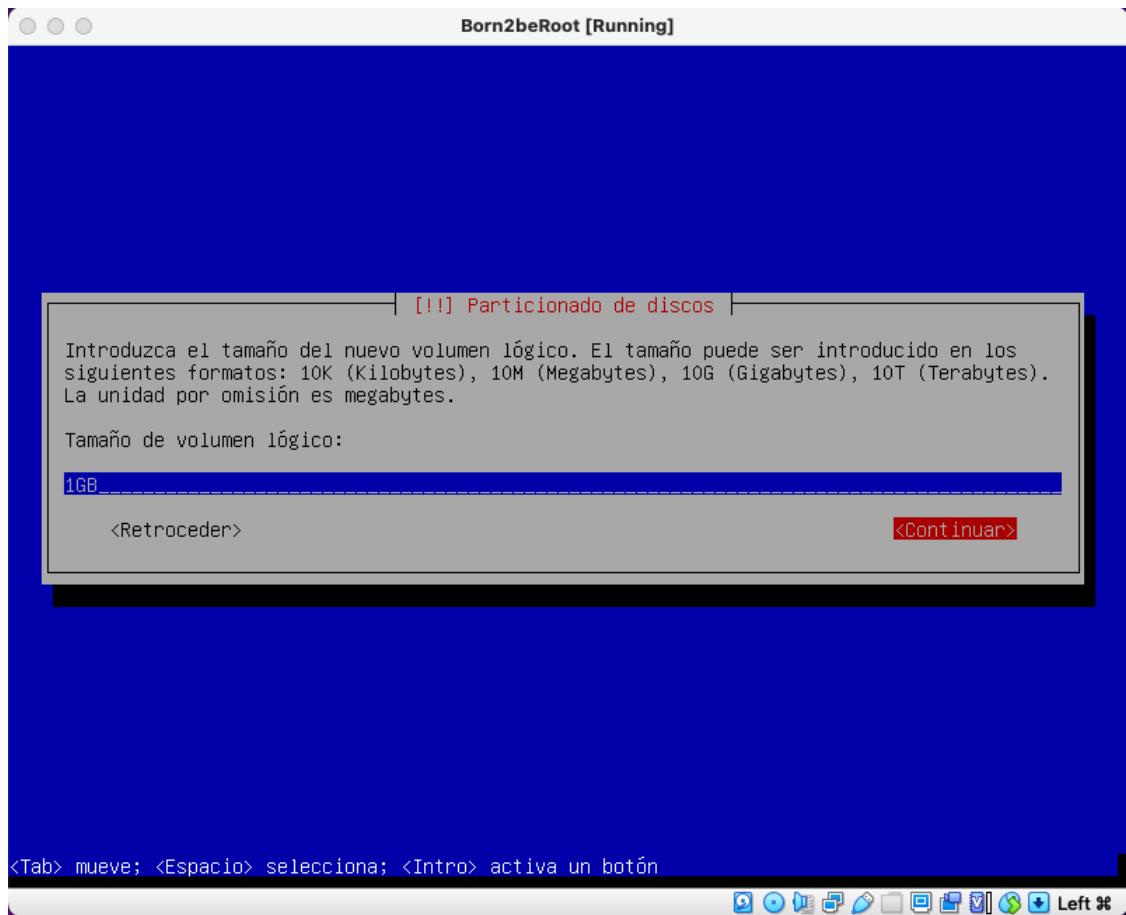
53. Otro más de nombre '**srv**' y tamaño '**1GB**'



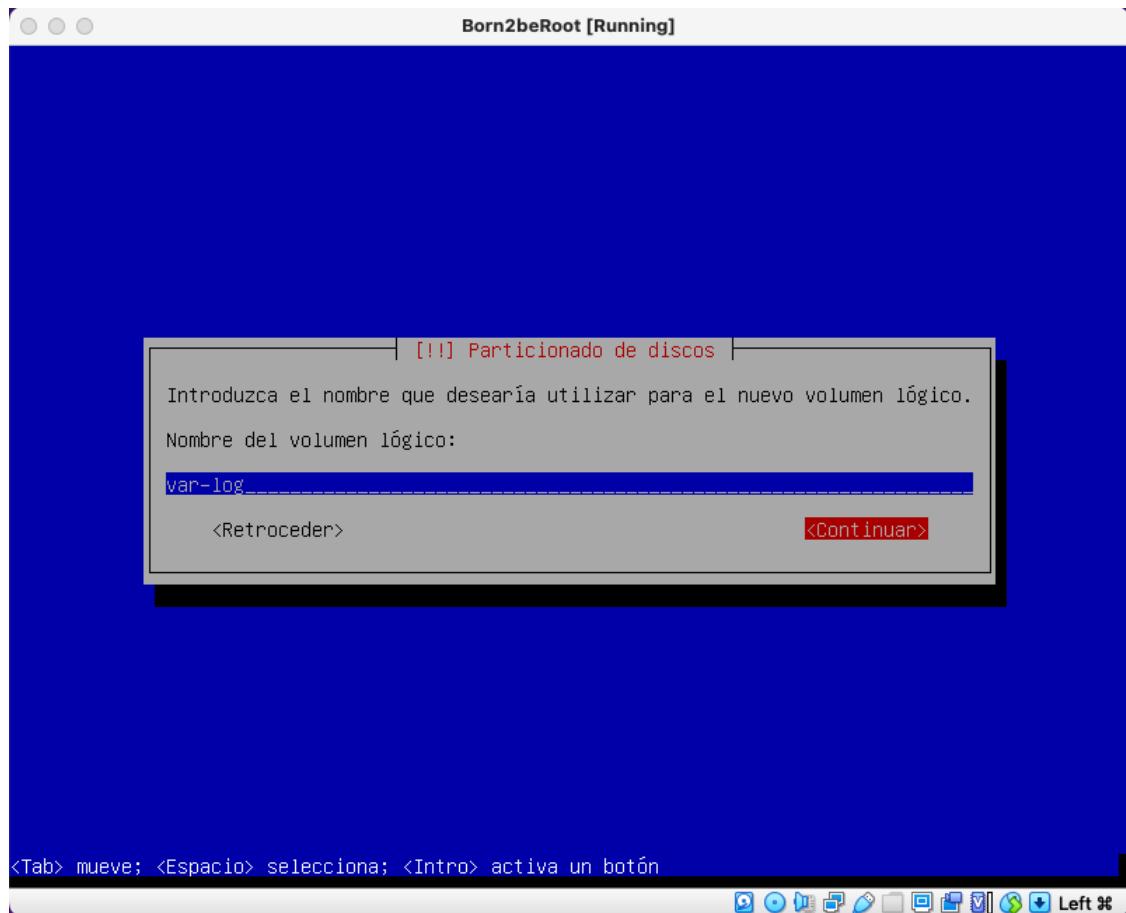


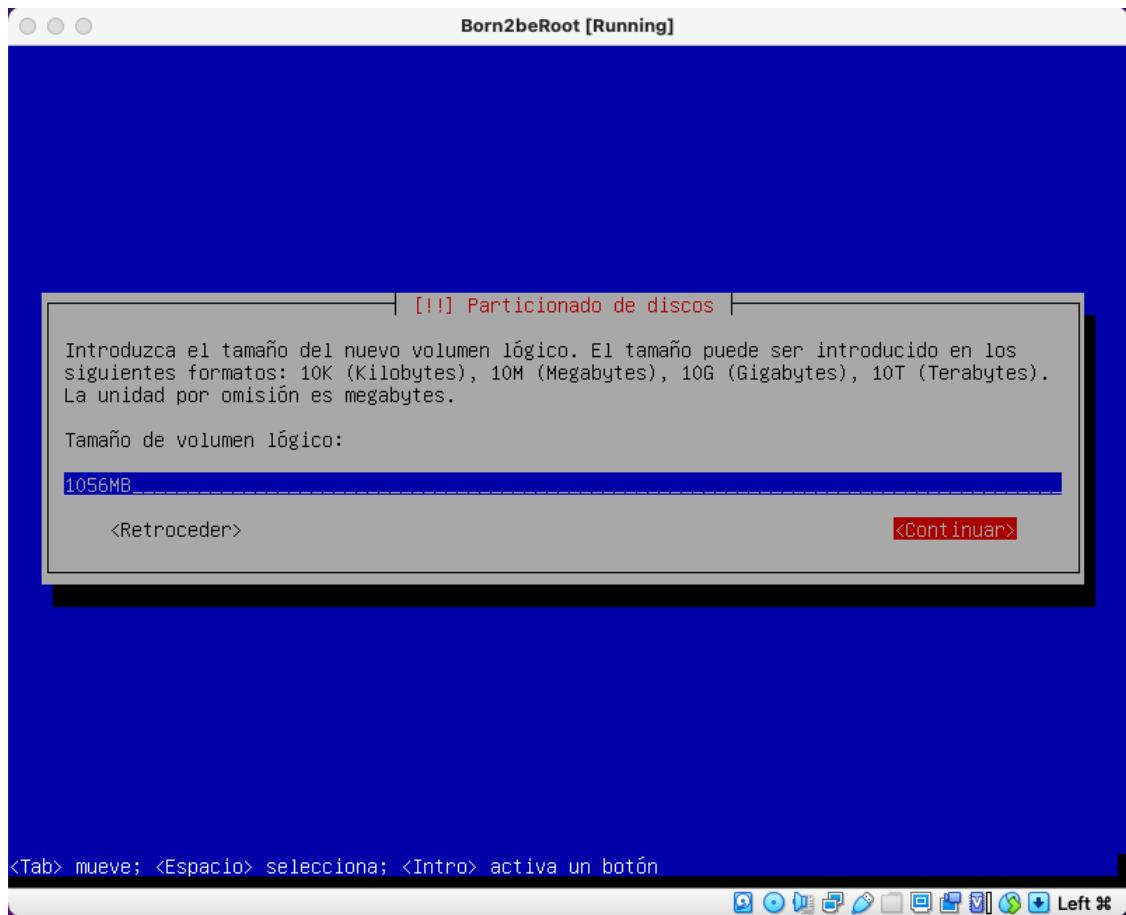
54. Otro de nombre '**tmp**' y tamaño '**1GB**



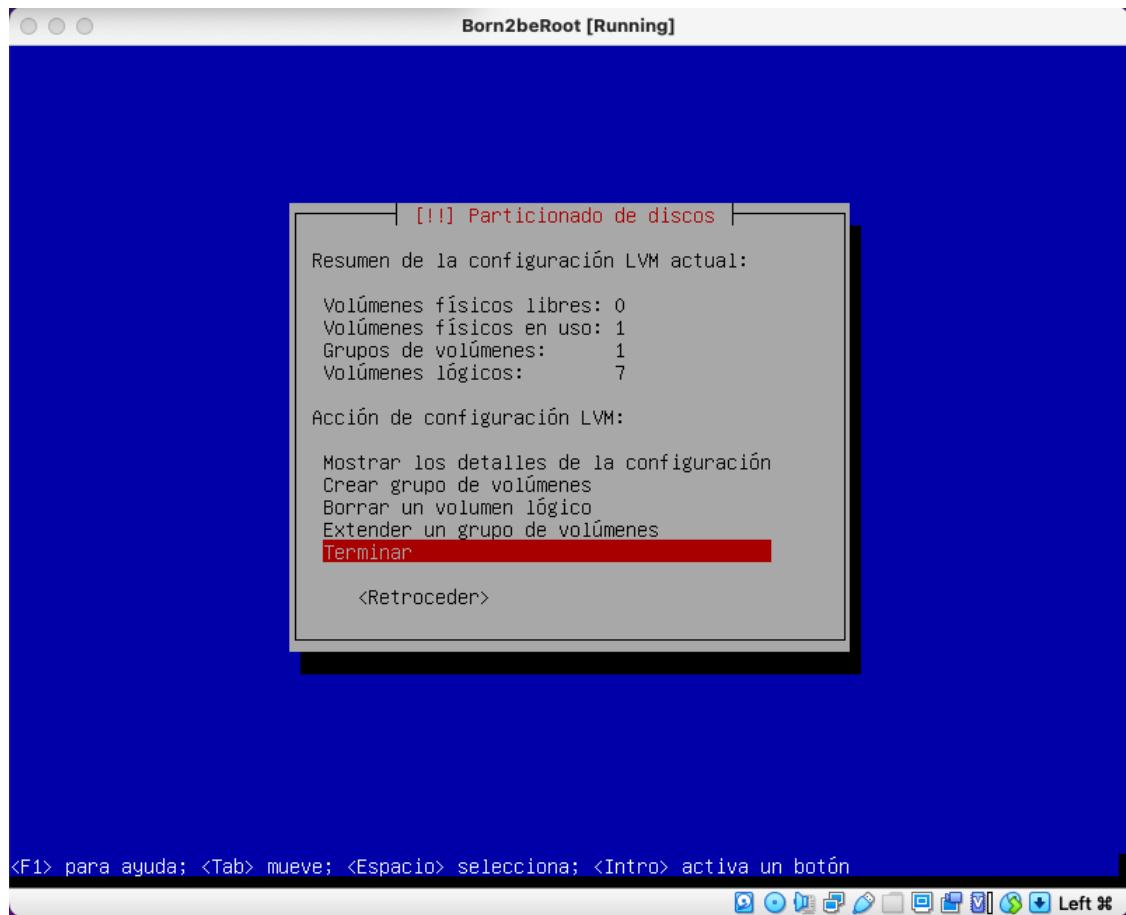


55. Y por último uno de nombre '**var-log**' y tamaño restante del grupo (**1056MB**)

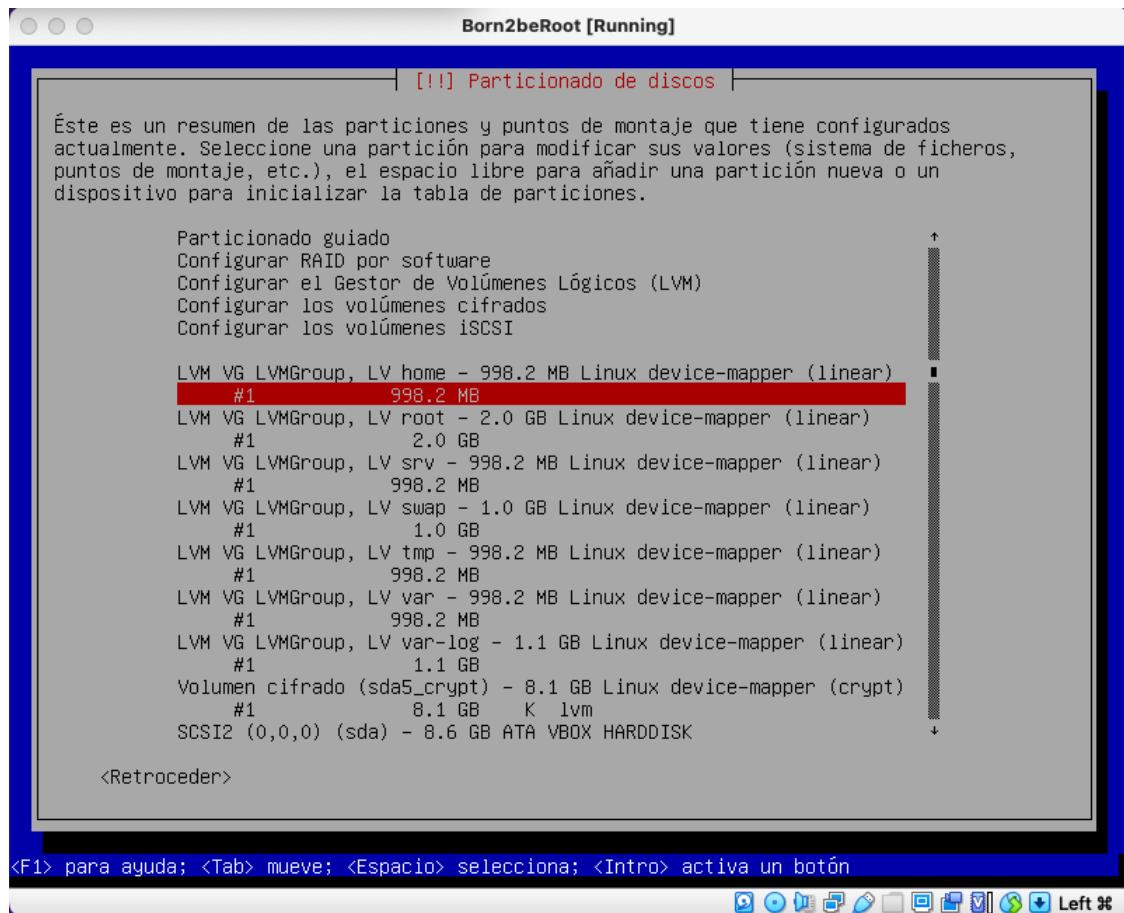




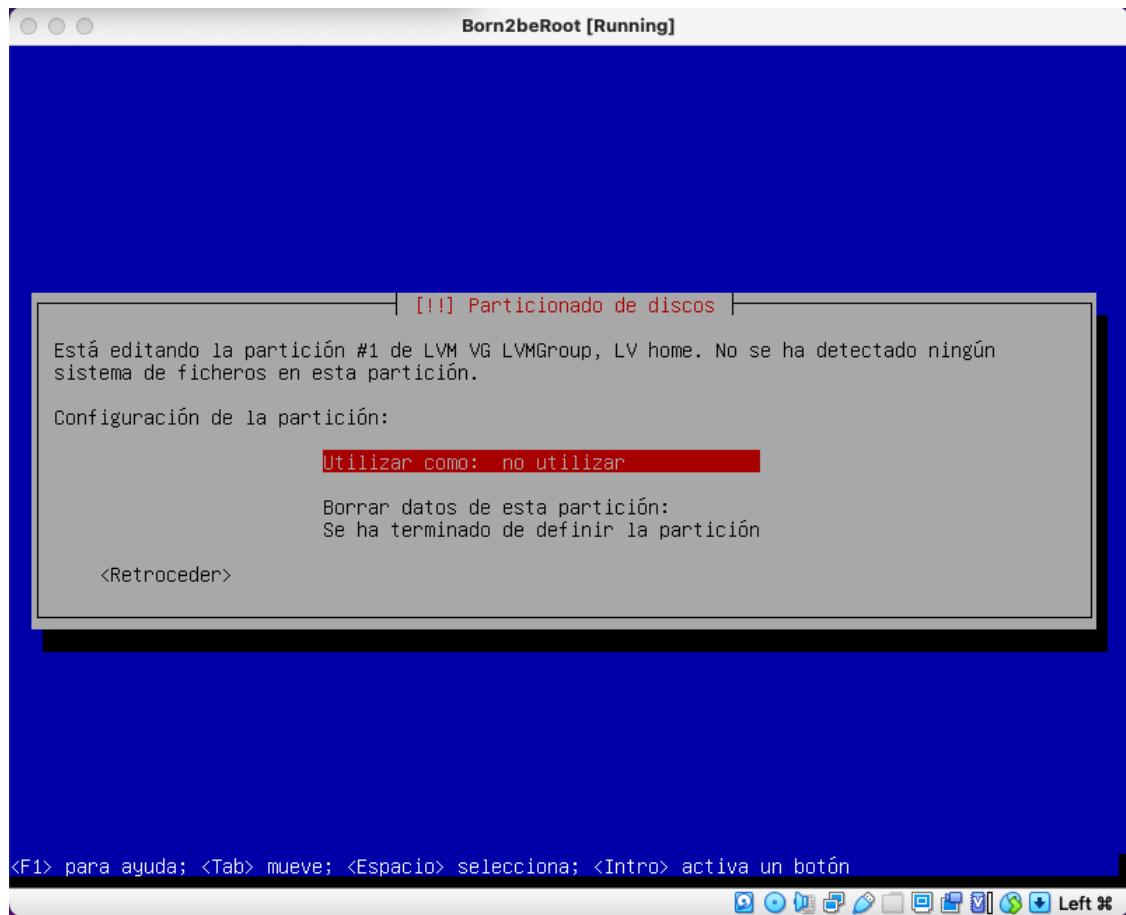
56. Seleccionamos la opción '**Terminar**' de la pantalla de configuración LVM y pulsamos **<Enter>**

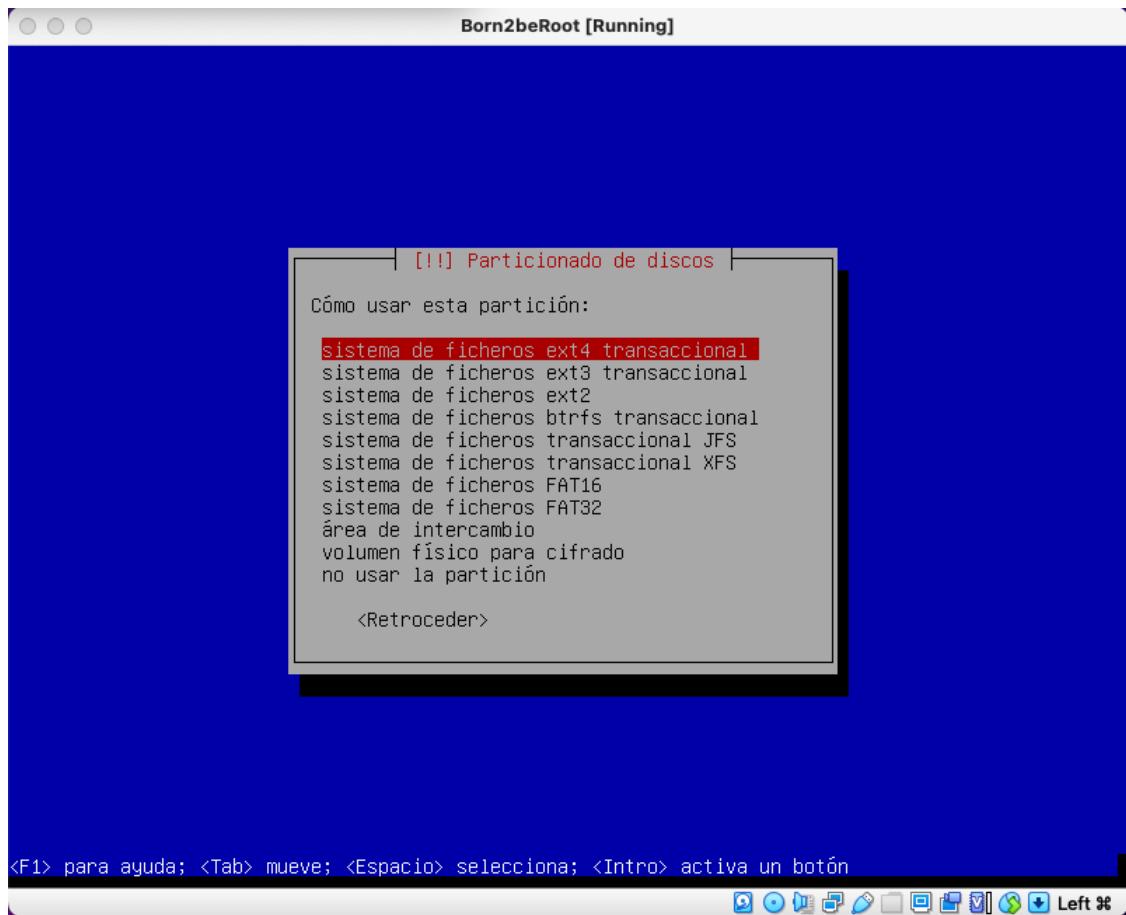


57. Ahora escogemos la partición lógica '#1 998.2 MB', pulsamos <Enter>

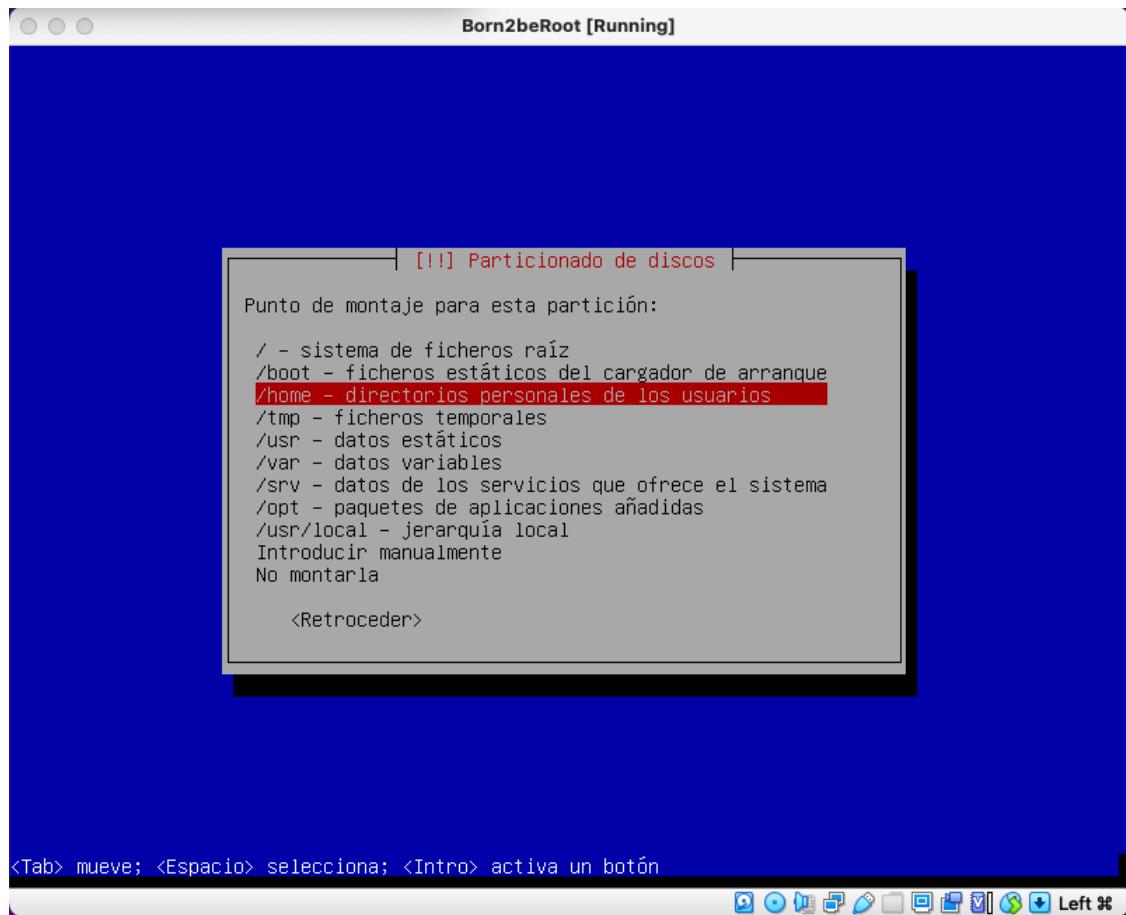


58. Y en la opción ‘Utilizar como’ escogemos ‘sistema de ficheros ext4 transaccional’

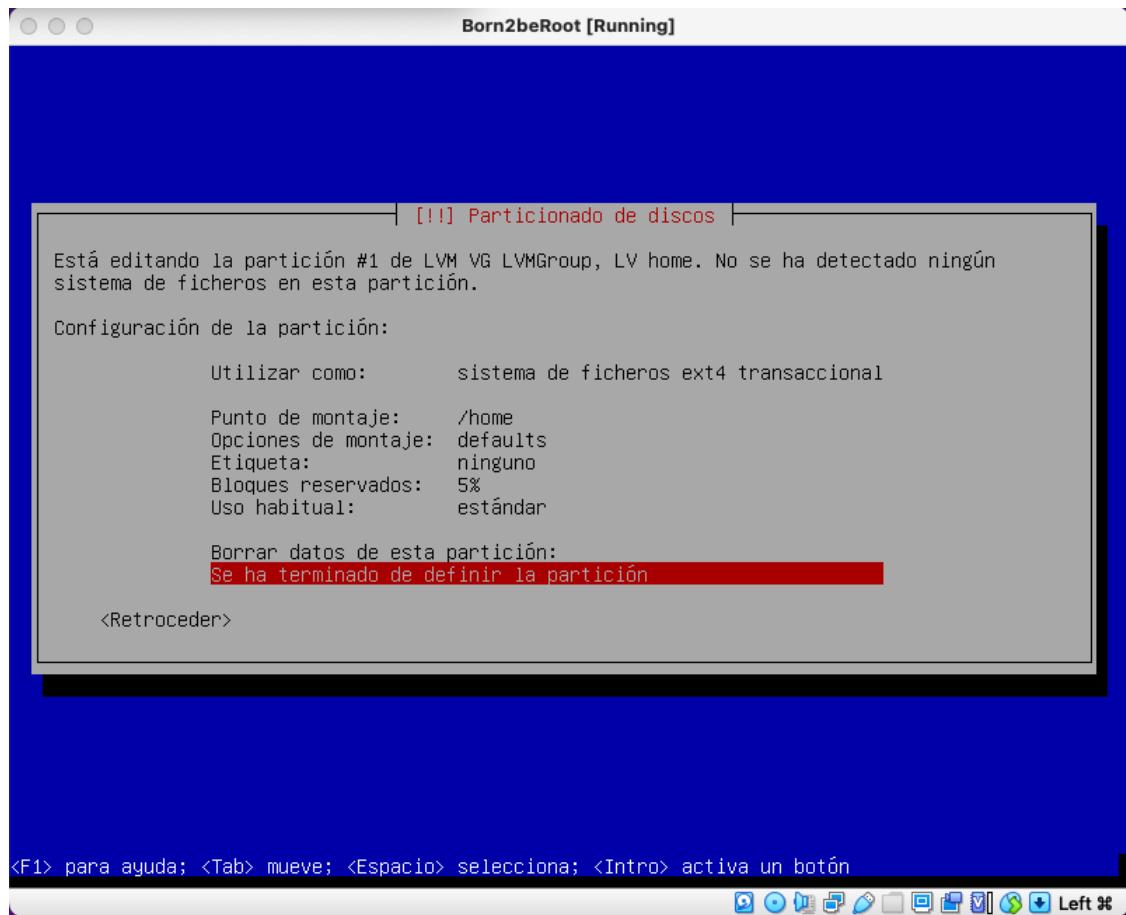




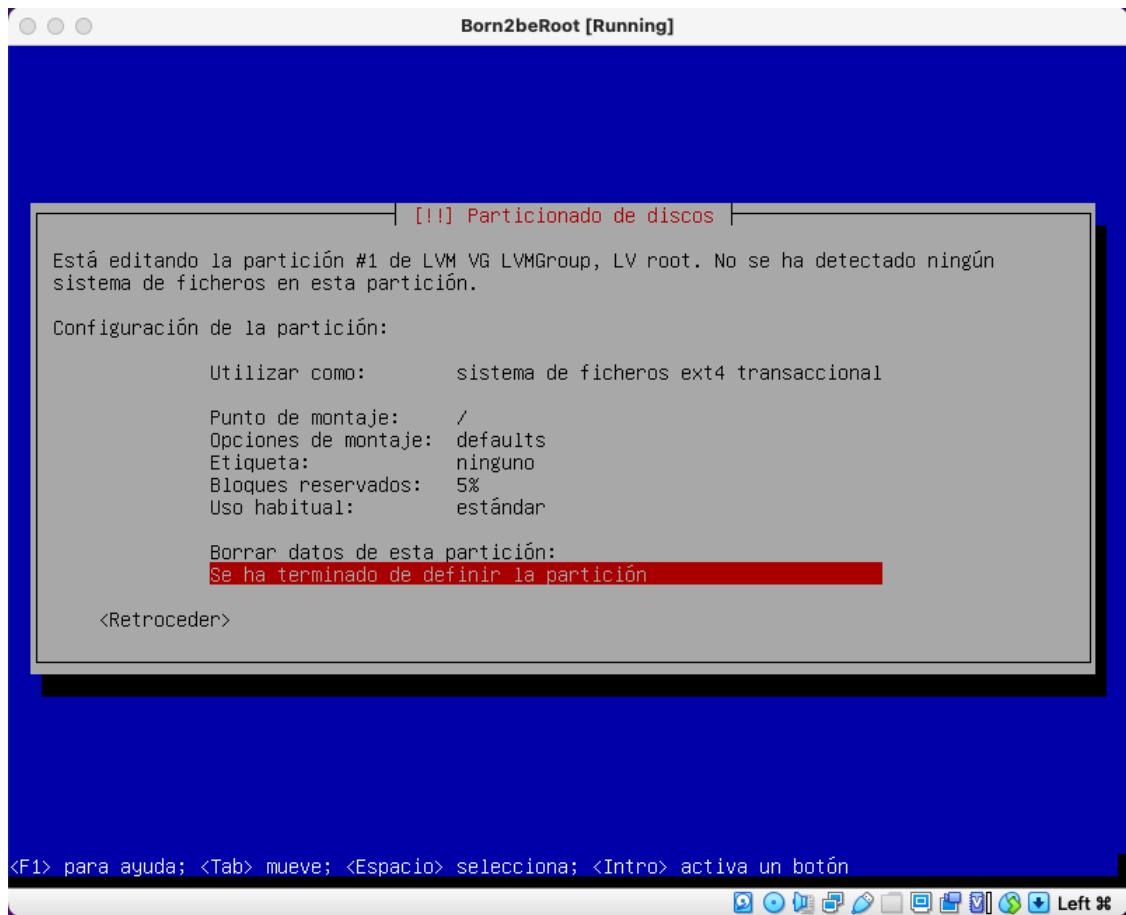
59. En punto de montaje elegimos '**/home - directorios personales de los usuarios**'



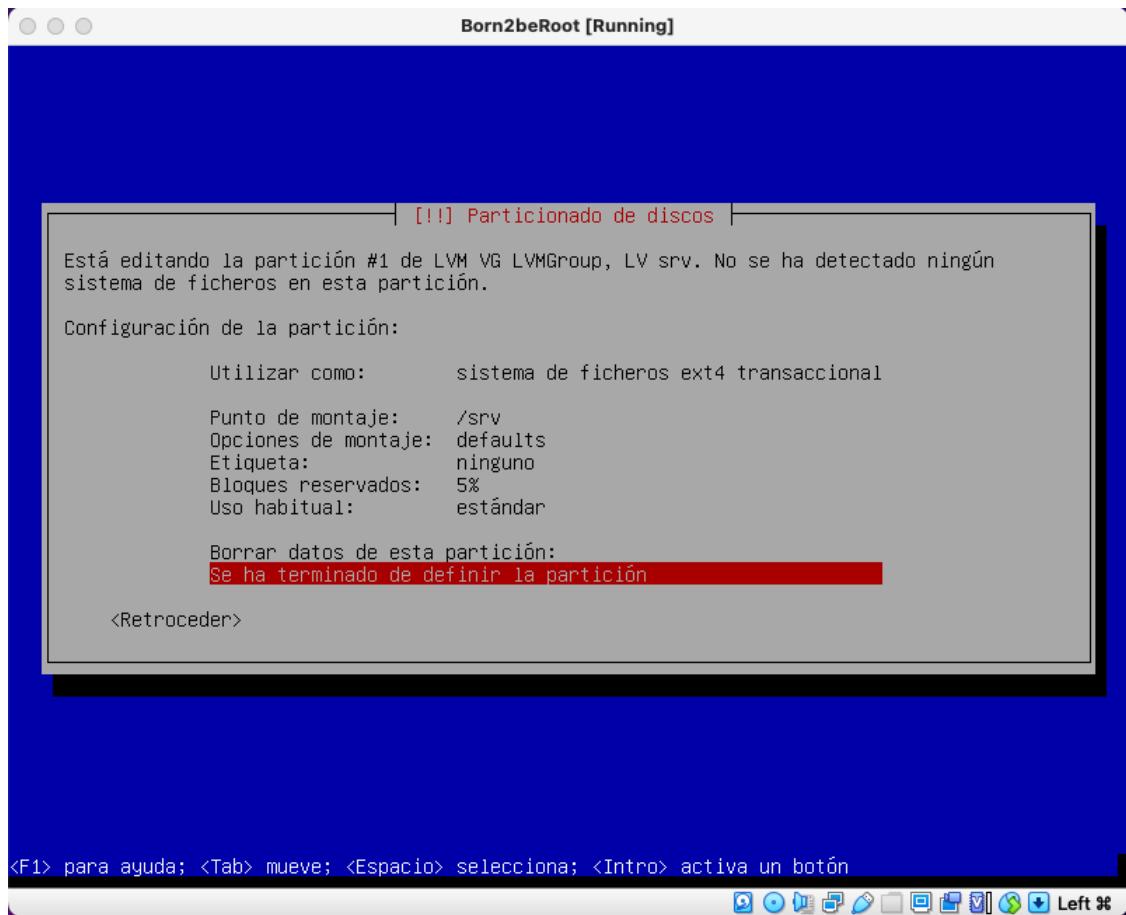
60. Y elegimos '**Se ha terminado de definir la partición**'



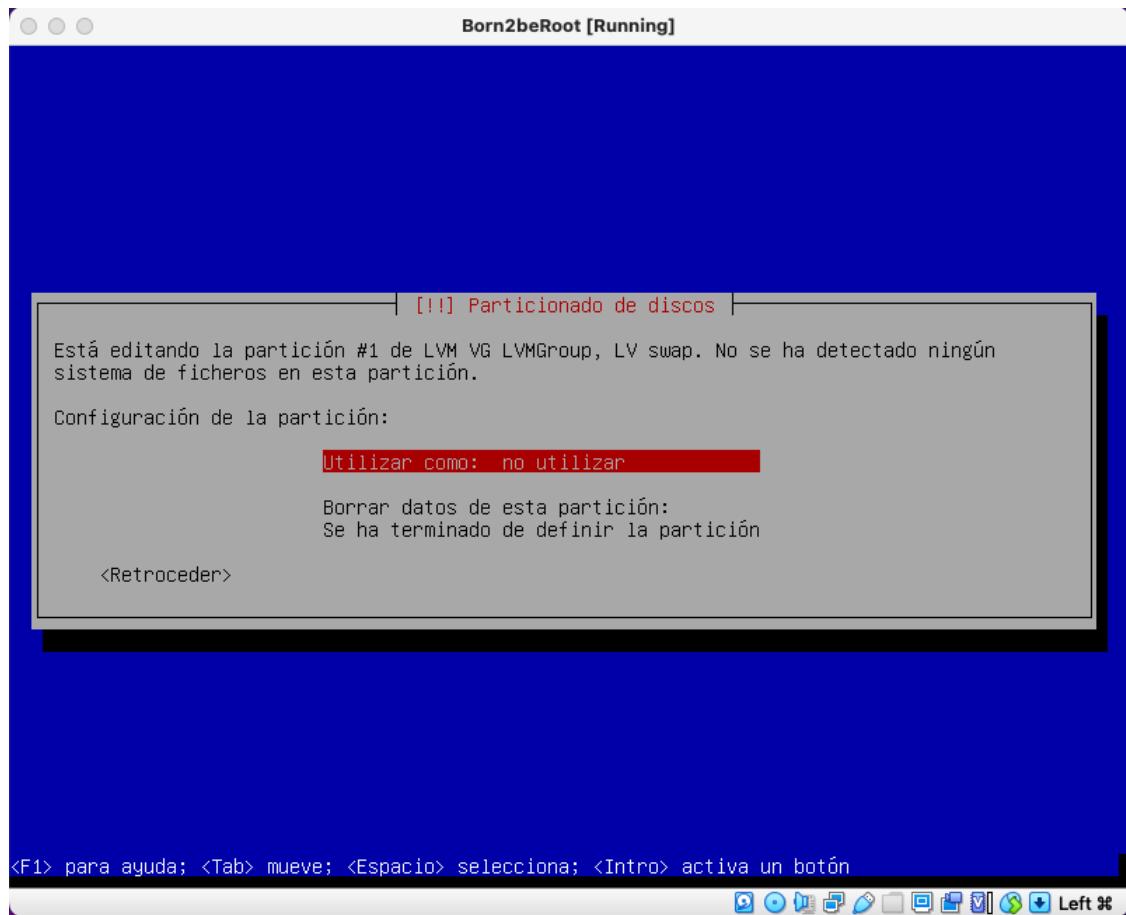
61. Hacemos lo propio con el resto de particiones. A continuación '*/root*'

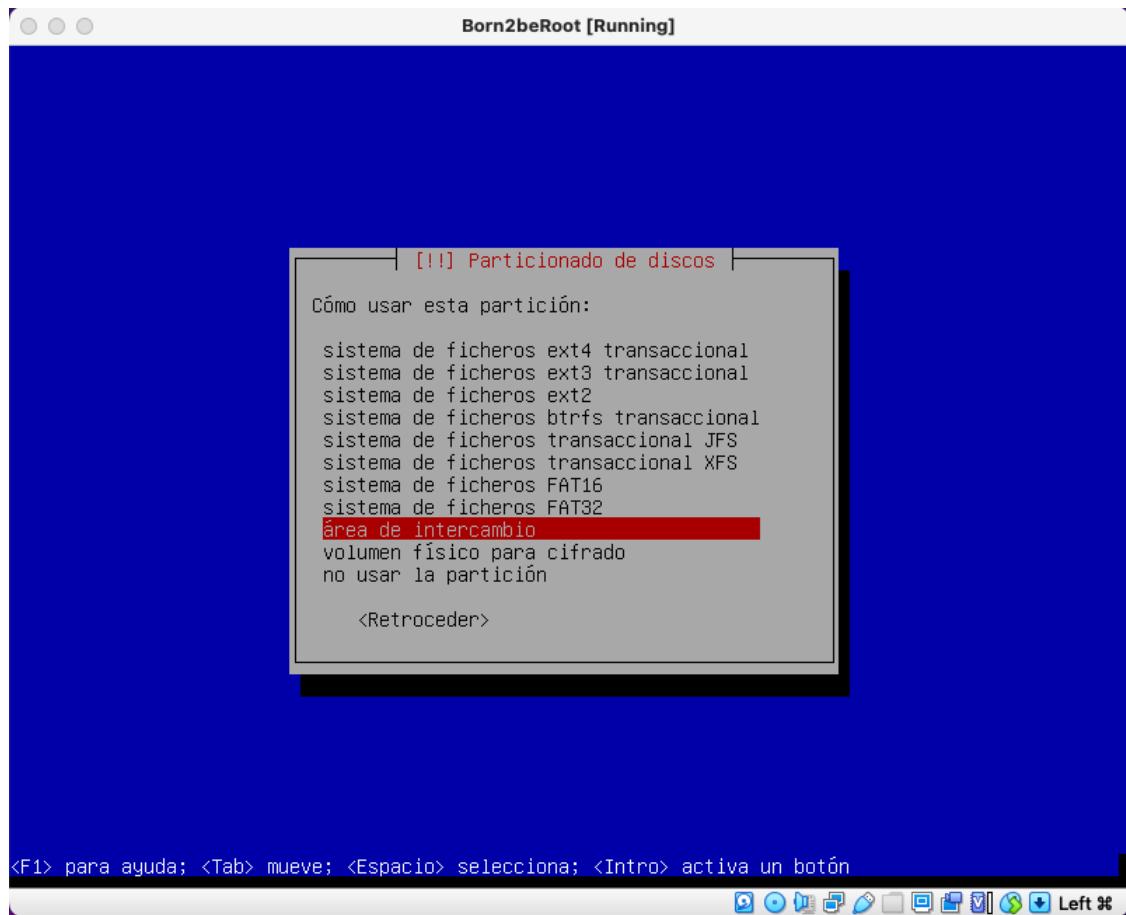


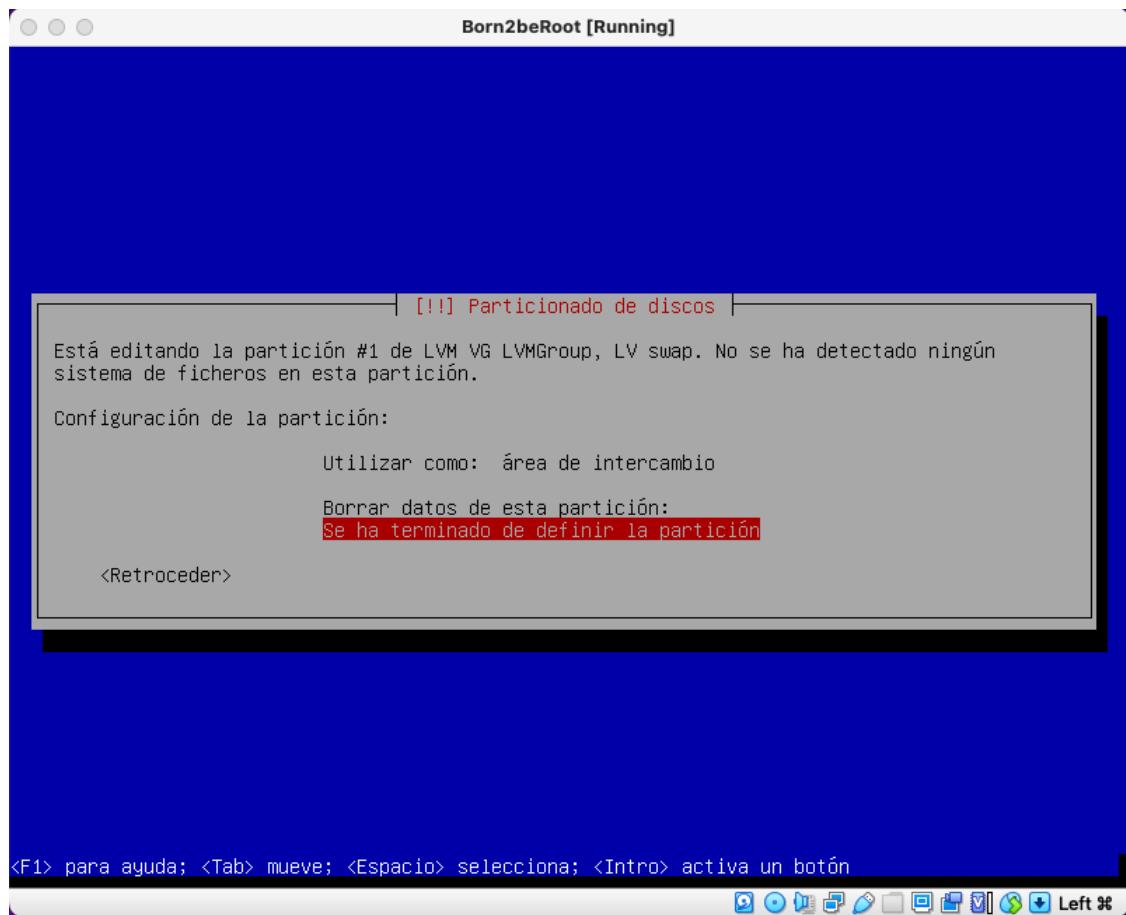
62. Luego '**/srv**'



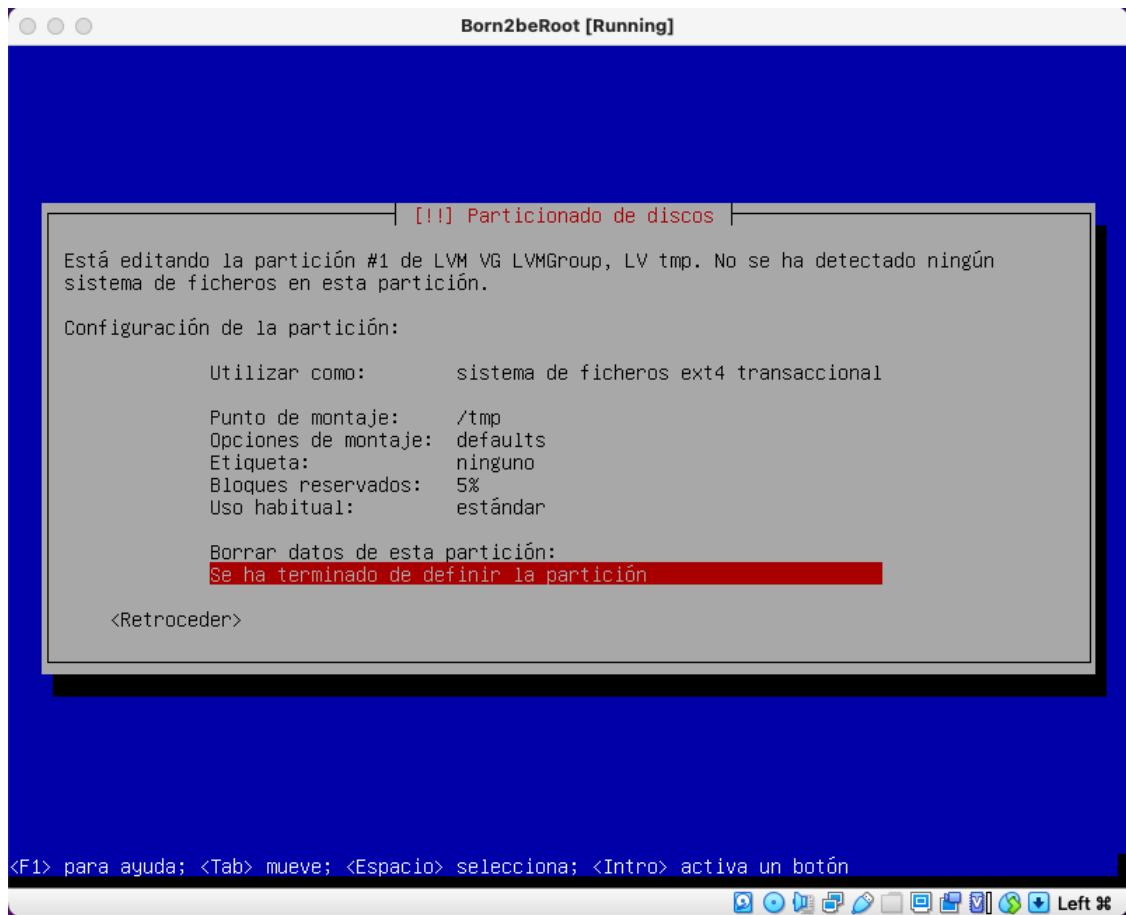
63. Más tarde '*lswap*'



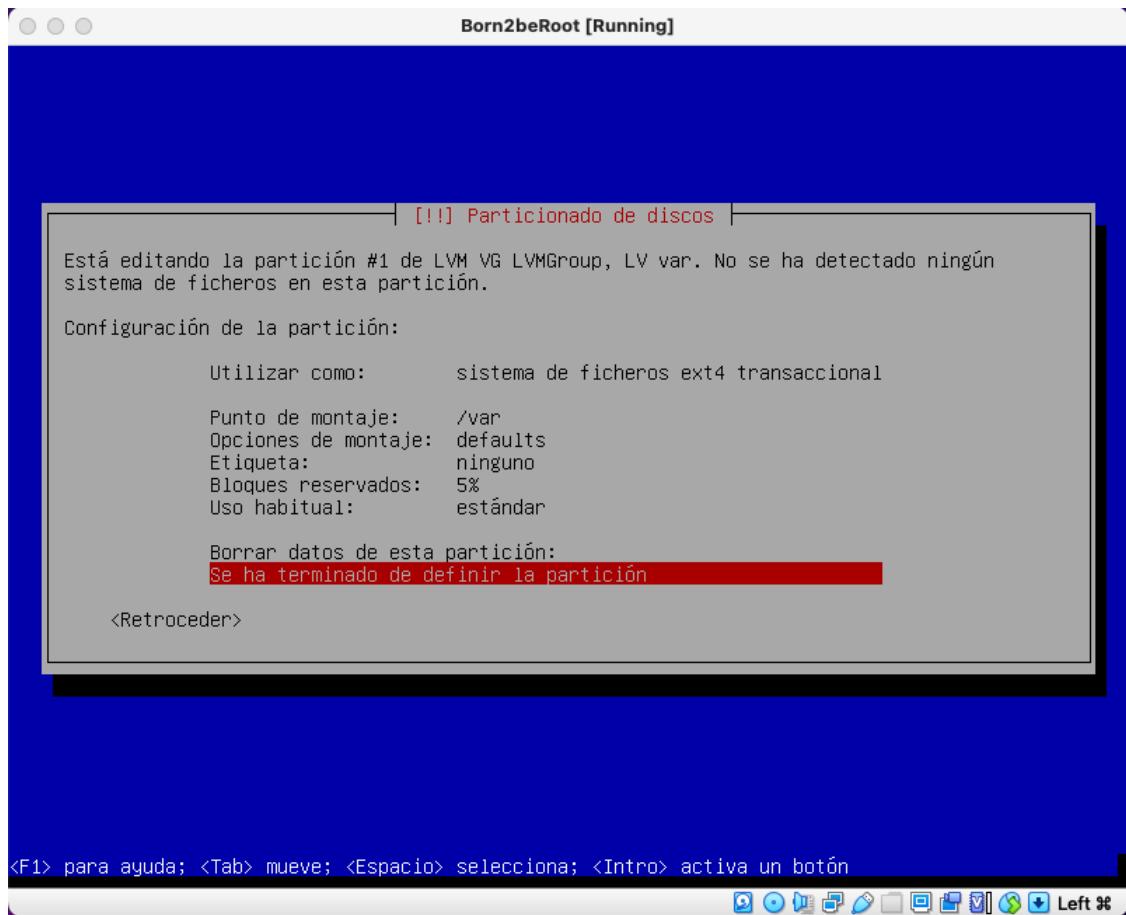




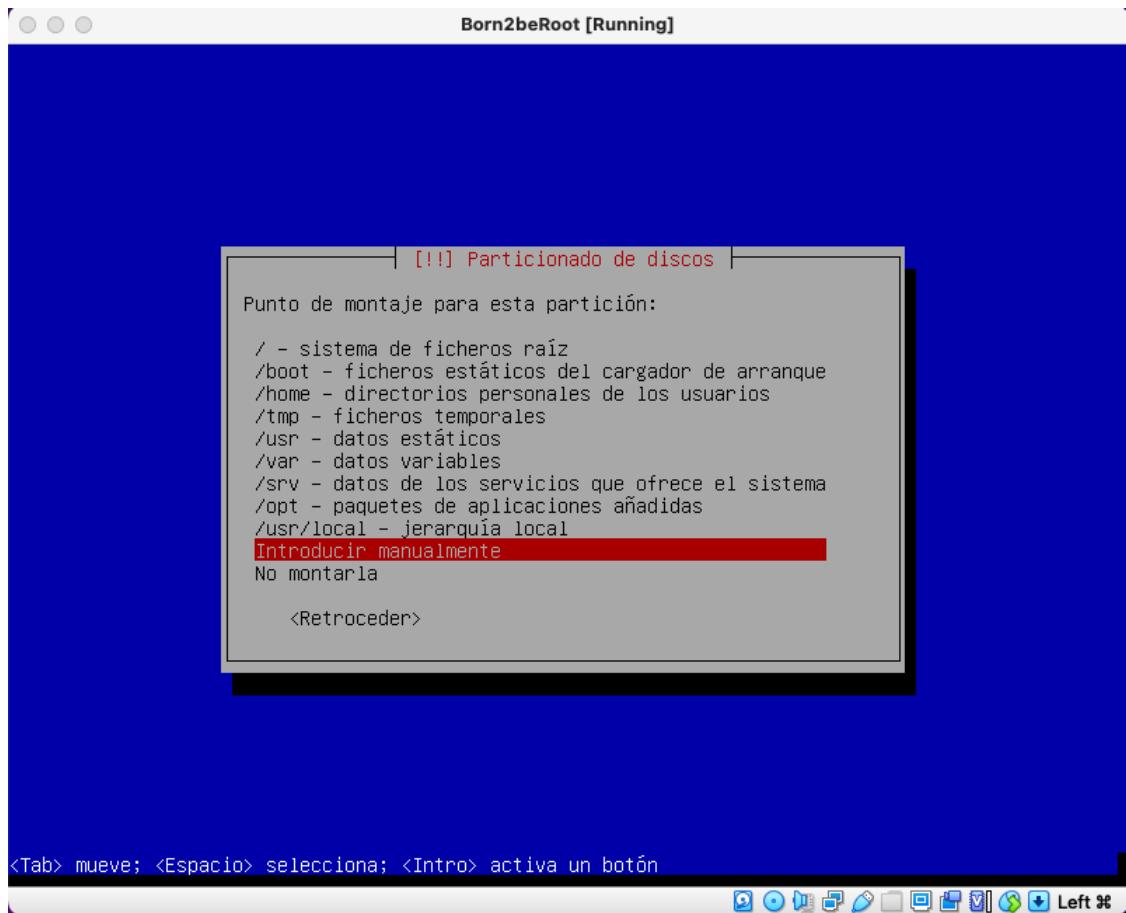
64. Continuamos con '**/tmp**'

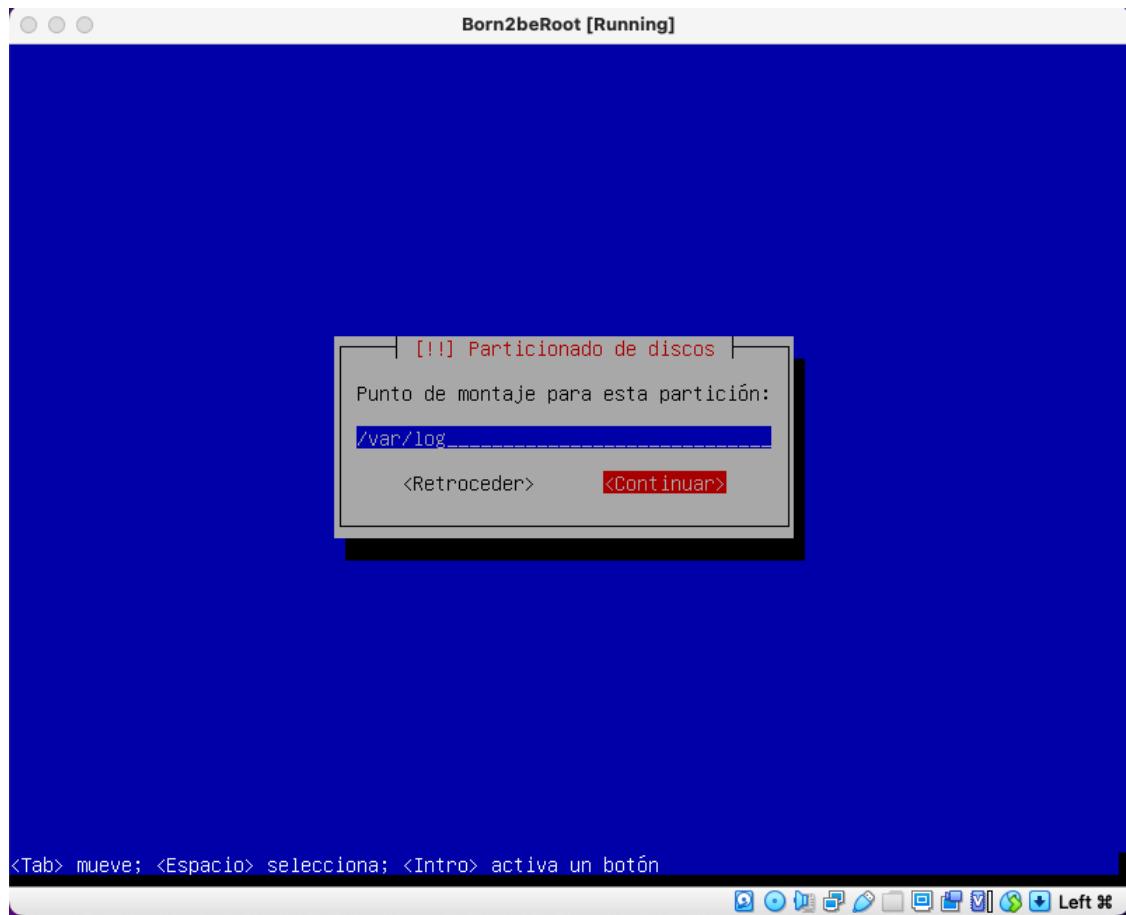


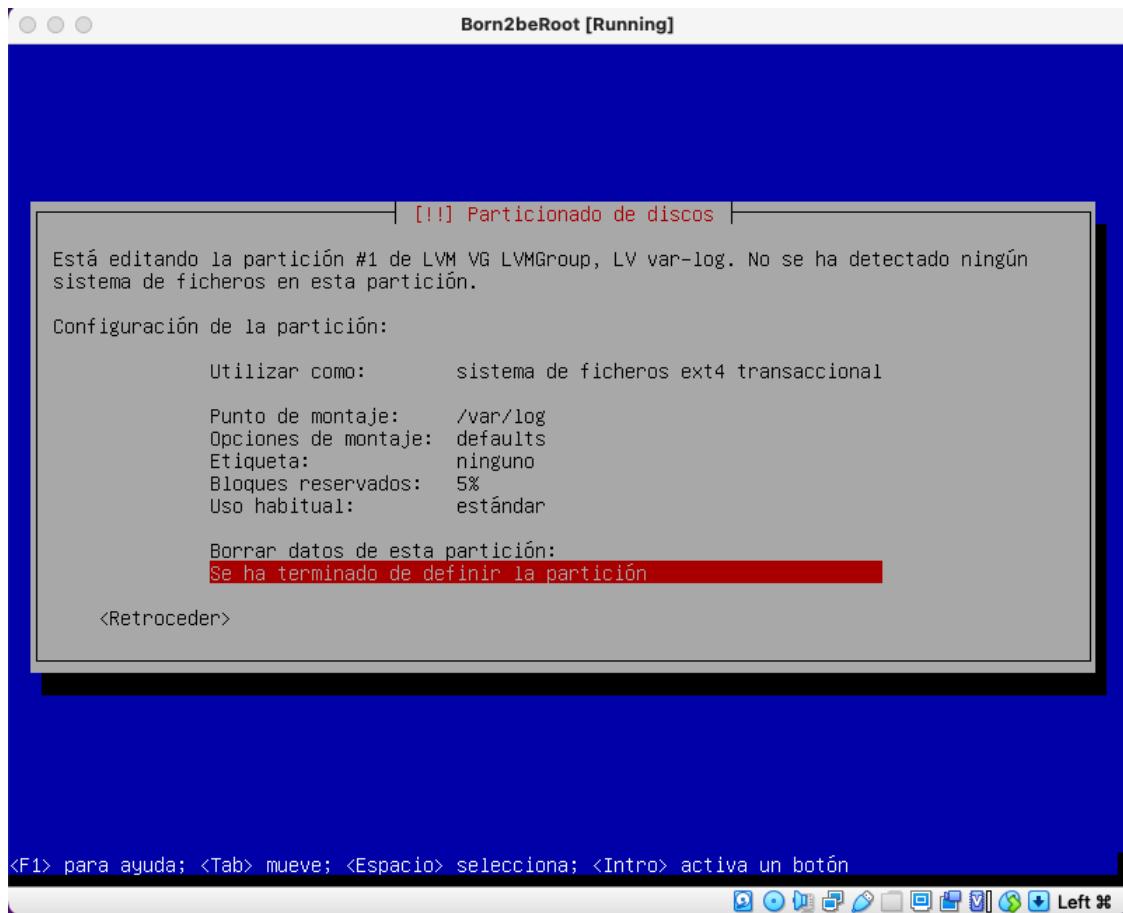
65. Seguimos con '**/var**'



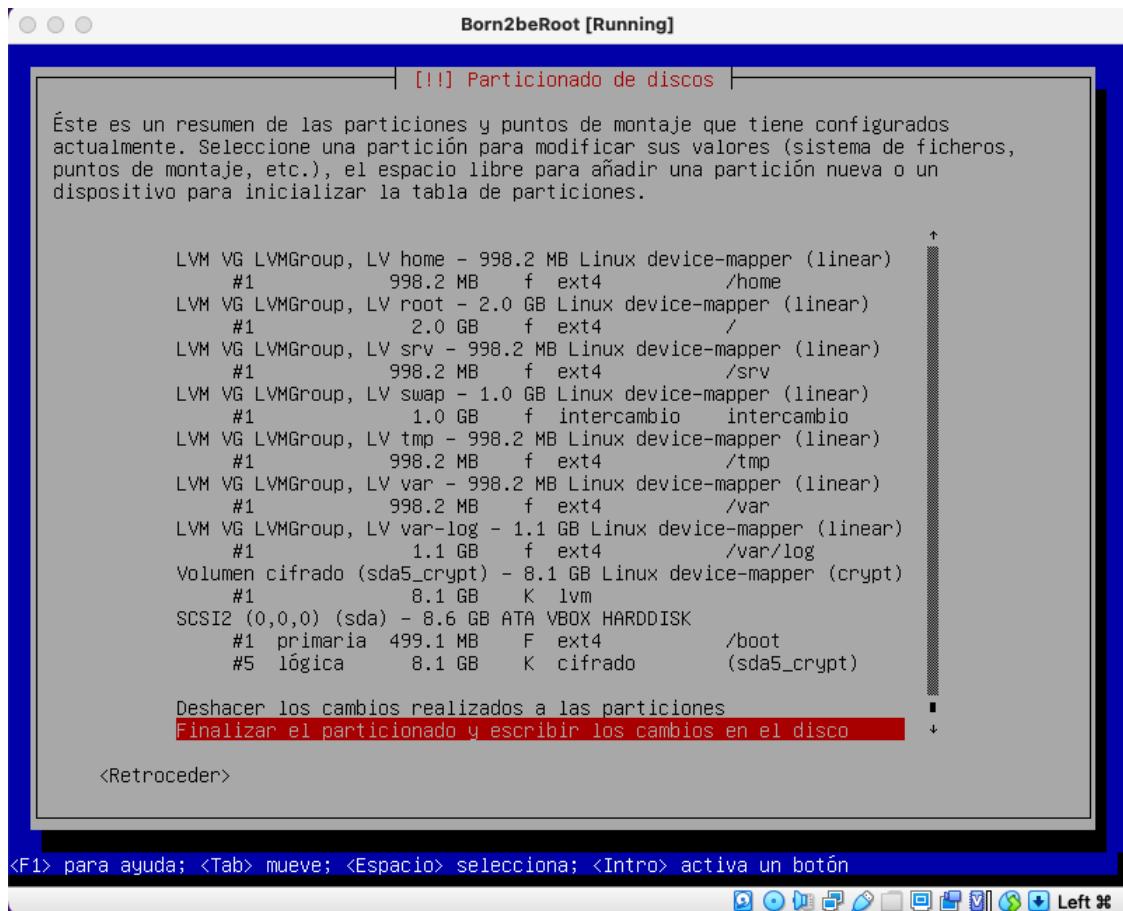
66. Y terminamos con '**/var-log**'



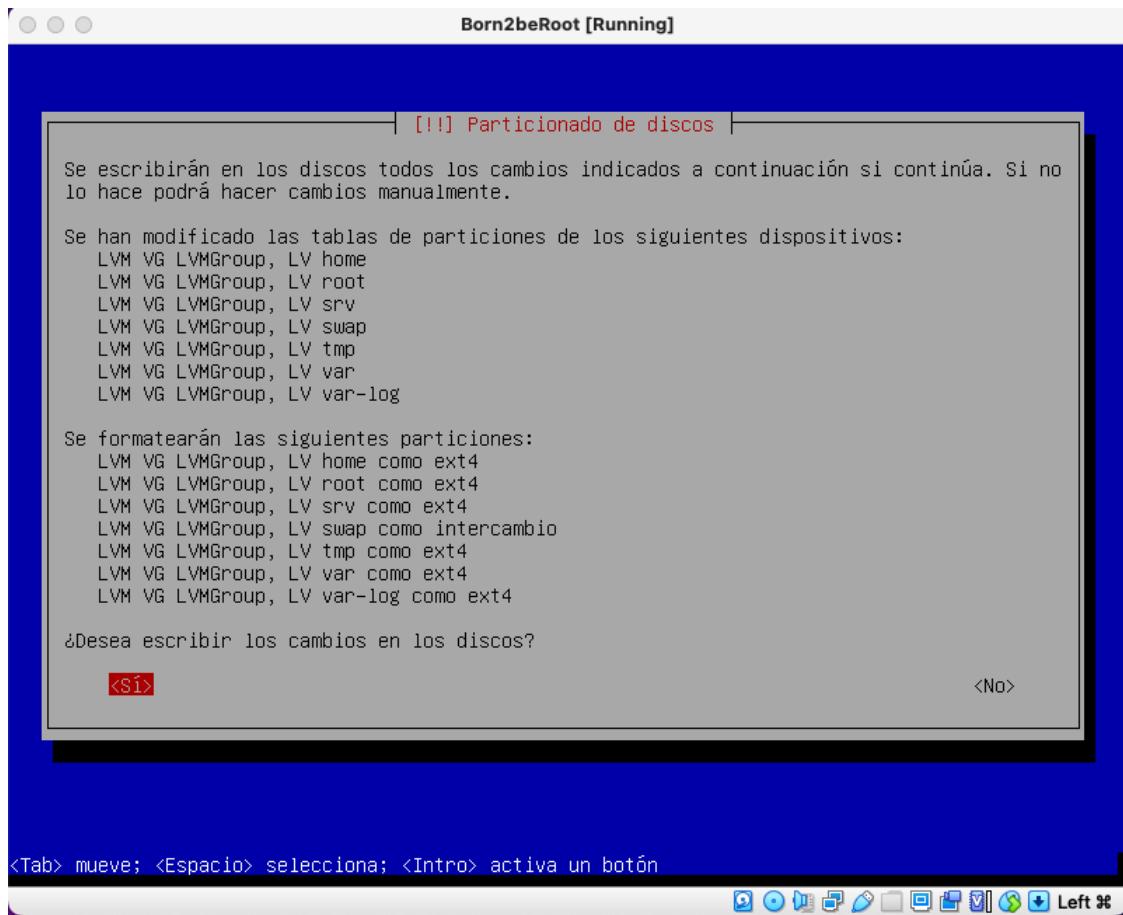




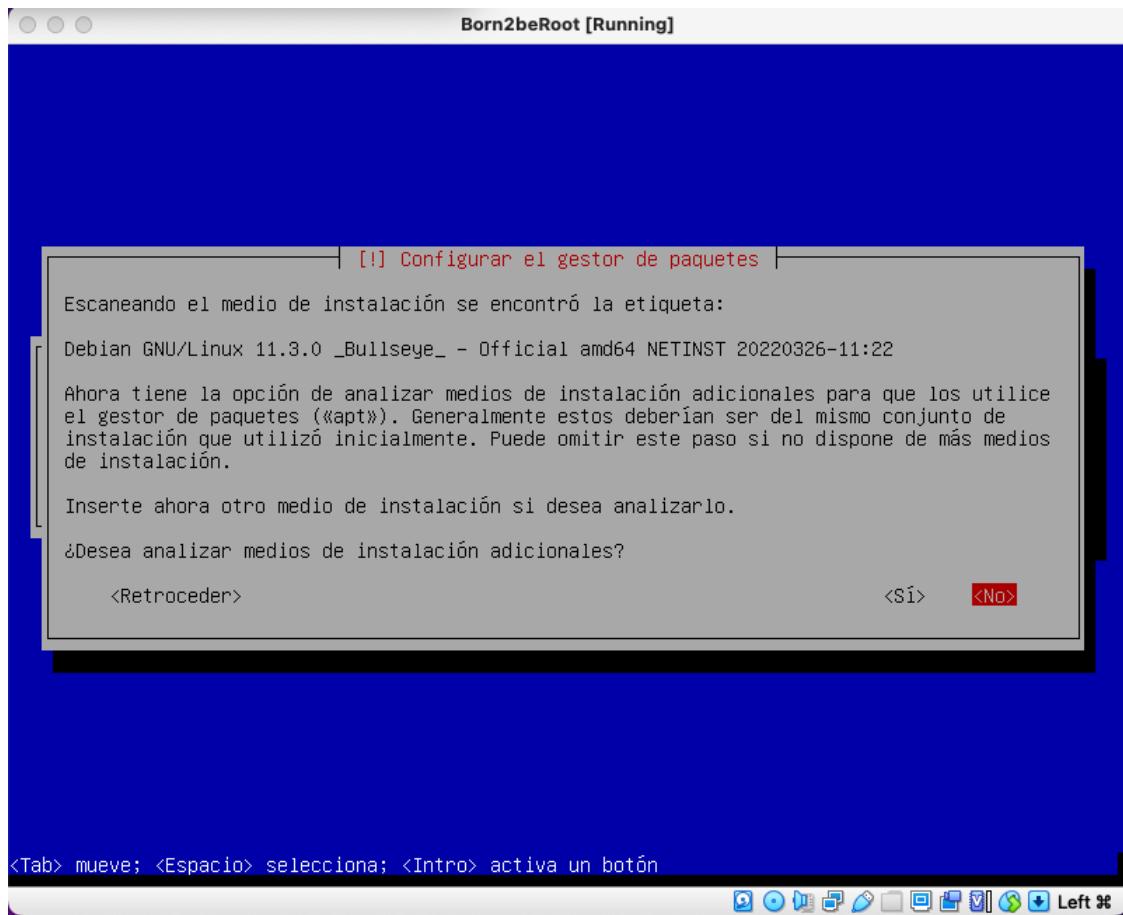
67. Una vez que hemos definido el sistema de archivos y el punto de montaje para todos los volúmenes lógicos, seleccionamos la opción '**Finalizar el particionado y escribir los cambios en el disco**' y pulsamos **<Enter>**



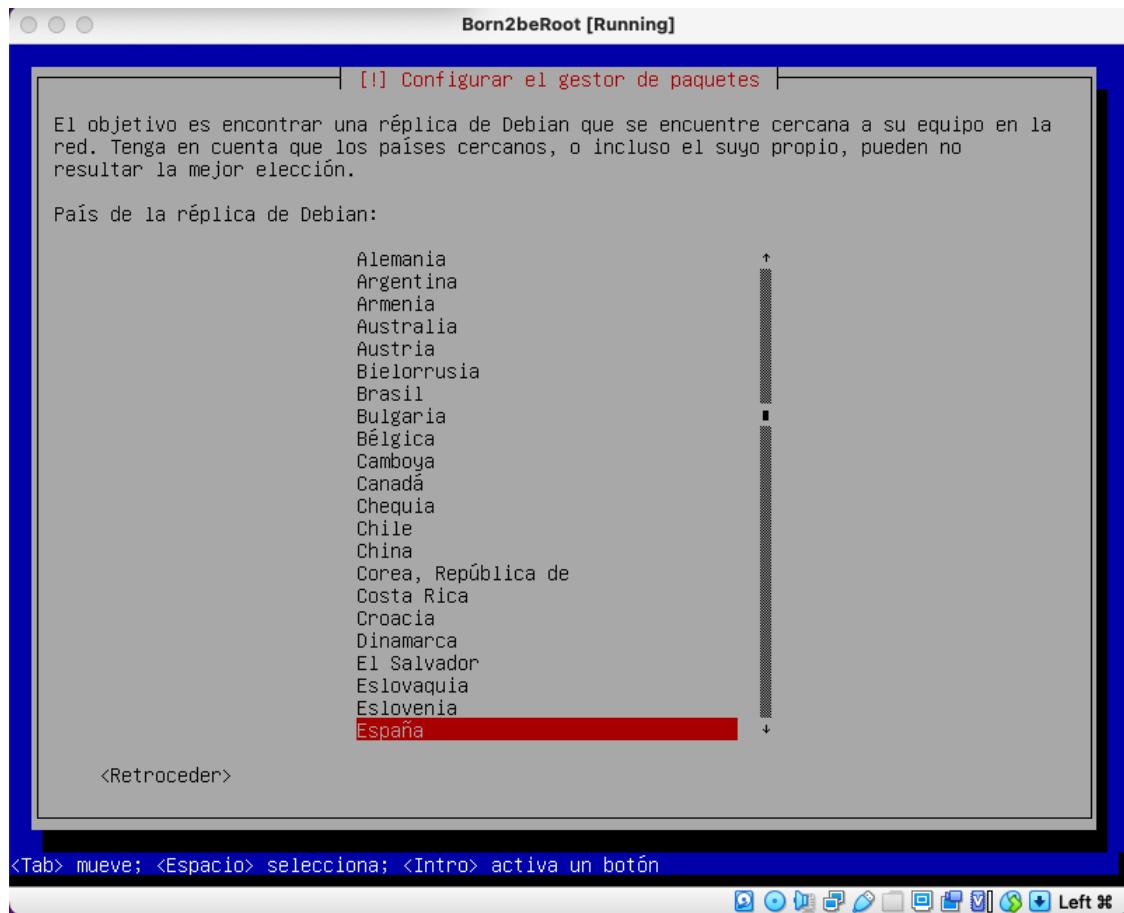
68. Veremos un resumen de los cambios a realizar en el disco. Seleccionamos **<Sí>** y pulsamos **<Enter>**



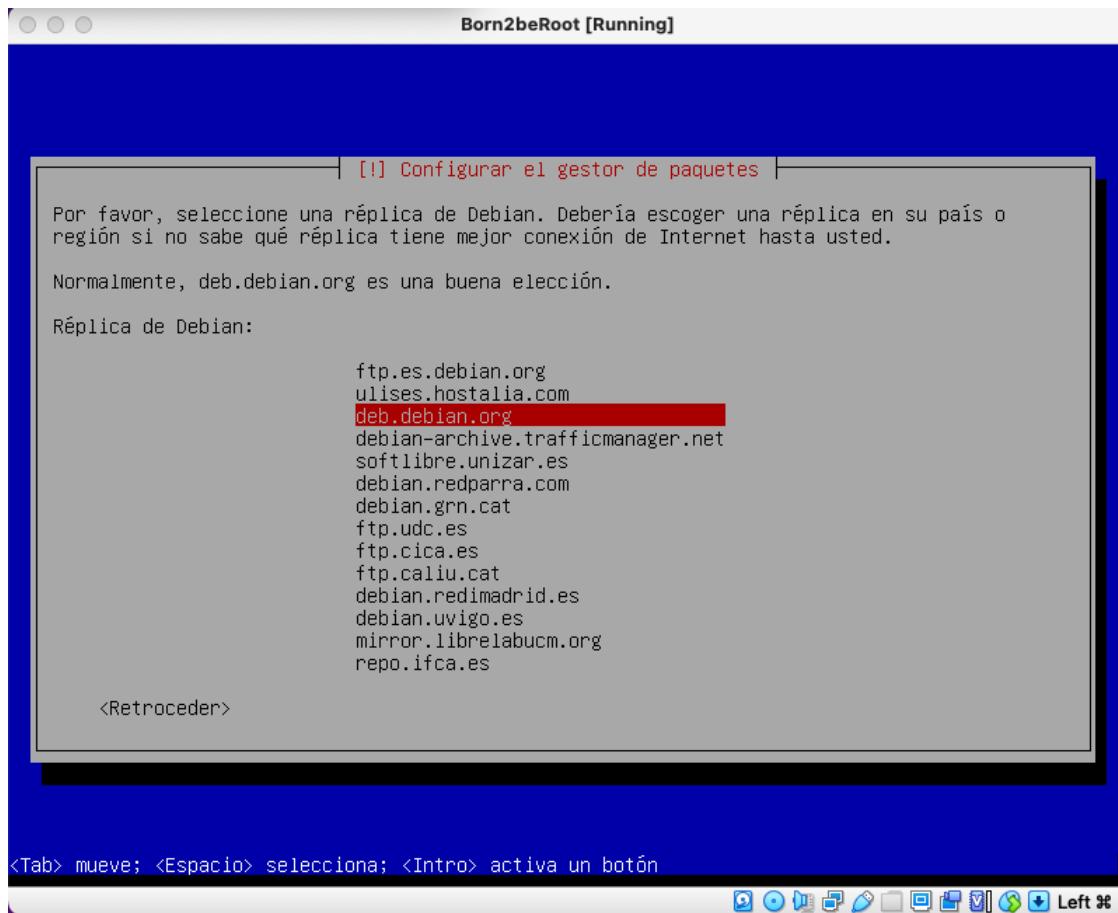
69. En la pantalla '**Configurar el gestor de paquetes**' seleccionamos **<No>** para no analizar medios adicionales. Pulsamos **<Enter>**



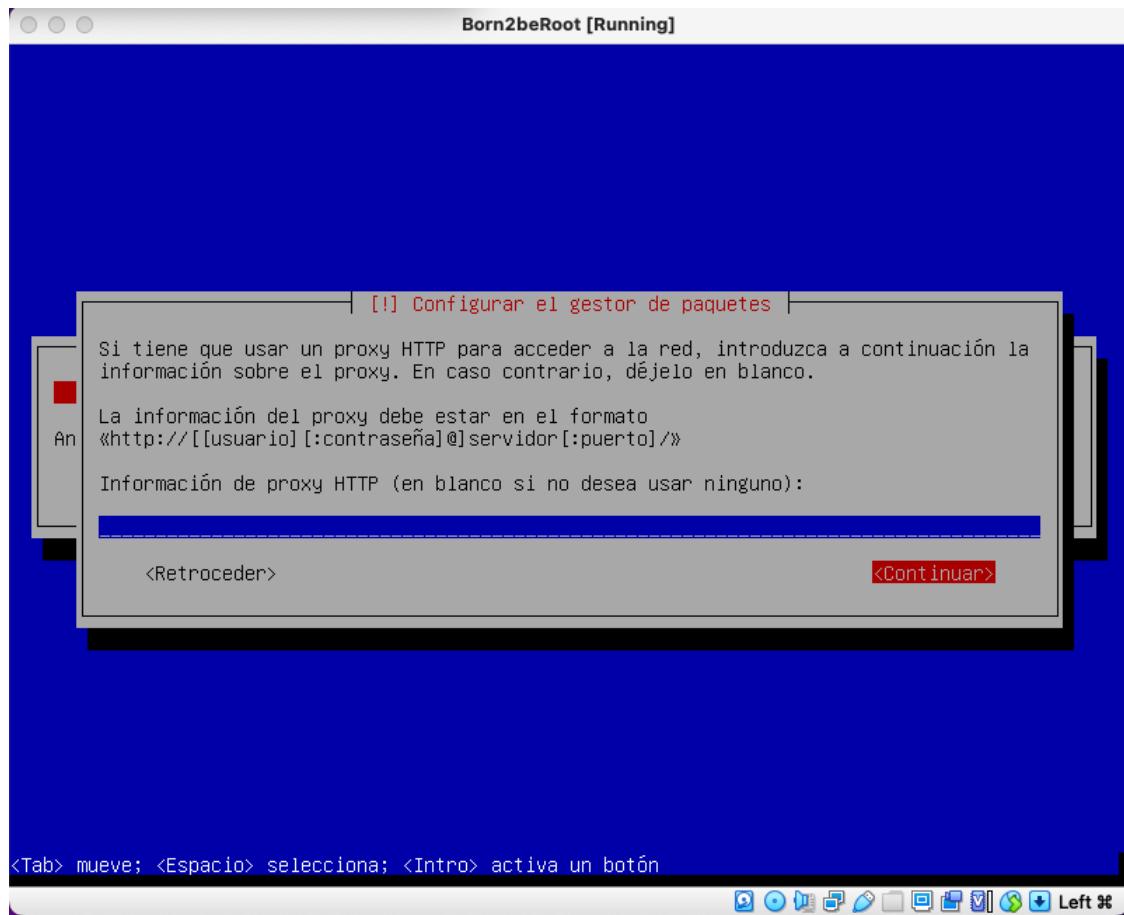
70. Escogemos el país para la réplica de paquetes (en nuestro caso España)



71. Elegimos la réplica de Debian en 'deb.debian.org'



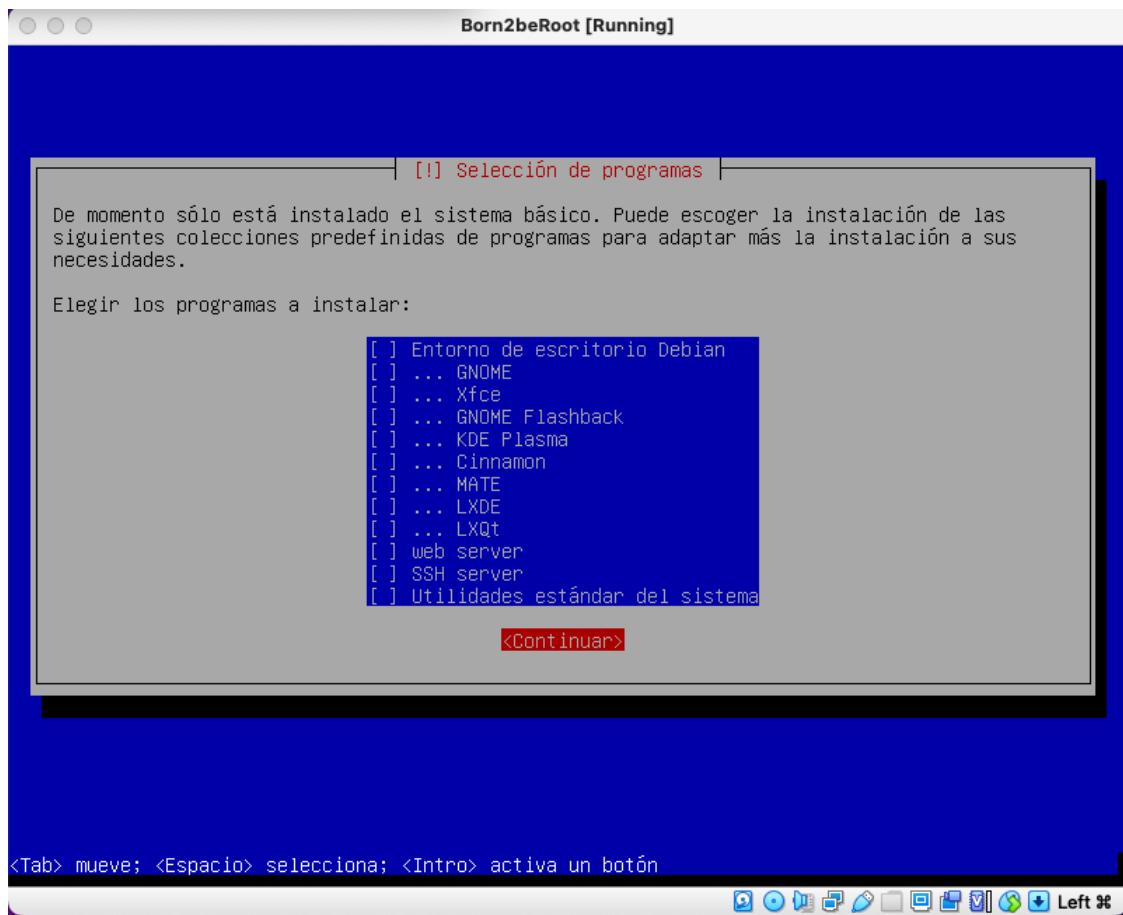
72. Dejamos en blanco el campo de nombre del proxy HTTP y seleccionamos [**<Continuar>**](#)



73. A la pregunta de si deseamos participar en la encuesta sobre el uso de los paquetes escogemos <**No**>



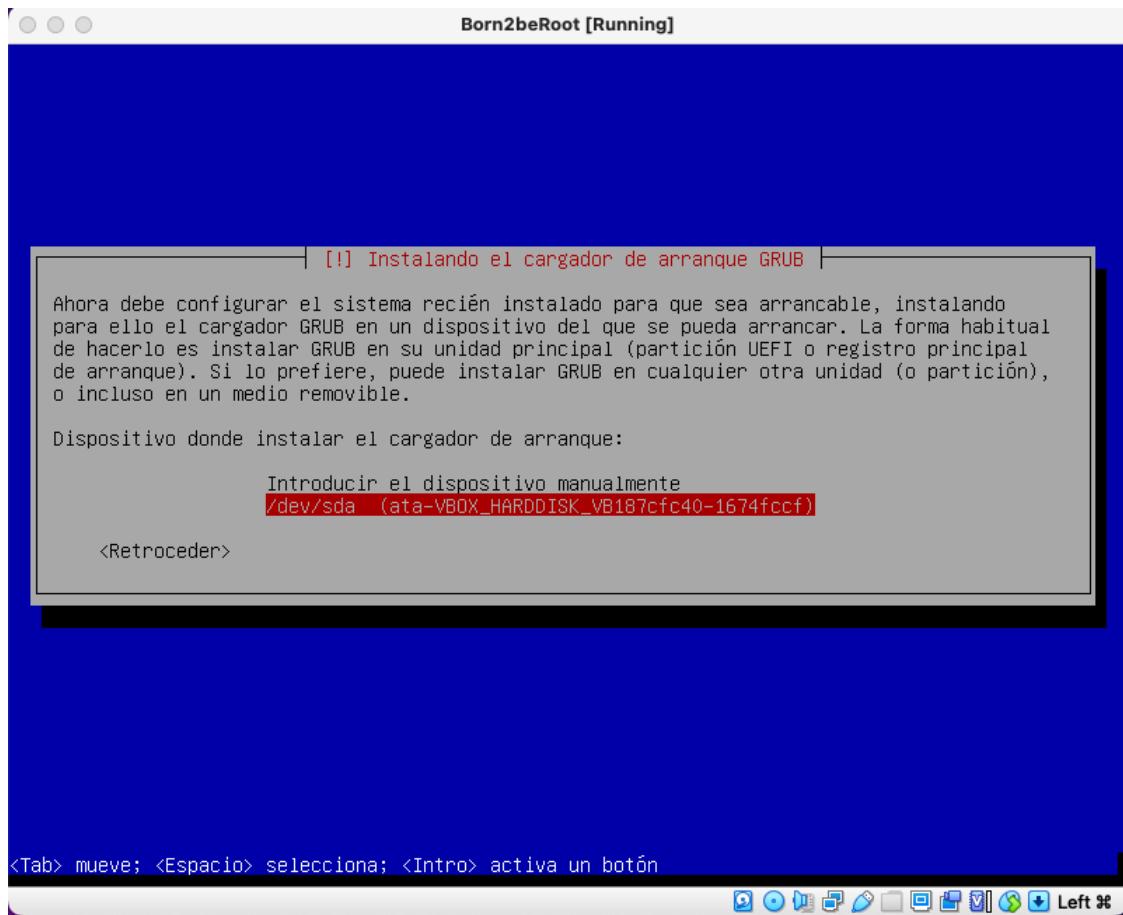
74. Desmarcamos todas las opciones de la ventana ‘Selección de programas’ (los configuraremos más tarde)



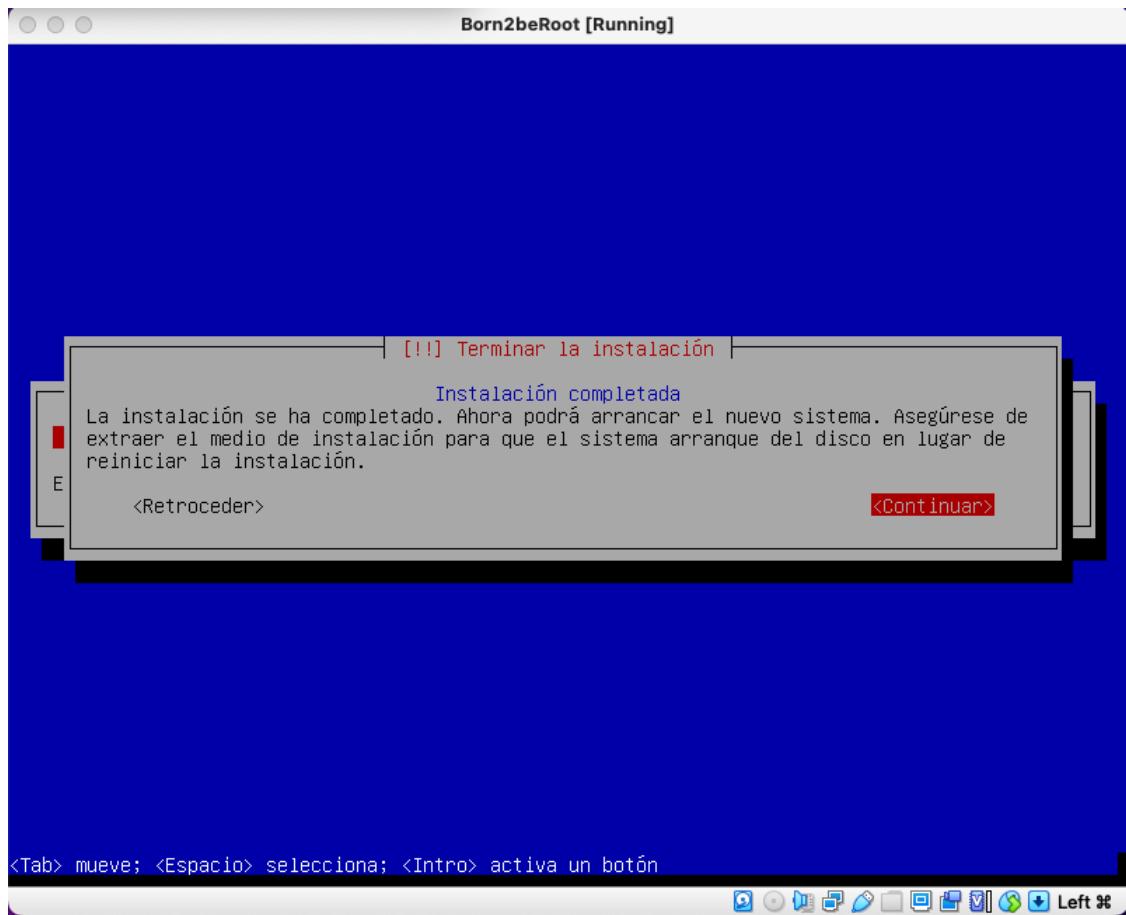
75. En la pantalla '**Instalando el cargador de arranque GRUB**' seleccionamos **<Sí>** para instalar GRUB en la unidad principal



76. Y seleccionamos '**/dev/sda (ata-VBOX_HARDDISK_...)**', para terminar pulsando **<Enter>**



77. Llegamos al final de la instalación del Sistema Operativo con la pantalla ‘**Terminar la Instalación**’ en la que se nos indica que debemos extraer el medio de instalación.
Seleccionamos [**<Continuar>**](#)



78. El sistema se reinicia. Introducimos la clave de descriptado del volumen y luego nos validamos con nuestro usuario. Ejecutamos el comando lsblk y obtenemos el listado de volúmenes y particiones solicitado.

```
fsanchez@fsanchez42:~$ lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda        8:0    0   8G  0 disk
└─sda1     8:1    0  476M 0 part  /boot
└─sda2     8:2    0   1K  0 part
└─sda5     8:5    0  7,5G 0 part
  └─sda5_crypt 254:0  0  7,5G 0 crypt
    ├─LVMGroup-root 254:1  0 1,9G 0 lvm   /
    ├─LVMGroup-swap 254:2  0 976M 0 lvm   [SWAP]
    ├─LVMGroup-home 254:3  0 952M 0 lvm   /home
    ├─LVMGroup-var  254:4  0 952M 0 lvm   /var
    ├─LVMGroup-srv  254:5  0 952M 0 lvm   /srv
    ├─LVMGroup-tmp  254:6  0 952M 0 lvm   /tmp
    └─LVMGroup-var--log 254:7  0 1008M 0 lvm   /var/log
sr0       11:0   1 1024M 0 rom

fsanchez@fsanchez42:~$
```

▼ Instalar Sudo

1. Nos validamos como root

- a. `su`

```
fsanchez@fsanchez42:/home$ su
Contraseña:
root@fsanchez42:/home# _
```

2. Ejecutamos los comandos apt-get para actualizar el repositorio y luego instalar sudo

- a. `apt-get update`

```

Born2beRoot [Running]
root@fsanchez42:/home# apt-get update
Obj:1 http://deb.debian.org/debian bullseye InRelease
Des:2 http://security.debian.org/debian-security bullseye-security InRelease [44,1 KB]
Des:3 http://deb.debian.org/debian bullseye-updates InRelease [39,4 KB]
Des:4 http://security.debian.org/debian-security bullseye-security/main Sources [121 KB]
Des:5 http://security.debian.org/debian-security bullseye-security/main amd64 Packages [146 KB]
Des:6 http://security.debian.org/debian-security bullseye-security/main Translation-en [90,3 KB]
Descargados 440 kB en 0s (1.089 kB/s)
Leyendo lista de paquetes... Hecho
root@fsanchez42:/home# _
```

b. `apt-get upgrade`

```

Born2beRoot [Running]
root@fsanchez42:/home# apt-get upgrade
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Calculando la actualización... Hecho
Se actualizarán los siguientes paquetes:
  libss1.1
1 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 1.558 kB de archivos.
Se utilizarán 0 B de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://security.debian.org/debian-security bullseye-security/main amd64 libss1.1.1 amd64 1.1.1n-0+deb11u2 [1.558 kB]
Descargados 1.558 kB en 0s (9.595 kB/s)
Preconfigurando paquetes ...
(Leyendo la base de datos ... 23944 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../libss1.1.1_1.1.1n-0+deb11u2_amd64.deb ...
Desempaquetando libss1.1:amd64 (1.1.1n-0+deb11u2) sobre (1.1.1n-0+deb11u1) ...
Configurando libss1.1:amd64 (1.1.1n-0+deb11u2) ...
Procesando disparadores para libc-bin (2.31-13+deb11u3) ...
root@fsanchez42:/home# _
```

c. `apt install sudo`

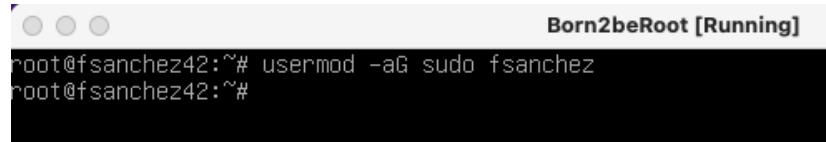
```

Born2beRoot [Running]
root@fsanchez42:/home# apt install sudo
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes NUEVOS:
  sudo
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 1.059 kB de archivos.
Se utilizarán 4.699 kB de espacio de disco adicional después de esta operación.
Des:1 http://deb.debian.org/debian bullseye/main amd64 sudo amd64 1.9.5p2-3 [1.059 kB]
Descargados 1.059 kB en 0s (6.537 kB/s)
Seleccionando el paquete sudo previamente no seleccionado.
(Leyendo la base de datos ... 23944 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../sudo_1.9.5p2-3_amd64.deb ...
Desempaquetando sudo (1.9.5p2-3) ...
Configurando sudo (1.9.5p2-3) ...
root@fsanchez42:/home# _
```

d. Obtenemos privilegios de root para unir el usuario al grupo sudo

i. `su -`

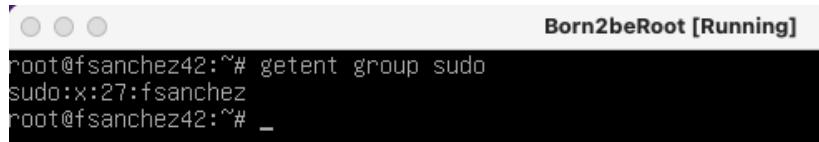
ii. `usermod -aG sudo fsanchez` (append Group sudo)



```
Born2beRoot [Running]
root@fsanchez42:~# usermod -aG sudo fsanchez
root@fsanchez42:~#
```

iii. Comprobamos que el usuario está dentro del grupo sudo con

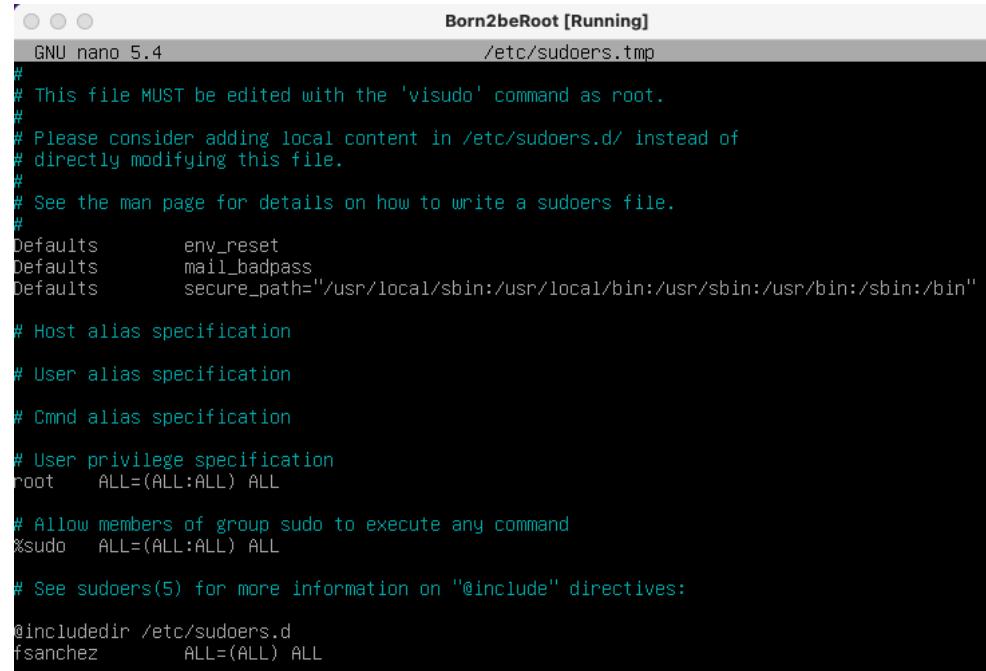
1. `getent group sudo`



```
Born2beRoot [Running]
root@fsanchez42:~# getent group sudo
sudo:x:27:fsanchez
root@fsanchez42:~#
```

iv. Editamos el archivo visudo para incluir a nuestro usuario

1. `sudo visudo`



```
GNU nano 5.4                               /etc/sudoers.tmp
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults      env_reset
Defaults      mail_badpass
Defaults      secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "@include" directives:

@includedir /etc/sudoers.d
fsanchez      ALL=(ALL) ALL
```

▼ Crear usuarios y grupos, cambiar el nombre del host

1. Para crear un nuevo usuario utilizamos el comando adduser

a. `sudo adduser <user_name>`

2. Podemos crear grupos con el comando `sudo groupadd user42`

3. Y le añadimos nuestro usuario `sudo usermod -aG user42 <your_user>`

```

fsanchez@fsanchez42:~$ sudo groupadd user42
fsanchez@fsanchez42:~$ usermod -aG user42 fsanchez
-bash: usermod: orden no encontrada
fsanchez@fsanchez42:~$ sudo usermod -aG user42 fsanchez
fsanchez@fsanchez42:~$ getent group user42
user42:x:1001:fsanchez
fsanchez@fsanchez42:~$
```

4. Para cambiar el nombre del host usamos el comando `hostnamectl`
 - a. Con `hostnamectl` vemos la información actual del host
 - b. luego ejecutamos `sudo hostnamectl set-hostname <nuevo_hostname>`
 - c. Es conveniente también modificar el nombre del host en el archivo `/etc/hosts`, así que ejecutamos `sudo nano /etc/hosts` y cambiamos la entrada `127.0.0.1 anterior_hostname` por `127.0.0.1 nuevo_hostname`

```

GNU nano 5.4                               /etc/hosts *
127.0.0.1      localhost
127.0.1.1      nuevo_hostname_
# The following lines are desirable for IPv6 capable hosts
::1      localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

The terminal window shows the `/etc/hosts` file being edited with `sudo nano`. The file contains the following content:

```

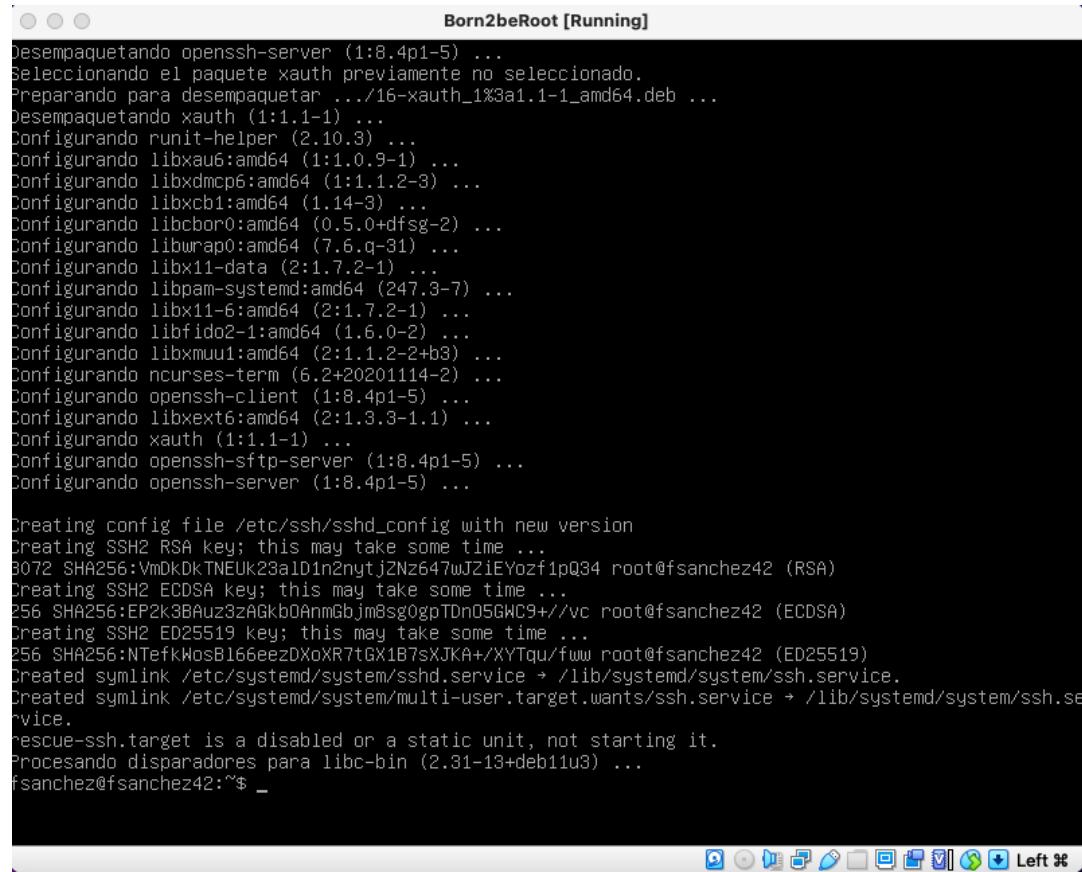
GNU nano 5.4                               /etc/hosts *
127.0.0.1      localhost
127.0.1.1      nuevo_hostname_
# The following lines are desirable for IPv6 capable hosts
::1      localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

The bottom of the window shows the nano editor's status bar with various keyboard shortcuts.

▼ Instalar y configurar el servicio SSH

1. Recopilamos los paquetes oportunos e instalamos SSH
 - a. `sudo apt-get update`

b. `sudo apt install openssh-server`



```
Born2beRoot [Running]
Desempaquetando openssh-server (1:8.4p1-5) ...
Seleccionando el paquete xauth previamente no seleccionado.
Preparando para desempaquetar .../16-xauth_1%3a1.1-1_amd64.deb ...
Desempaquetando xauth (1:1.1-1) ...
Configurando runit-helper (2.10.3) ...
Configurando libxau6:amd64 (1:1.0.9-1) ...
Configurando libxdmcp6:amd64 (1:1.1.2-3) ...
Configurando libxcb1:amd64 (1.14-3) ...
Configurando libcbor0:amd64 (0.5.0+dfsg-2) ...
Configurando libwrap0:amd64 (7.6.q-31) ...
Configurando libx11-data (2:1.7.2-1) ...
Configurando libpam-systemd:amd64 (247.3-7) ...
Configurando libx11-6:amd64 (2:1.7.2-1) ...
Configurando libfido2-1:amd64 (1.6.0-2) ...
Configurando libxmuu1:amd64 (2:1.1.2-2+b3) ...
Configurando ncurses-term (6.2+20201114-2) ...
Configurando openssh-client (1:8.4p1-5) ...
Configurando libxext6:amd64 (2:1.3.3-1.1) ...
Configurando xauth (1:1.1-1) ...
Configurando openssh-sftp-server (1:8.4p1-5) ...
Configurando openssh-server (1:8.4p1-5) ...

Creating config file /etc/ssh/sshd_config with new version
Creating SSH2 RSA key; this may take some time ...
3072 SHA256:VmDkDKTNEUK23a1Din2nytjZNz647wJ2iEYozf1pQ34 root@fsanchez42 (RSA)
Creating SSH2 ECDSA key; this may take some time ...
256 SHA256:EP2k3BAuz3zAGkb0AnmGbjm8sg0gpTDn05GWC9+/vc root@fsanchez42 (ECDSA)
Creating SSH2 ED25519 key; this may take some time ...
256 SHA256:NTefkWosB166eezDXoXR7tGX1B7sxJKA+/XYTqu/fuu root@fsanchez42 (ED25519)
Created symlink /etc/systemd/system/sshd.service → /lib/systemd/system/ssh.service.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /lib/systemd/system/ssh.service.
rescue-ssh.target is a disabled or a static unit, not starting it.
Procesando disparadores para libc-bin (2.31-13+deb11u3) ...
fsanchez@fsanchez42:~$ _
```

2. Comprobamos el estado del servidor SSH

a. `sudo systemctl status ssh`

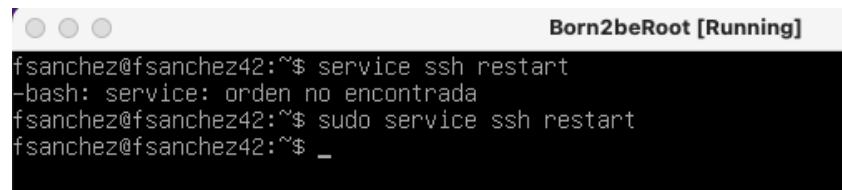


```
Born2beRoot [Running]
fsanchez@fsanchez42:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2022-05-18 17:41:09 CEST; 2min 56s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Process: 2145 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 2146 (sshd)
      Tasks: 1 (limit: 1128)
     Memory: 1.0M
        CPU: 13ms
       CGroup: /system.slice/ssh.service
                   └─2146 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

may 18 17:41:09 fsanchez42 systemd[1]: Starting OpenBSD Secure Shell server...
may 18 17:41:09 fsanchez42 sshd[2146]: Server listening on 0.0.0.0 port 22.
may 18 17:41:09 fsanchez42 sshd[2146]: Server listening on :: port 22.
may 18 17:41:09 fsanchez42 systemd[1]: Started OpenBSD Secure Shell server.
fsanchez@fsanchez42:~$
```

3. Reiniciamos el servicio SSH

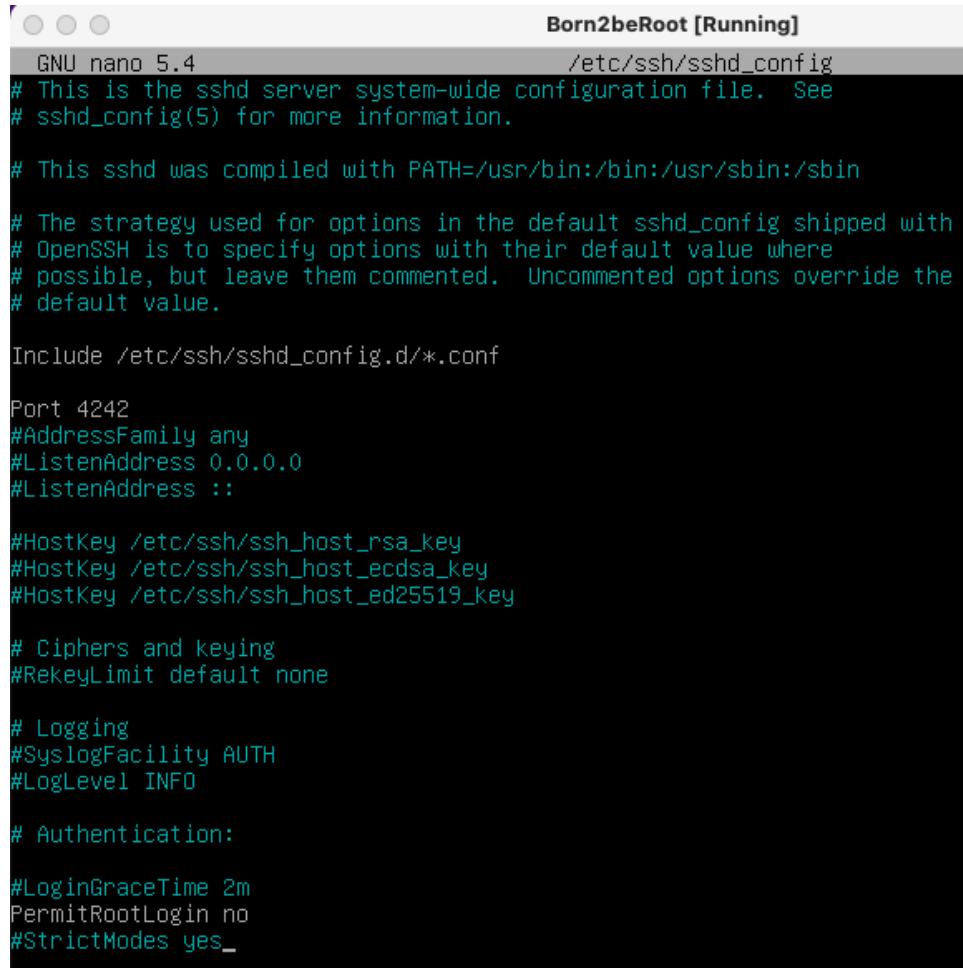
a. `sudo service ssh restart`



```
Born2beRoot [Running]
fsanchez@fsanchez42:~$ service ssh restart
-bash: service: orden no encontrada
fsanchez@fsanchez42:~$ sudo service ssh restart
fsanchez@fsanchez42:~$ -
```

4. Editamos la configuración para cambiar el puerto de conexión por defecto

- a. `sudo nano /etc/ssh/sshd_config`
- b. Cambiar la línea `#Port 22` por **Port 4242**
- c. Y bloqueamos el acceso del root con la línea **PermitRootLogin no**



```
GNU nano 5.4 /etc/ssh/sshd_config
# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 4242
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes_
```

5. Comprobamos que el cambio de puerto ha funcionado

- a. `sudo service ssh restart`
- b. `sudo grep Port /etc/ssh/sshd_config`

```
Born2beRoot [Running]
fsanchez@fsanchez42:~$ sudo service ssh restart
fsanchez@fsanchez42:~$ sudo grep Port /etc/ssh/sshd_config
Port 4242
#GatewayPorts no
fsanchez@fsanchez42:~$ _
```

▼ Instalar y configurar el servicio UFW

1. Recopilamos los paquetes oportunos e instalamos UFW

- a. `sudo apt-get update`
- b. `sudo apt-get install ufw`

```
Born2beRoot [Running]
Configurando libnetfilter-conntrack3:amd64 (1.0.8-3) ...
Configurando python3.9 (3.9.2-1) ...
Configurando iptables (1.8.7-1) ...
update-alternatives: utilizando /usr/sbin/iptables-legacy para proveer /usr/sbin/iptables (iptables) en modo automático
update-alternatives: utilizando /usr/sbin/ip6tables-legacy para proveer /usr/sbin/ip6tables (ip6tables) en modo automático
update-alternatives: utilizando /usr/sbin/iptables-nft para proveer /usr/sbin/iptables (iptables) en modo automático
update-alternatives: utilizando /usr/sbin/ip6tables-nft para proveer /usr/sbin/ip6tables (ip6tables) en modo automático
update-alternatives: utilizando /usr/sbin/arptables-nft para proveer /usr/sbin/arptables (arptables) en modo automático
update-alternatives: utilizando /usr/sbin/ebttables-nft para proveer /usr/sbin/ebttables (ebtables) en modo automático
Configurando python3 (3.9.2-3) ...
running python rtupdate hooks for python3.9...
running python post-rtupdate hooks for python3.9...
Configurando ufw (0.36-7.1) ...
Creating config file /etc/ufw/before.rules with new version
Creating config file /etc/ufw/before6.rules with new version
Creating config file /etc/ufw/after.rules with new version
Creating config file /etc/ufw/after6.rules with new version
Created symlink /etc/systemd/system/multi-user.target.wants/ufw.service → /lib/systemd/system/ufw.service.
Procesando disparadores para rsyslog (8.2102.0-2) ...
Procesando disparadores para libc-bin (2.31-18+deb11u8) ...
Procesando disparadores para ca-certificates (20210119) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
fsanchez@fsanchez42:~$ _
```

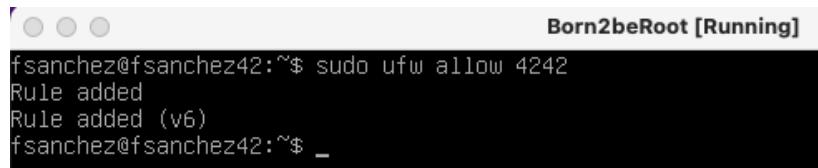
2. Habilitamos el servicio UFW para que se inicie con el sistema

- a. `sudo ufw enable`

```
Born2beRoot [Running]
fsanchez@fsanchez42:~$ sudo ufw enable
Firewall is active and enabled on system startup
fsanchez@fsanchez42:~$ _
```

3. Añadimos la regla para el puerto 4242

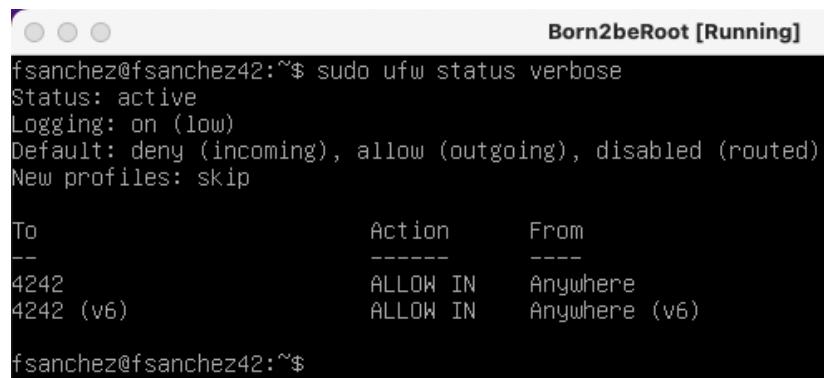
a. `sudo ufw allow 4242`



```
Born2beRoot [Running]
fsanchez@fsanchez42:~$ sudo ufw allow 4242
Rule added
Rule added (v6)
fsanchez@fsanchez42:~$ _
```

4. Comprobamos el estado del servicio UFW

a. `sudo ufw status verbose`



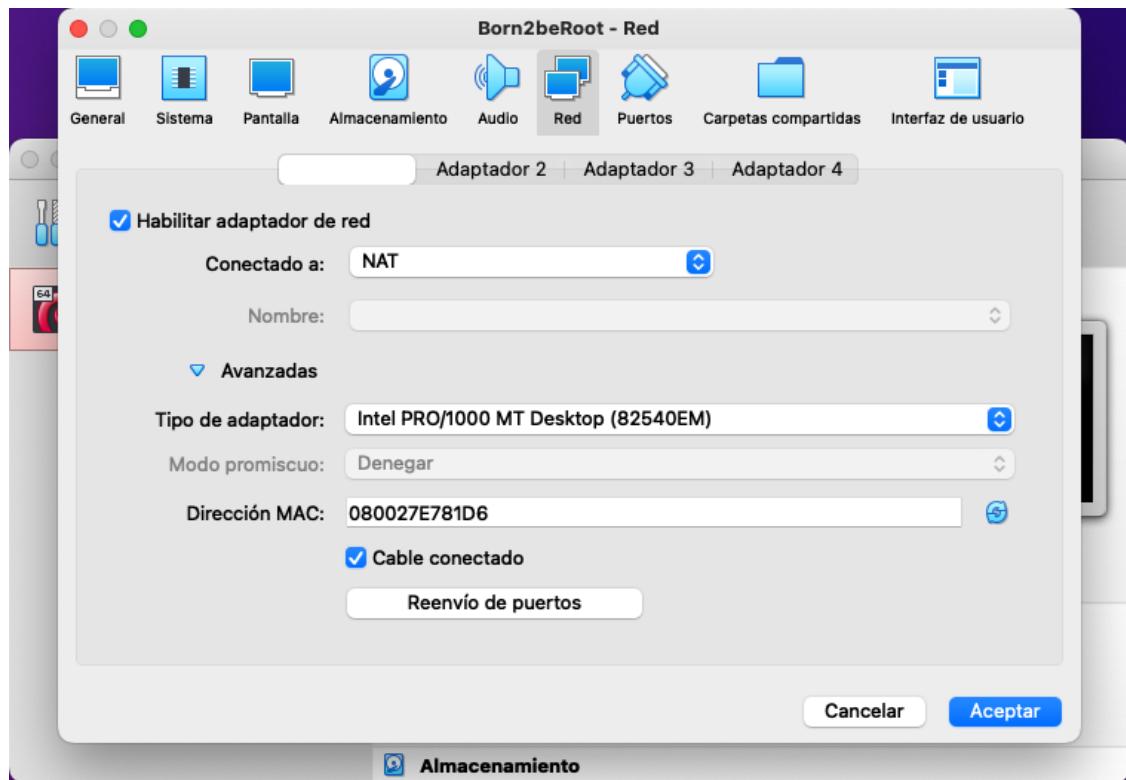
```
Born2beRoot [Running]
fsanchez@fsanchez42:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To                         Action      From
--                         ----      ---
4242                       ALLOW IN   Anywhere
4242 (v6)                   ALLOW IN   Anywhere (v6)

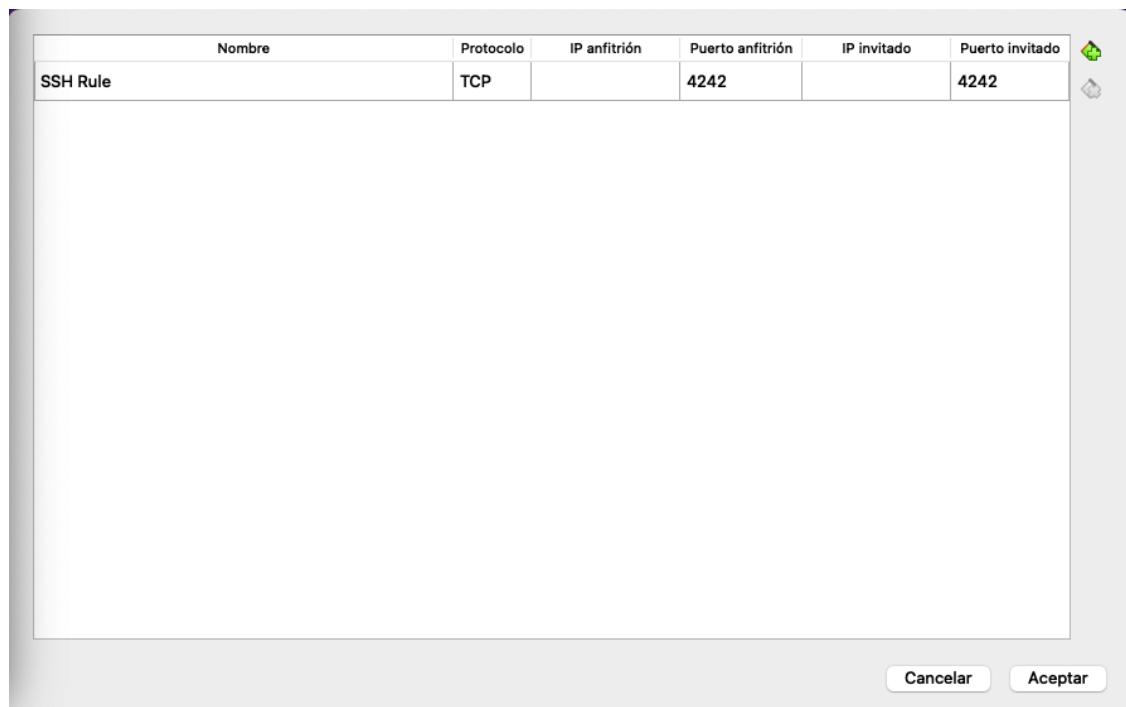
fsanchez@fsanchez42:~$
```

▼ Comprobar que los servicios SSH y UFW funcionan correctamente

1. Apagamos la máquina virtual
2. En el administrador de VirtualBox, seleccionamos Configuración y en la pestaña Red, seleccionamos el Adaptador 1, desplegamos Avanzadas y pulsamos sobre el botón Reenvío de puertos



3. En esa pantalla vamos a definir una regla (pulsando sobre el icono del +) para redirigir las peticiones entrantes por el puerto externo 4242 (el abierto por el servicio UFW) al puerto interno 4242 (que es sobre el que opera el servicio SSH)



4. Reiniciamos nuestra máquina virtual y desde un iTerm2 de nuestro equipo anfitrión (el que tiene instalado VirtualBox), ejecutamos los siguientes comandos:
- `ssh your_user@127.0.0.1 -p 4242`
 - Introducimos nuestro password en la máquina virtual y una vez autenticados, ejecutamos algún comando para comprobar que estamos viendo la consola remota segura de nuestro servidor
 - Escribimos exit para abandonar la sesión y esta vez intentamos conectarnos con el usuario root así:
 - `ssh root@127.0.0.1 -p 4242`
 - Introducimos el password del root de la máquina virtual. Si lo hemos configurado bien, nos devolverá un mensaje en consola de permiso denegado.

```

fran@MacBook:~
> ssh fsanchez@127.0.01 -p 4242
fsanchez@127.0.0.1's password:
Linux fsanchez42 5.10.0-14-amd64 #1 SMP Debian 5.10.113-1 (2022-04-29) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed May 18 19:01:39 2022 from 10.0.2.2
fsanchez@fsanchez42:~$ cd ..
fsanchez@fsanchez42:/home$ ls -l
total 20
drwxr-xr-x 3 fsanchez fsanchez 4096 may 18 18:45 fsanchez
drwx----- 2 root      root     16384 may 17 00:21 lost+found
fsanchez@fsanchez42:/home$ exit
cerrar sesión
Connection to 127.0.0.1 closed.
> ssh root@127.0.01 -p 4242
root@127.0.0.1's password:
Permission denied, please try again.
root@127.0.0.1's password:
Connection closed by 127.0.0.1 port 4242

```

▼ Configurar una política de contraseñas fuerte

- Primero actualizamos la apt-get e instalamos el servicio libpam-pwquality
 - `sudo apt-get update`

b. `sudo apt-get install libpam-pwquality`

```
Born2beRoot [Running]
Des:8 http://deb.debian.org/debian bullseye/main amd64 libpam-pwquality amd64 1.4.4-1 [13,8 kB]
Descargados 757 kB en 0s (5.193 kB/s)
Seleccionando el paquete libmagic-mgc previamente no seleccionado.
(Leyendo la base de datos ... 28665 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../0-libmagic-mgc_1%3a5.39-3_amd64.deb ...
Desempaquetando libmagic-mgc (1:5.39-3) ...
Seleccionando el paquete libmagic1:amd64 previamente no seleccionado.
Preparando para desempaquetar .../1-libmagic1_1%3a5.39-3_amd64.deb ...
Desempaquetando libmagic1:amd64 (1:5.39-3) ...
Seleccionando el paquete file previamente no seleccionado.
Preparando para desempaquetar .../2-file_1%3a5.39-3_amd64.deb ...
Desempaquetando file (1:5.39-3) ...
Seleccionando el paquete libcrack2:amd64 previamente no seleccionado.
Preparando para desempaquetar .../3-libcrack2_2.9.6-3.4_amd64.deb ...
Desempaquetando libcrack2:amd64 (2.9.6-3.4) ...
Seleccionando el paquete cracklib-runtime previamente no seleccionado.
Preparando para desempaquetar .../4-cracklib-runtime_2.9.6-3.4_amd64.deb ...
Desempaquetando cracklib-runtime (2.9.6-3.4) ...
Seleccionando el paquete libpwquality-common previamente no seleccionado.
Preparando para desempaquetar .../5-libpwquality-common_1.4.4-1_all.deb ...
Desempaquetando libpwquality-common (1.4.4-1) ...
Seleccionando el paquete libpwquality1:amd64 previamente no seleccionado.
Preparando para desempaquetar .../6-libpwquality1_1.4.4-1_amd64.deb ...
Desempaquetando libpwquality1:amd64 (1.4.4-1) ...
Seleccionando el paquete libpam-pwquality:amd64 previamente no seleccionado.
Preparando para desempaquetar .../7-libpam-pwquality_1.4.4-1_amd64.deb ...
Desempaquetando libpam-pwquality:amd64 (1.4.4-1) ...
Configurando libpwquality-common (1.4.4-1) ...
Configurando libmagic-mgc (1:5.39-3) ...
Configurando libmagic1:amd64 (1:5.39-3) ...
Configurando file (1:5.39-3) ...
Configurando libcrack2:amd64 (2.9.6-3.4) ...
Configurando cracklib-runtime (2.9.6-3.4) ...
Configurando libpwquality1:amd64 (1.4.4-1) ...
Configurando libpam-pwquality:amd64 (1.4.4-1) ...
Procesando disparadores para libc-bin (2.31-13+deb11u3) ...
fsanchez@fsanchez42:~$ _
```

2. Vamos a editar el archivo /etc/pam.d/common-password

a. `sudo nano /etc/pam.d/common-password`

b. editamos las siguientes líneas:

i. `password requisite pam_pwquality.o retry=3` y le añadimos al final:

1. `minlen=10 ucredit=-1 dcredit=-1 maxrepeat=3 reject_username difok=7 enforce_for_root`

```

GNU nano 5.4                               Born2beRoot [Running]
#                                         /etc/pam.d/common-password

#
# This file is included from other service-specific PAM config files,
# and should contain a list of modules that define the services to be
# used to change user passwords. The default is pam_unix.

# Explanation of pam_unix options:
# The "yescrypt" option enables
# hashed passwords using the yescrypt algorithm, introduced in Debian
# #11. Without this option, the default is Unix crypt. Prior releases
# used the option "sha512"; if a shadow password hash will be shared
# between Debian 11 and older releases replace "yescrypt" with "sha512"
# for compatibility. The "obscure" option replaces the old
# #OBSCURE_CHECKS_ENAB' option in login.defs. See the pam_unix manpage
# for other options.

# As of pam 1.0.1-6, this file is managed by pam-auth-update by default.
# To take advantage of this, it is recommended that you configure any
# local modules either before or after the default block, and use
# pam-auth-update to manage selection of other modules. See
# pam-auth-update(8) for details.

# here are the per-package modules (the "Primary" block)
password      requisite          pam_puquality.so retry=3 minlen=10 ucredir=-1 dcred>
password      [success=1 default=ignore]    pam_unix.so obscure use_authtok try_first_pass yes>
# here's the fallback if no module succeeds
password      requisite          pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
password      required           pam_permit.so
# and here are more per-package modules (the "Additional" block)
# end of pam-auth-update config

```

3. Ahora editaremos el archivo /etc/login.defs

- `sudo nano /etc/login.defs`
- Cambiamos las siguientes líneas
- PASS_MAX_DAYS 30**
PASS_MIN_DAYS 2
PASS_WARN_AGE 7

```

GNU nano 5.4                               /etc/login.defs *

#-
ERASECHAR      0177
KILLCHAR       025
UMASK          022

#
# Password aging controls:
#
#      PASS_MAX_DAYS   Maximum number of days a password may be used.
#      PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#      PASS_WARN_AGE    Number of days warning given before a password expires.
#
PASS_MAX_DAYS  30
PASS_MIN_DAYS  2
PASS_WARN_AGE  7

#
# Min/max values for automatic uid selection in useradd
#
UID_MIN         1000
UID_MAX         60000
# System accounts
#SYS_UID_MIN    100
#SYS_UID_MAX    999

#
# Min/max values for automatic gid selection in groupadd
#
GID_MIN         1000
GID_MAX         60000
# System accounts
#SYS_GID_MIN    100
#SYS_GID_MAX    999

^G Ayuda      ^D Guardar     ^W Buscar     ^K Cortar     ^T Ejecutar     ^C Ubicación M-U Deshacer
^X Salir      ^R Leer fich.  ^Y Reemplazar ^U Pegar      ^J Justificar ^I Ir a línea M-E Rehacer
Left ⌘

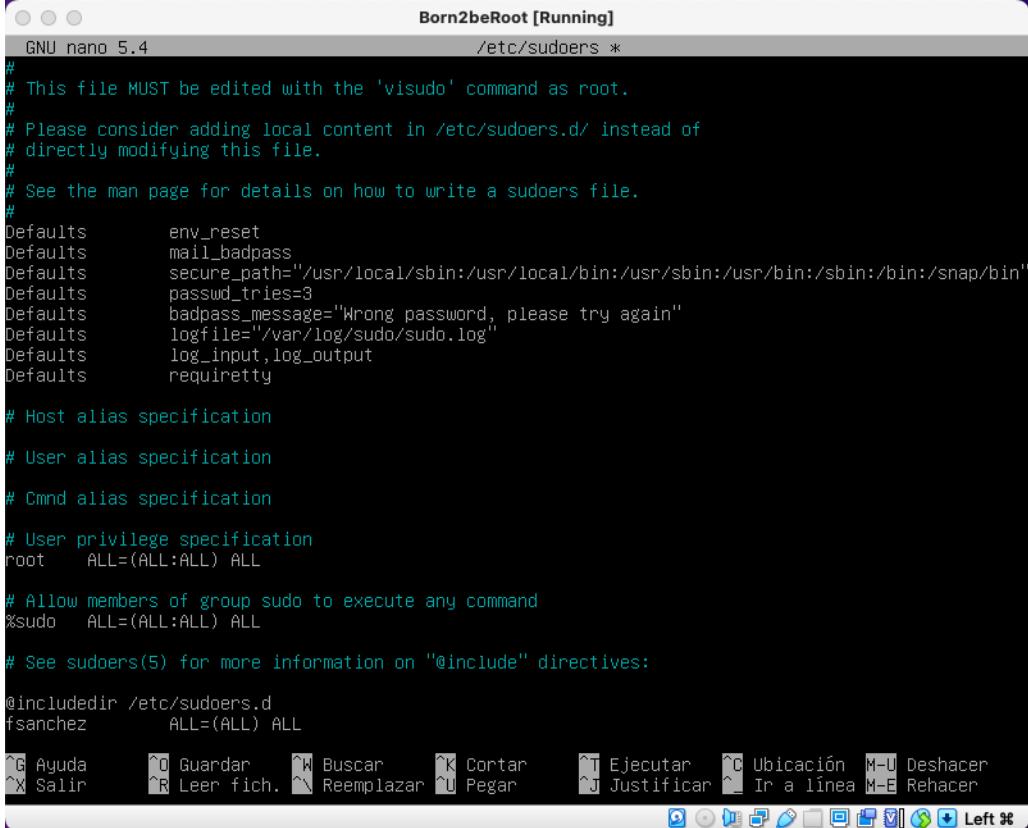
```

▼ Implementar una política de contraseñas fuerte para el grupo sudo

1. Vamos a editar el archivo /etc/sudoers
 - a. `sudo nano /etc/sudoers`
 - b. Vamos a añadir las siguientes líneas
 - i. **Defaults**
`secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"`
 (añadimos /snap/bin a los directorios permitidos para sudo)
 - ii. **Defaults passwd_tries=3** (limitamos a 3 los intentos de password erronea)
 - iii. **Defaults badpass_message="Ups! the password is wrong, please try again"**
 (Definimos un mensaje para cuando se introduce una contraseña erronea al hacer sudo)
 - iv. **Defaults logfile="/var/log/sudo/sudo.log"** (definimos un archivo de log para escribir los comandos realizados en modo sudo)

⚠ Nos aseguramos que la carpeta /var/log/sudo existe en sistema, si no, la creamos manualmente

- v. **Defaults log_input,log_output** (indicamos que el log de sudo debe ser de todos los mensajes de entrada y salida estándar)
- vi. **Defaults requiretty** (Definimos como obligatorio el modo tty)



```

Born2beRoot [Running]
GNU nano 5.4
/etc/sudoers *

#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults    env_reset
Defaults    mail_badpass
Defaults    secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"
Defaults    passwd_tries=3
Defaults    badpass_message="Wrong password, please try again"
Defaults    logfile="/var/log/sudo/sudo.log"
Defaults    log_input,log_output
Defaults    requiretty

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

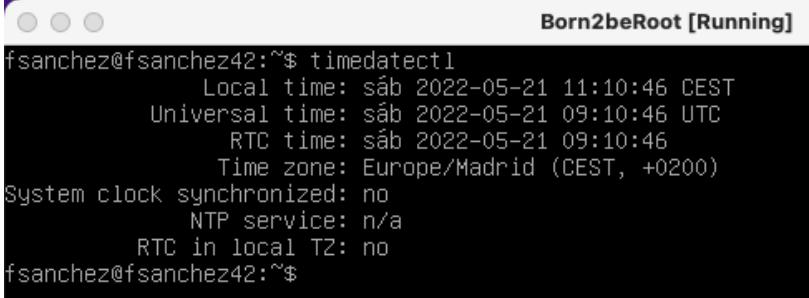
# See sudoers(5) for more information on "@include" directives:

@includedir /etc/sudoers.d
fsanchez    ALL=(ALL) ALL

^G Ayuda      ^O Guardar     ^W Buscar     ^K Cortar     ^T Ejecutar     ^C Ubicación M-U Deshacer
^X Salir      ^R Leer fich.  ^E Reemplazar ^U Pegar      ^J Justificar  ^I Ir a línea M-E Rehacer
Left %
```

▼ Creación del script de ejecución automática

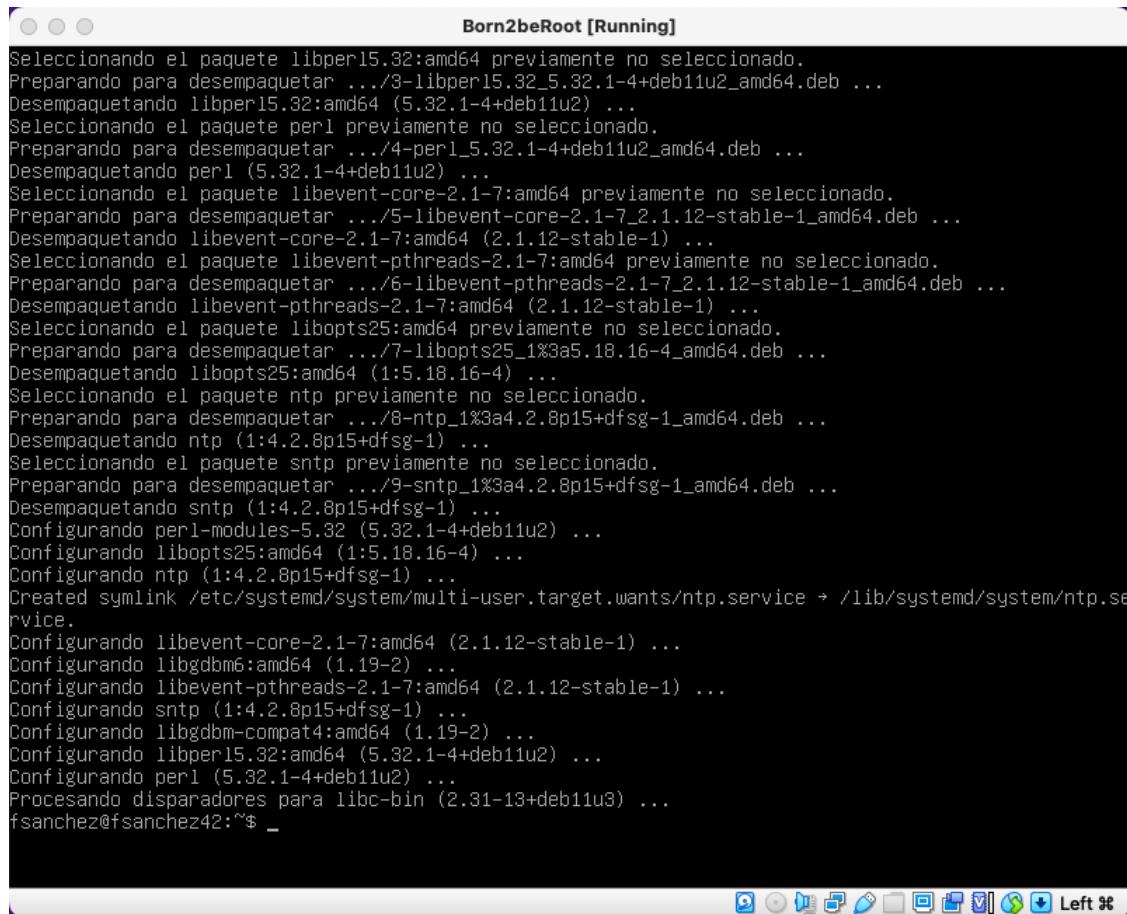
1. Primero vamos a asegurarnos que la fecha y hora del sistema están bien configurados, de lo contrario, el servicio que ejecutará el script no lo hará en el momento adecuado.
2. Comprobamos que los servicios de tiempo del sistema están instalados y bien configurados con el comando `timedatectl`



```

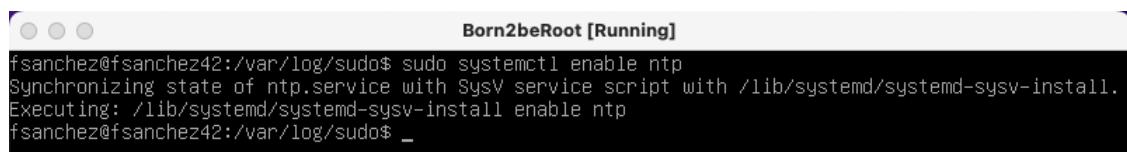
Born2beRoot [Running]
fsanchez@fsanchez42:~$ timedatectl
        Local time: sábado 2022-05-21 11:10:46 CEST
        Universal time: sábado 2022-05-21 09:10:46 UTC
                  RTC time: sábado 2022-05-21 09:10:46
                 Time zone: Europe/Madrid (CEST, +0200)
System clock synchronized: no
          NTP service: n/a
       RTC in local TZ: no
fsanchez@fsanchez42:~$
```

3. Si el servicio NTP no está activo, vamos a hacer que lo esté, primero instalando el servicio con `sudo apt install ntp`



```
Born2beRoot [Running]
Seleccionando el paquete libperl5.32:amd64 previamente no seleccionado.
Preparando para desempaquetar .../3-libperl5.32_5.32.1-4+deb11u2_amd64.deb ...
Desempaquetando libperl5.32:amd64 (5.32.1-4+deb11u2) ...
Seleccionando el paquete perl previamente no seleccionado.
Preparando para desempaquetar .../4-perl_5.32.1-4+deb11u2_amd64.deb ...
Desempaquetando perl (5.32.1-4+deb11u2) ...
Seleccionando el paquete libevent-core-2.1-7:amd64 previamente no seleccionado.
Preparando para desempaquetar .../5-libevent-core-2.1-7_2.1.12-stable-1_amd64.deb ...
Desempaquetando libevent-core-2.1-7:amd64 (2.1.12-stable-1) ...
Seleccionando el paquete libevent-pthreads-2.1-7:amd64 previamente no seleccionado.
Preparando para desempaquetar .../6-libevent-pthreads-2.1-7_2.1.12-stable-1_amd64.deb ...
Desempaquetando libevent-pthreads-2.1-7:amd64 (2.1.12-stable-1) ...
Seleccionando el paquete libopts25:amd64 previamente no seleccionado.
Preparando para desempaquetar .../7-libopts25_1%3a5.18.16-4_amd64.deb ...
Desempaquetando libopts25:amd64 (1:5.18.16-4) ...
Seleccionando el paquete ntp previamente no seleccionado.
Preparando para desempaquetar .../8-ntp_1%3a4.2.8p15+dfsg-1_amd64.deb ...
Desempaquetando ntp (1:4.2.8p15+dfsg-1) ...
Seleccionando el paquete sntp previamente no seleccionado.
Preparando para desempaquetar .../9-sntp_1%3a4.2.8p15+dfsg-1_amd64.deb ...
Desempaquetando sntp (1:4.2.8p15+dfsg-1) ...
Configurando perl-modules-5.32 (5.32.1-4+deb11u2) ...
Configurando libopts25:amd64 (1:5.18.16-4) ...
Configurando ntp (1:4.2.8p15+dfsg-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ntp.service → /lib/systemd/system/ntp.service.
Configurando libevent-core-2.1-7:amd64 (2.1.12-stable-1) ...
Configurando libgdbm6:amd64 (1.19-2) ...
Configurando libevent-pthreads-2.1-7:amd64 (2.1.12-stable-1) ...
Configurando sntp (1:4.2.8p15+dfsg-1) ...
Configurando libgdbm-compat4:amd64 (1.19-2) ...
Configurando libperl5.32:amd64 (5.32.1-4+deb11u2) ...
Configurando perl (5.32.1-4+deb11u2) ...
Procesando disparadores para libc-bin (2.31-13+deb11u3) ...
fsanchez@fsanchez42:~$ _
```

4. A continuación, nos aseguramos que el servicio NTP se inicia con el sistema ejecutando `sudo systemctl enable ntp`



```
Born2beRoot [Running]
fsanchez@fsanchez42:/var/log/sudo$ sudo systemctl enable ntp
Synchronizing state of ntp.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable ntp
fsanchez@fsanchez42:/var/log/sudo$ _
```

5. Como nos dice el mensaje, tendremos que habilitar **NTP** con el servicio **SysV** ejecutando el comando `sudo /lib/systemd/systemd-sysv-install enable ntp`
6. Y Comprobamos que tenemos el servicio cargado y activo con `sudo systemctl status ntp.service`, aunque es posible que las solicitudes al servidor de tiempo den resultados incorrectos o inesperados.

```

Born2beRoot [Running]
fsanchez@fsanchez42:/var/log/sudo$ sudo systemctl status ntp.service
● ntp.service - Network Time Service
    Loaded: loaded (/lib/systemd/system/ntp.service; enabled; vendor preset: enabled)
      Active: active (running) since Sat 2022-05-21 11:17:38 CEST; 15min ago
        Docs: man:ntpd(8)
       Main PID: 593 (ntpd)
          Tasks: 2 (limit: 1128)
         Memory: 2.1M
            CPU: 126ms
           CGroup: /system.slice/ntp.service
               └─593 /usr/sbin/ntpd -p /var/run/ntpd.pid -g -u 106:113

may 21 11:18:55 fsanchez42 ntpd[593]: receive: Unexpected origin timestamp 0xe6332cf.34c6a0c4 does>
may 21 11:18:55 fsanchez42 ntpd[593]: receive: Unexpected origin timestamp 0xe6332cf.34ca0b9e does>
may 21 11:18:55 fsanchez42 ntpd[593]: receive: Unexpected origin timestamp 0xe6332cf.34ca5305 does>
may 21 11:18:55 fsanchez42 ntpd[593]: receive: Unexpected origin timestamp 0xe6332cf.34c7b3cd does>
may 21 11:18:55 fsanchez42 ntpd[593]: receive: Unexpected origin timestamp 0xe6332cf.34c22fb4 does>
may 21 11:24:37 fsanchez42 ntpd[593]: kernel reports TIME_ERROR: 0x41: Clock Unsynchronized
may 21 11:27:56 fsanchez42 ntpd[593]: 112.213.34.20 local addr 10.0.2.15 -> <null>
may 21 11:28:00 fsanchez42 ntpd[593]: 178.215.228.24 local addr 10.0.2.15 -> <null>
may 21 11:30:10 fsanchez42 ntpd[593]: 90.165.120.190 local addr 10.0.2.15 -> <null>
may 21 11:32:49 fsanchez42 ntpd[593]: 5.56.160.3 local addr 10.0.2.15 -> <null>
fsanchez@fsanchez42:/var/log/sudo$

```

7. Por eso, vamos a configurar NTP para que sincronice con nuestro servidor de hora preferido, editando el archivo `/etc/ntp.conf` → `sudo nano /etc/ntp.conf`
8. Añadimos la línea **server hora.roa.es** (que es el servidor de tiempo oficial de España)

```

GNU nano 5.4                               /etc/ntp.conf
# /etc/ntp.conf, configuration for ntpd; see ntp.conf(5) for help

driftfile /var/lib/ntp/ntp.drift

# Leap seconds definition provided by tzdata
leapfile /usr/share/zoneinfo/leap-seconds.list

# Enable this if you want statistics to be logged.
#statsdir /var/log/ntpstats/

statistics loopstats peerstats clockstats
filegen loopstats file loopstats type day enable
filegen peerstats file peerstats type day enable
filegen clockstats file clockstats type day enable

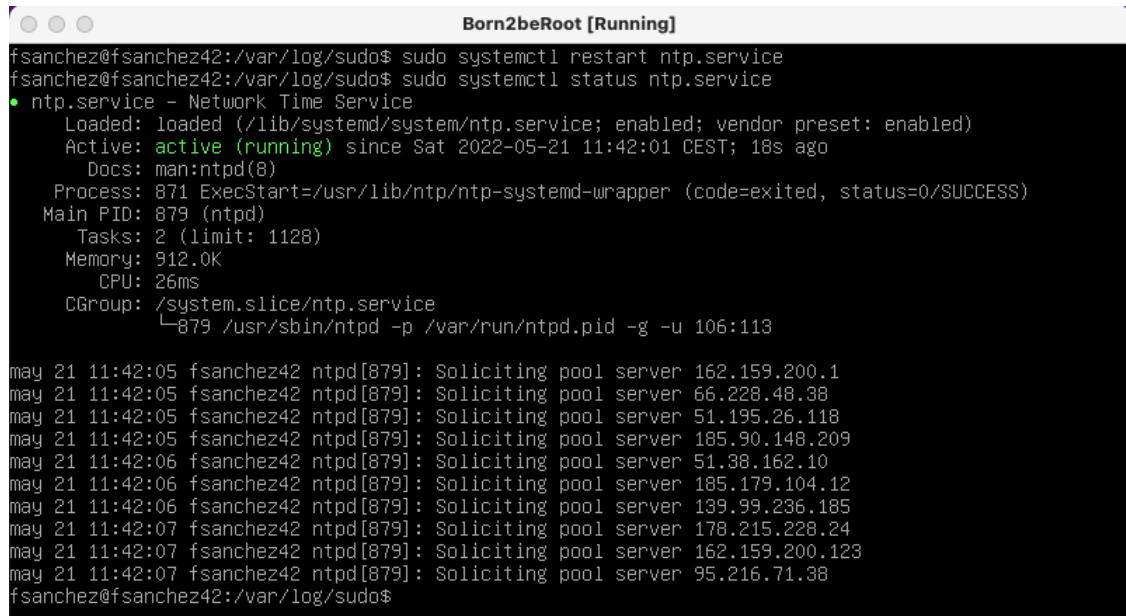
# You do need to talk to an NTP server or two (or three).
#server ntp.your-provider.example

# pool.ntp.org maps to about 1000 low-stratum NTP servers. Your server will
# pick a different set every time it starts up. Please consider joining the
# pool: <http://www.pool.ntp.org/join.html>
server hora.roa.es
pool 0.debian.pool.ntp.org iburst
pool 1.debian.pool.ntp.org iburst
pool 2.debian.pool.ntp.org iburst
pool 3.debian.pool.ntp.org iburst

# Access control configuration; see /usr/share/doc/ntp-doc/html/accept.html for
# details. The web page <http://support.ntp.org/bin/view/Support/AccessRestrictions>
# might also be helpful.
#
[ 61 líneas escritas ]
^Q Ayuda      ^O Guardar      ^W Buscar      ^K Cortar      ^T Ejecutar      ^C Ubicación      M-U Deshacer
^X Salir      ^R Leer fich.  ^Y Reemplazar  ^U Pegar       ^J Justificar   ^_ Ir a línea M-E Rehacer

```

9. Reiniciamos el servicio NTP con `sudo systemctl restart ntp.service`
10. Volvemos a ejecutar el comando `sudo systemctl status ntp.service` y comprobamos que todas las llamadas son correctas



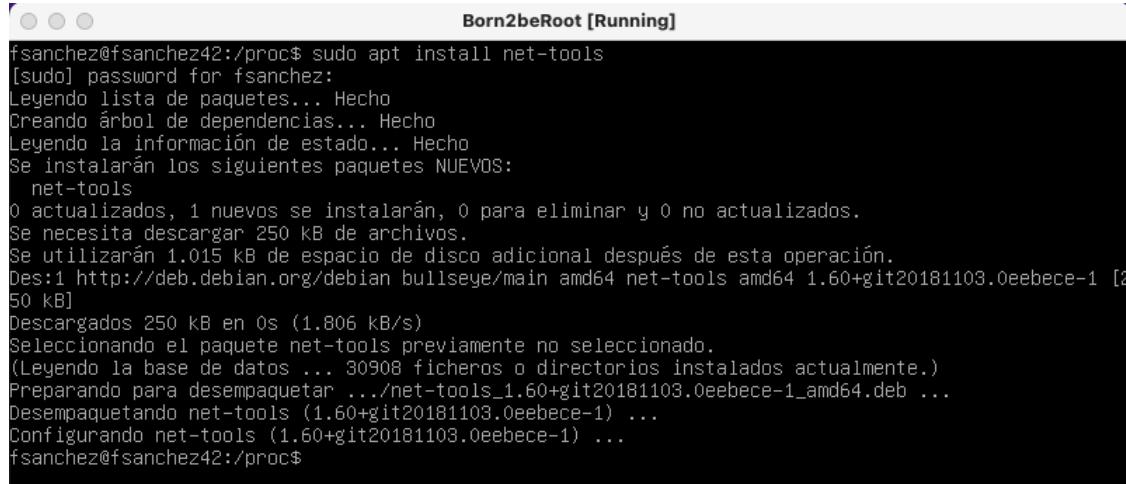
```

Born2beRoot [Running]
fsanchez@fsanchez42:/var/log/sudo$ sudo systemctl restart ntp.service
fsanchez@fsanchez42:/var/log/sudo$ sudo systemctl status ntp.service
● ntp.service - Network Time Service
    Loaded: loaded (/lib/systemd/system/ntp.service; enabled; vendor preset: enabled)
    Active: active (running) since Sat 2022-05-21 11:42:01 CEST; 18s ago
      Docs: man:ntpd(8)
   Process: 871 ExecStart=/usr/lib/ntp/ntp-systemd-wrapper (code=exited, status=0/SUCCESS)
 Main PID: 879 (ntpd)
    Tasks: 2 (limit: 1128)
   Memory: 912.0K
      CPU: 26ms
     CGroup: /system.slice/ntp.service
             └─879 /usr/sbin/ntpd -p /var/run/ntpd.pid -g -u 106:113

may 21 11:42:05 fsanchez42 ntpd[879]: Soliciting pool server 162.159.200.1
may 21 11:42:05 fsanchez42 ntpd[879]: Soliciting pool server 66.228.48.38
may 21 11:42:05 fsanchez42 ntpd[879]: Soliciting pool server 51.195.26.118
may 21 11:42:05 fsanchez42 ntpd[879]: Soliciting pool server 185.90.148.209
may 21 11:42:06 fsanchez42 ntpd[879]: Soliciting pool server 51.38.162.10
may 21 11:42:06 fsanchez42 ntpd[879]: Soliciting pool server 185.179.104.12
may 21 11:42:06 fsanchez42 ntpd[879]: Soliciting pool server 139.99.236.185
may 21 11:42:07 fsanchez42 ntpd[879]: Soliciting pool server 178.215.228.24
may 21 11:42:07 fsanchez42 ntpd[879]: Soliciting pool server 162.159.200.123
may 21 11:42:07 fsanchez42 ntpd[879]: Soliciting pool server 95.216.71.38
fsanchez@fsanchez42:/var/log/sudo$

```

11. A continuación vamos a instalar las herramientas de red (net tools), que nos van a hacer falta para obtener información de las conexiones activas. Lo haremos ejecutando el comando `sudo apt install net-tools`



```

Born2beRoot [Running]
fsanchez@fsanchez42:/proc$ sudo apt install net-tools
[sudo] password for fsanchez:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes NUEVOS:
  net-tools
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 250 kB de archivos.
Se utilizarán 1.015 kB de espacio de disco adicional después de esta operación.
Des:1 http://deb.debian.org/debian bullseye/main amd64 net-tools amd64 1.60+git20181103.0eebece-1 [250 kB]
Descargados 250 kB en 0s (1.806 kB/s)
Seleccionando el paquete net-tools previamente no seleccionado.
(Leyendo la base de datos ... 30908 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../net-tools_1.60+git20181103.0eebece-1_amd64.deb ...
Desempaquetando net-tools (1.60+git20181103.0eebece-1) ...
Configurando net-tools (1.60+git20181103.0eebece-1) ...
fsanchez@fsanchez42:/proc$ 

```

12. Creamos el archivo monitoring.sh dentro de la carpeta /usr/local/bin

```

Born2beRoot [Running]
fsanchez@fsanchez42:/usr/local/bin$ sudo touch monitorin.sh
fsanchez@fsanchez42:/usr/local/bin$ ls
monitorin.sh
fsanchez@fsanchez42:/usr/local/bin$ _

```

13. Lo editamos con nano para añadir todas las instrucciones que queremos que se muestren en pantalla, incluyendo el uso de wall, que nos permitirá dar el formato final solicitado: `sudo nano`

`monitorin.sh`

La línea que se corta es: `phdisk = $(df -Bm | grep '^/dev/' | grep -v '/boot$' | mawk '{ut += $3} {ft += $2} END {printf("%d"), ut/ft*100}'")`

```

GNU nano 5.4                               /usr/local/bin/monitoring.sh
#!/bin/bash

arch=$(uname -a)
phcpu=$(grep "physical id" /proc/cpuinfo | sort | uniq | wc -l)
vrcpu=$(grep "processor" /proc/cpuinfo | wc -l)
freeram=$(free -m | mawk '$1 == "Mem:" {print $2}')
usedram=$(free -m | mawk '$1 == "Mem:" {print $3}')
phram=$(free | mawk '$1 == "Mem:" {printf("%.2f"), $3/$2*100}')
freedisk=$(df -Bg | grep '^/dev/' | grep -v '/boot' | mawk '{ft += $2} END {print ft}')
usedisk=$(df -Bm | grep '^/dev/' | grep -v '/boot$' | mawk '{ut += $3} END {print ut}')
phdisk=$(df -Bm | grep '^/dev/' | grep -v '/boot$' | mawk '{ut += $3} {ft += $2} END {printf("%d"), ut/ft*100}')
usedcpu=$(top -bn1 | grep '^%Cpu' | cut -c 9- | xargs | mawk '{printf("%.1f%%"), $1 + $3}')
lastboot=$(who -b | mawk '$1 == "arranque" {print $4 " " $5}')
lvm=$(lsblk | grep "lvm" | wc -l)
checklvm=$(if [ $lvm -eq 0 ]; then echo no; else echo yes; fi)
tcpconex=$(cat /proc/net/sockstat{,6} | mawk '$1 == "TCP:" {print $3}')
userslog=$(users | wc -w)
ipv4=$(hostname -I)
mac=$(ip link show | mawk '$1 == "link/ether" {print $2}')
sudocommands=$(journalctl _COMM=sudo | grep COMMAND | wc -l)

wall " #Architecture: $arch
#CPU physical : $phcpu
#vCPU : $vrcpu
#Memory Usage: $usedram/${freeram}MB ($phram%)
#Disk Usage: $usedisk/${freedisk}Gb ($phdisk%)
#CPU load: $usedcpu
#Last boot: $lastboot
#LVM use: $checklvm
#Conexions TCP : $tcpconex ESTABLISHED
#User log: $userslog
#Network: IP $ipv4 ($mac)
#Sudo : $sudocommands cmd"

```

^G Ayuda ^O Guardar ^W Buscar ^K Cortar ^T Ejecutar ^D Ubicación M-U Deshacer
 ^X Salir ^R Leer fich. ^N Reemplazar ^U Pegar ^J Justificar ^_ Ir a línea M-E Rehacer
 ^L ^P ^C ^S ^I ^A ^F ^G Left %

14. Ahora vamos a modificar la configuración del crontab del root para añadir la tarea de enseñar por pantalla el archivo de monitorización que hemos creado. Usamos `sudo crontab -u root -e`

15. Añadimos la línea `/10 * * * * sh /usr/local/bin/monitoring.sh` al archivo

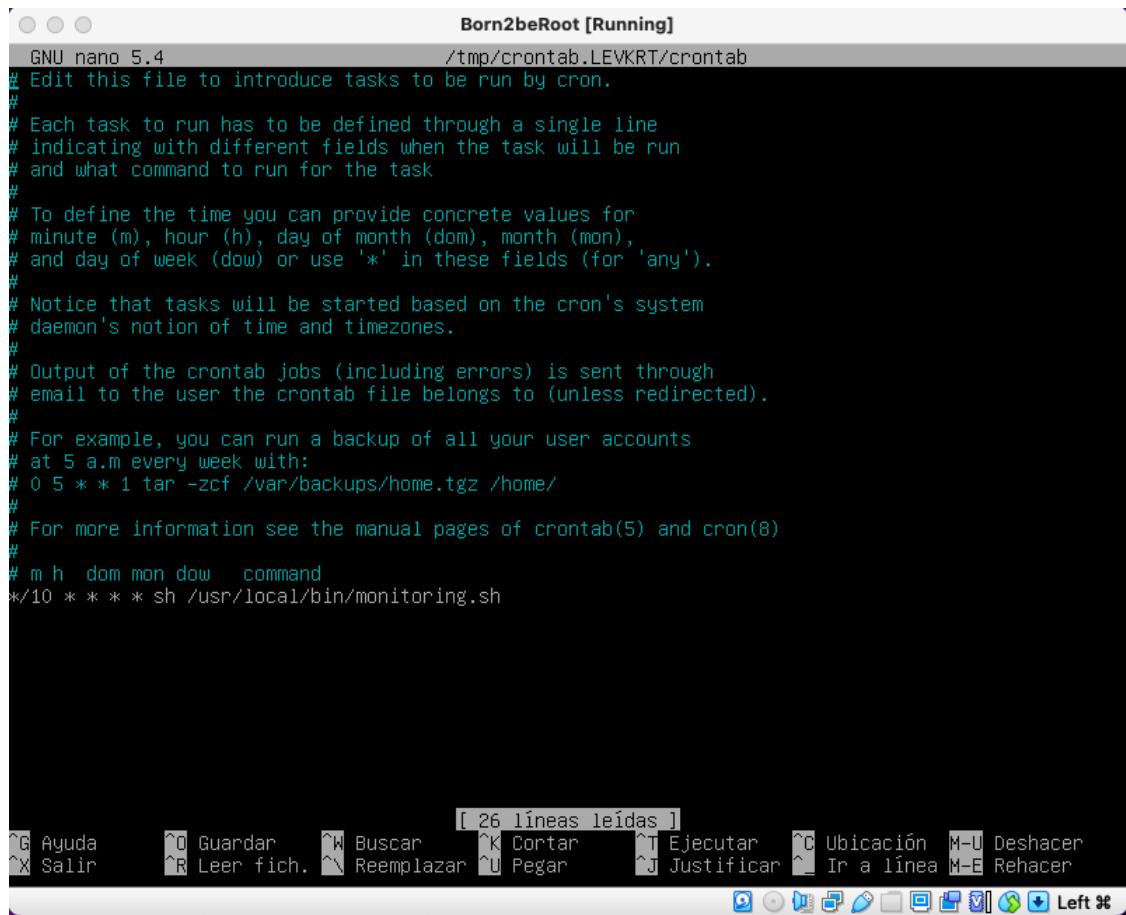
Born2beRoot [Running]

GNU nano 5.4 /tmp/crontab.LEVKRT/crontab

```
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
*/10 * * * * sh /usr/local/bin/monitoring.sh
```

[26 líneas leídas]

^G Ayuda ^O Guardar ^W Buscar ^K Cortar ^T Ejecutar ^C Ubicación M-U Deshacer
^X Salir ^R Leer fich. ^Y Reemplazar ^U Pegar ^J Justificar ^L Ir a línea M-E Rehacer



[Volver a la Tabla de Contenidos](#)