

LUKS Encryption



# Bastionado de redes y sistemas

T3A7.- PROTECCIÓN DEL SISTEMA DE FICHERO LUKS  
PEDRO MANUEL GARCÍA ÁLVAREZ

# Índice

<b><u>1.- IDENTIFICACIÓN DEL VOLUMEN A CIFRAR .....</u></b>	<b><u>2</u></b>
<b><u>2.- INSTALACIÓN Y PREPARACIÓN.....</u></b>	<b><u>10</u></b>
<b><u>3.- CIFRADO DEL DISCO .....</u></b>	<b><u>12</u></b>
<b><u>4.- VER INFORMACIÓN SOBRE SISTEMA CIFRADO.....</u></b>	<b><u>14</u></b>
<b><u>5.- COPIA DE SEGURIDAD DE LA CABECERA .....</u></b>	<b><u>15</u></b>
<b><u>6.- AÑADIR NUEVAS CONTRASEÑAS (SLOTS) .....</u></b>	<b><u>16</u></b>
<b><u>7.- ELIMINAR CONTRASEÑA (SLOTS).....</u></b>	<b><u>17</u></b>
<b><u>8.- ABRIR EL CONTENEDOR CIFRADO .....</u></b>	<b><u>18</u></b>
<b><u>9.- FORMATEO Y MONTAJE DE LA UNIDAD.....</u></b>	<b><u>19</u></b>
<b><u>11.- REAPERTURA DEL DISCO .....</u></b>	<b><u>21</u></b>
<b><u>12.- MONTAJE AUTOMÁTICO DEL VOLUMEN EN EL ARRANQUE .....</u></b>	<b><u>21</u></b>

## 1.- Identificación del volumen a cifrar

Los comandos `lsblk`, `fdisk -l` y `blkid` son utilidades en Linux para listar información sobre los dispositivos de bloques (discos y particiones). Aquí tienes una breve explicación de cada uno:

**lsblk:** Muestra información sobre los dispositivos de bloques disponibles en el sistema, incluyendo discos duros, particiones y dispositivos de almacenamiento extraíbles como pendrives. Muestra la jerarquía de los dispositivos en un formato de árbol.

**fdisk -l:** Muestra una lista detallada de todas las particiones en todos los discos disponibles en el sistema. Proporciona información sobre el tamaño de las particiones, el tipo de sistema de archivos, etc.

**blkid:** Muestra información sobre los identificadores únicos de los dispositivos de bloques, como el UUID (identificador único universal) y el tipo de sistema de archivos. Es útil para identificar dispositivos de forma única, incluso si los nombres de dispositivo cambian.

Para identificar el dispositivo que deseas cifrar, puedes usar cualquiera de estos comandos y buscar el dispositivo que acabas de añadir. Una vez identificado, en este caso, `/dev/sdb` y su partición `/dev/sdb1`, puedes proceder a cifrarlo.

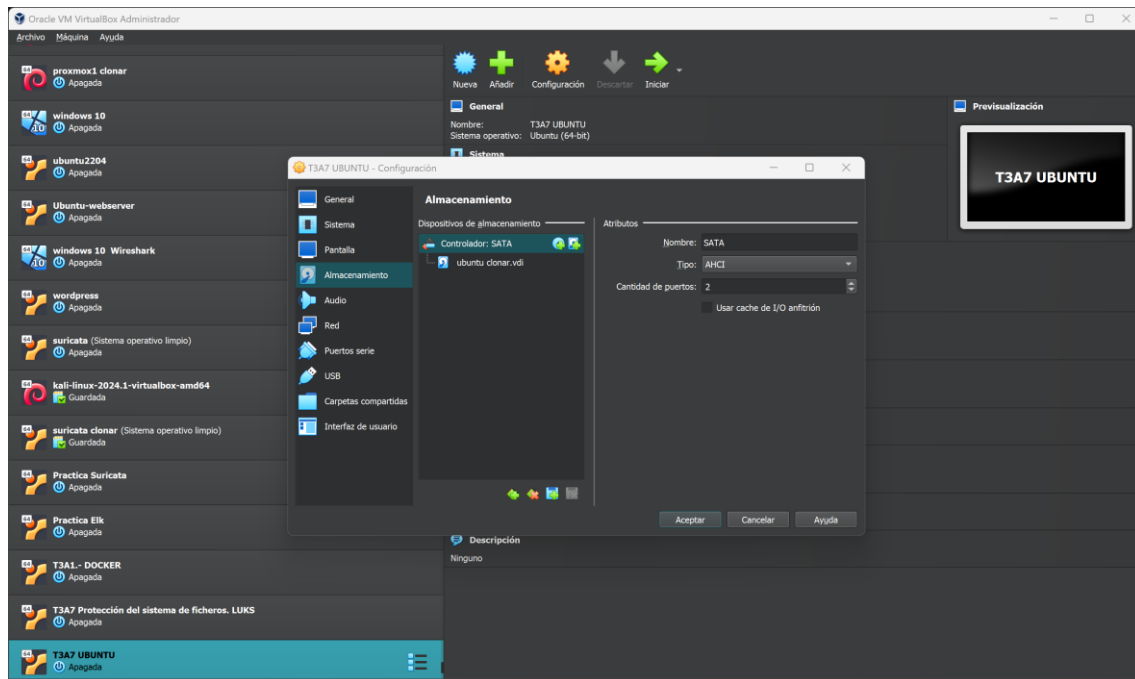
Recuerda que el cifrado de un dispositivo eliminará toda la información que contenga. Si el dispositivo es nuevo y necesita ser particionado, puedes utilizar el comando `fdisk` para crear una partición en él antes de cifrarlo. Por ejemplo:

```
fdisk /dev/sdb
```

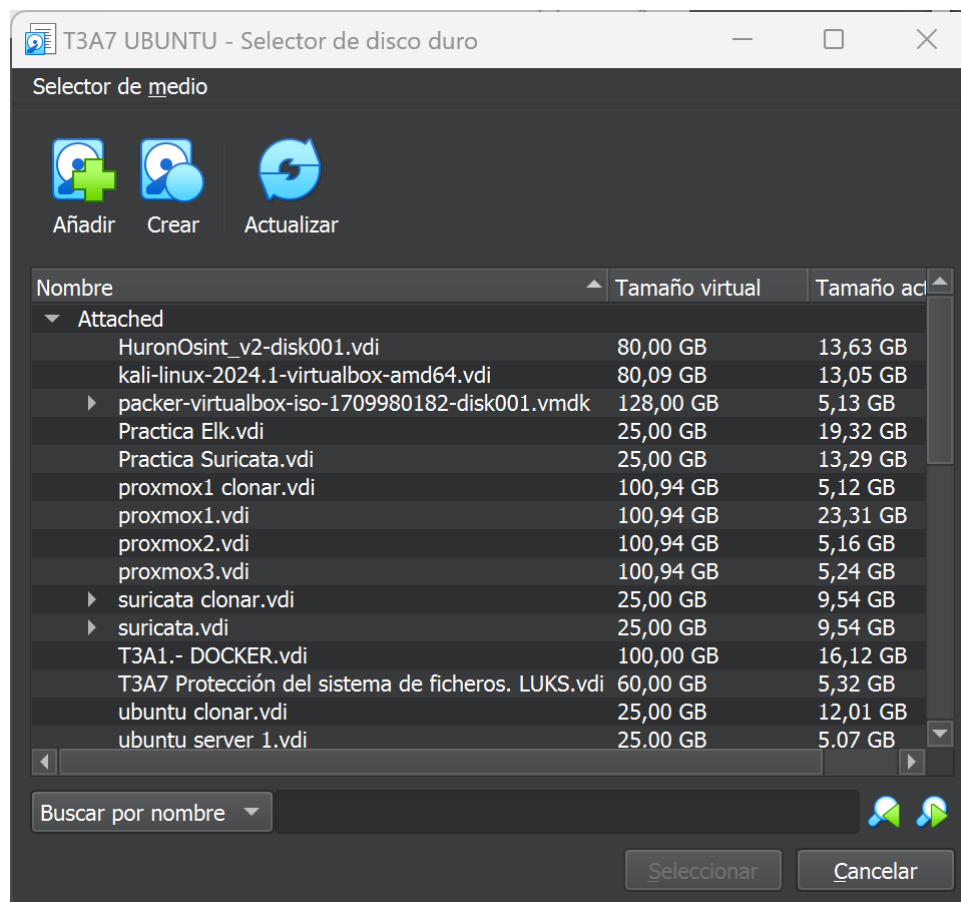
Luego, puedes seguir las instrucciones para crear una nueva partición y escribir los cambios. Una vez que tengas la partición creada, puedes proceder con el cifrado.

Es importante tener precaución al cifrar dispositivos, ya que una vez cifrados, los datos pueden volverse inaccesibles si se olvida la contraseña o la clave de

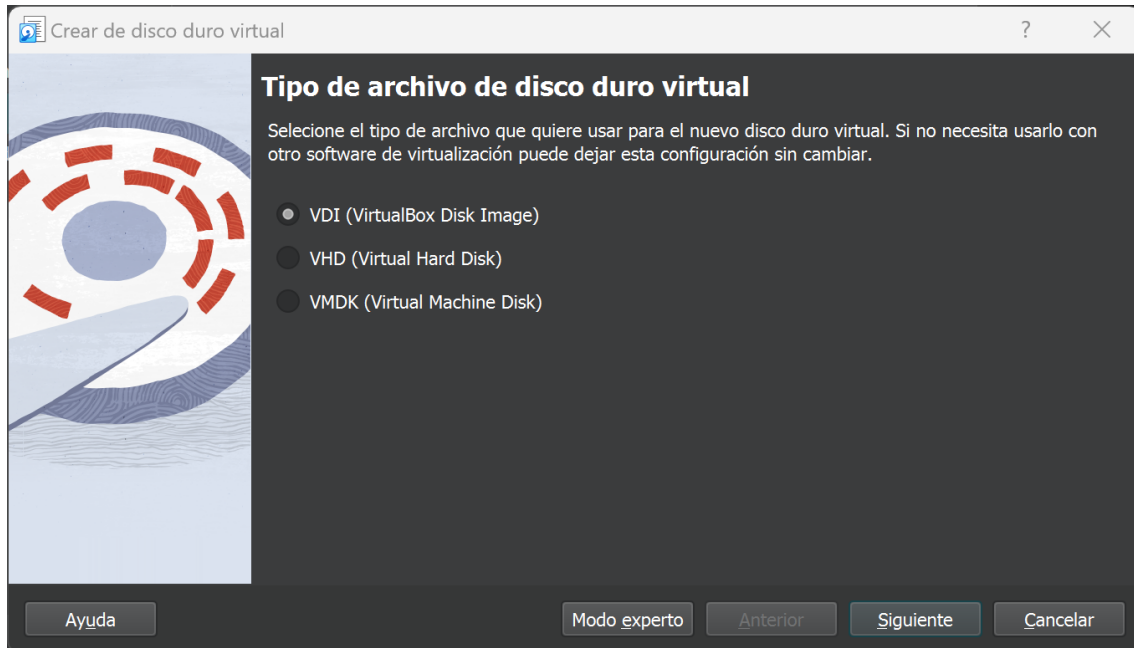
cifrado. Asegúrate de hacer copias de seguridad de los datos importantes antes de realizar el cifrado.



En esta interfaz, se procede a la creación de un disco duro de 1GB en VirtualBox con el objetivo de encriptarlo. Esto se realiza seleccionando el icono de suma [+] para agregar un disco duro.



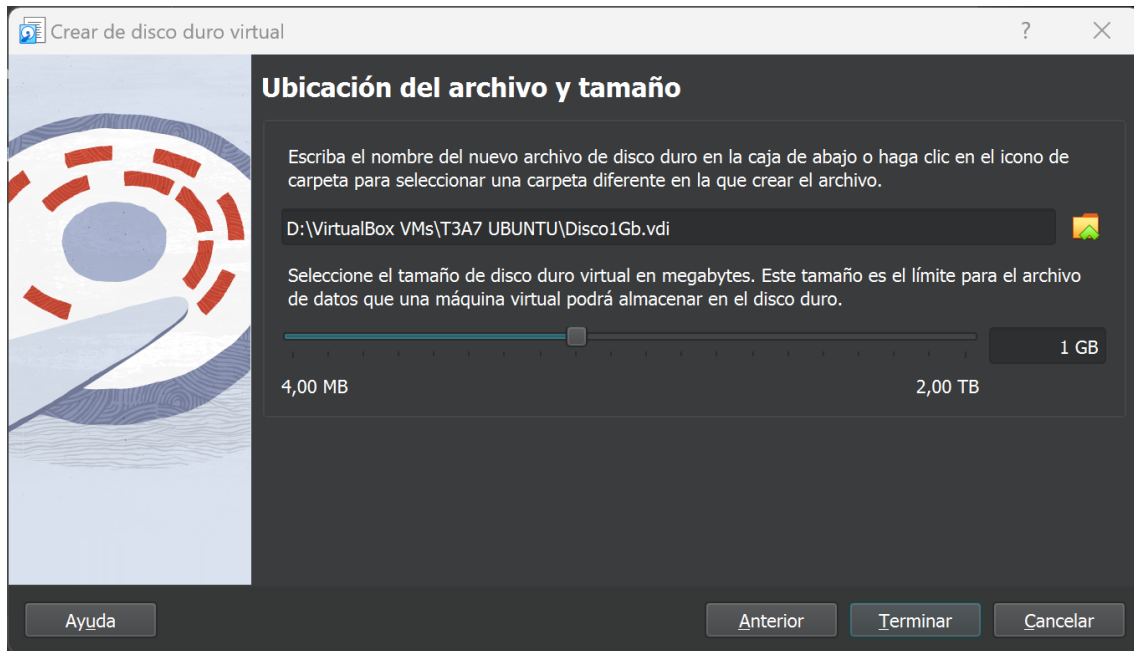
En esta pantalla, se inicia la creación del disco duro de 1GB haciendo clic en el icono de "Crear".



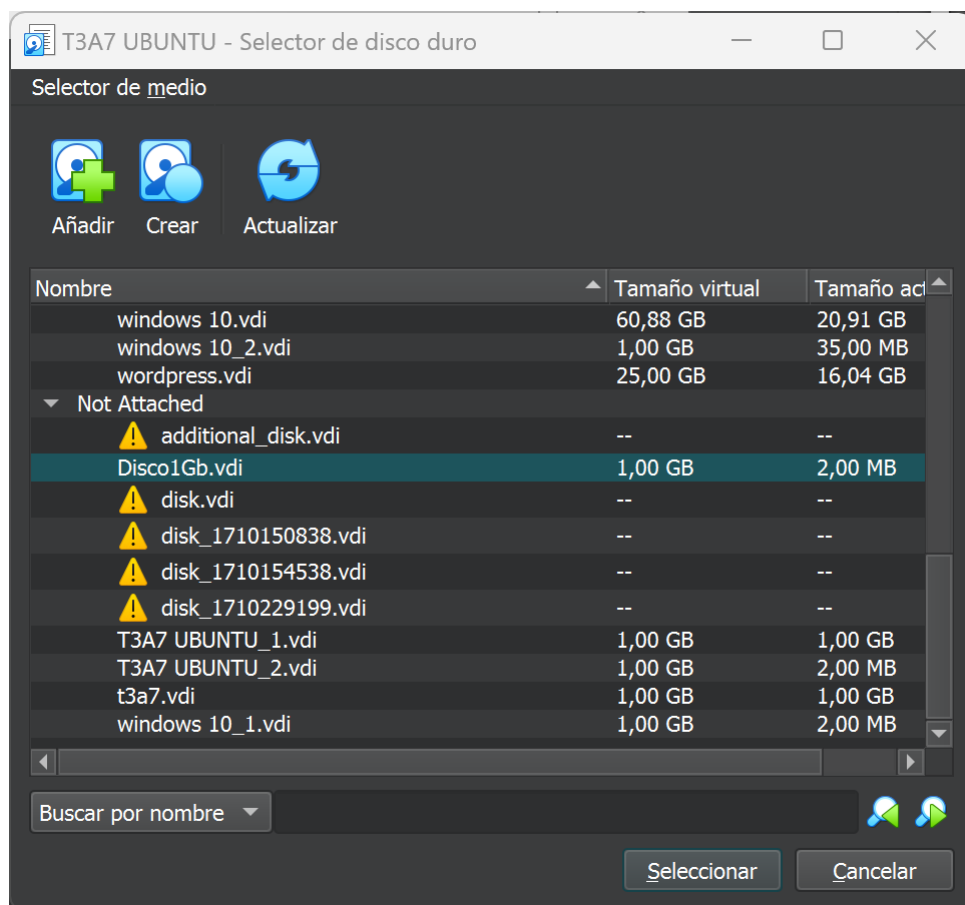
En este paso, se elige la opción "VDI (VirtualBox Disk Image)" y se hace clic en el botón "Siguiente" para avanzar.



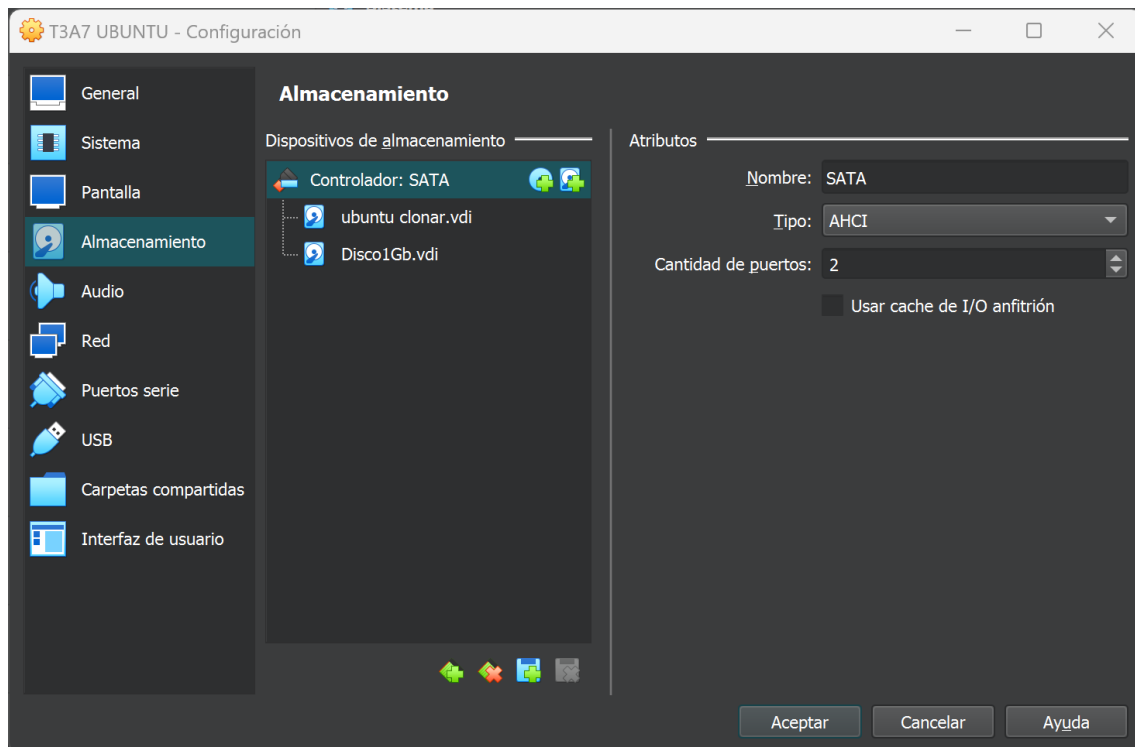
En este paso, se opta por mantener los parámetros predeterminados y se presiona el botón "Siguiente".



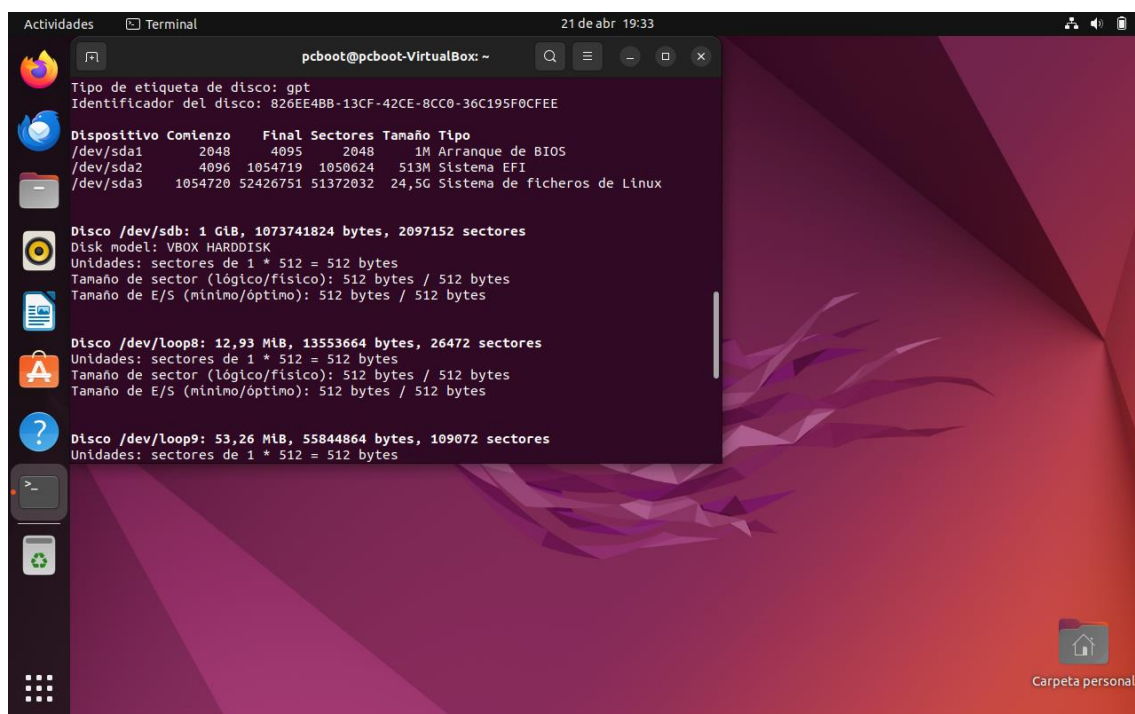
En esta etapa, se selecciona el tamaño de 1GB y se presiona el botón "Terminar" para continuar. El disco duro creado recibe el nombre "Disco1Gb.vdi".



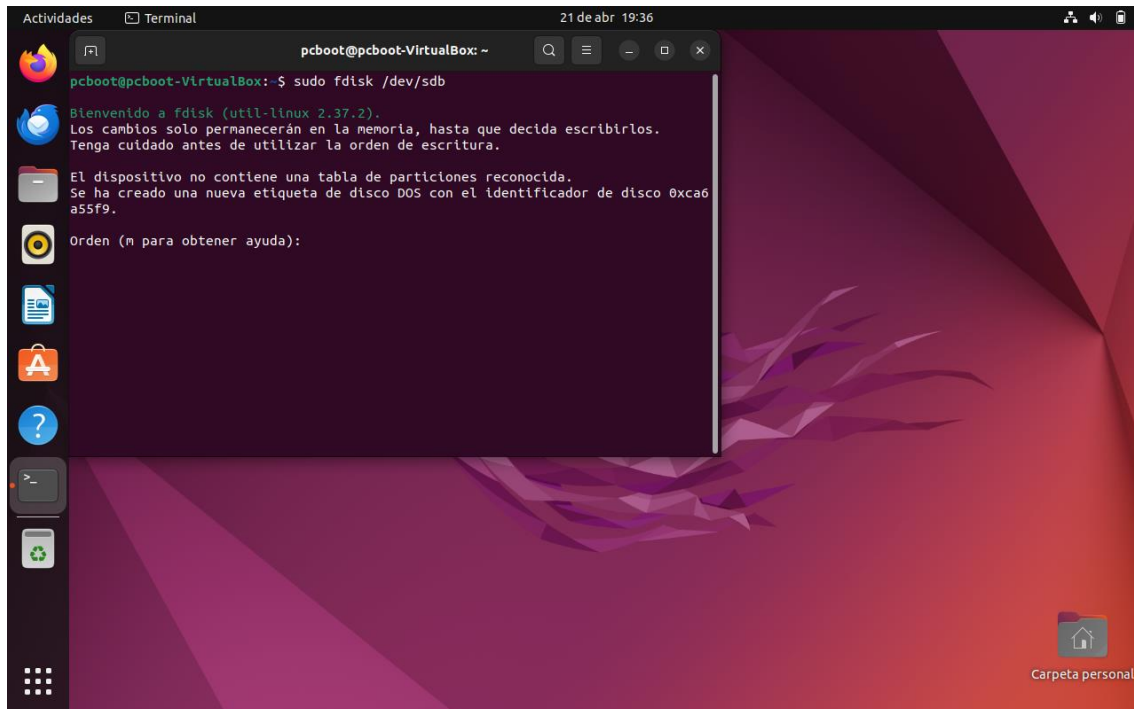
En este paso, se agrega el disco de 1GB a la máquina virtual llamada "Disco1Gb". Posteriormente, se hace clic en el botón "Seleccionar" para confirmar la selección.



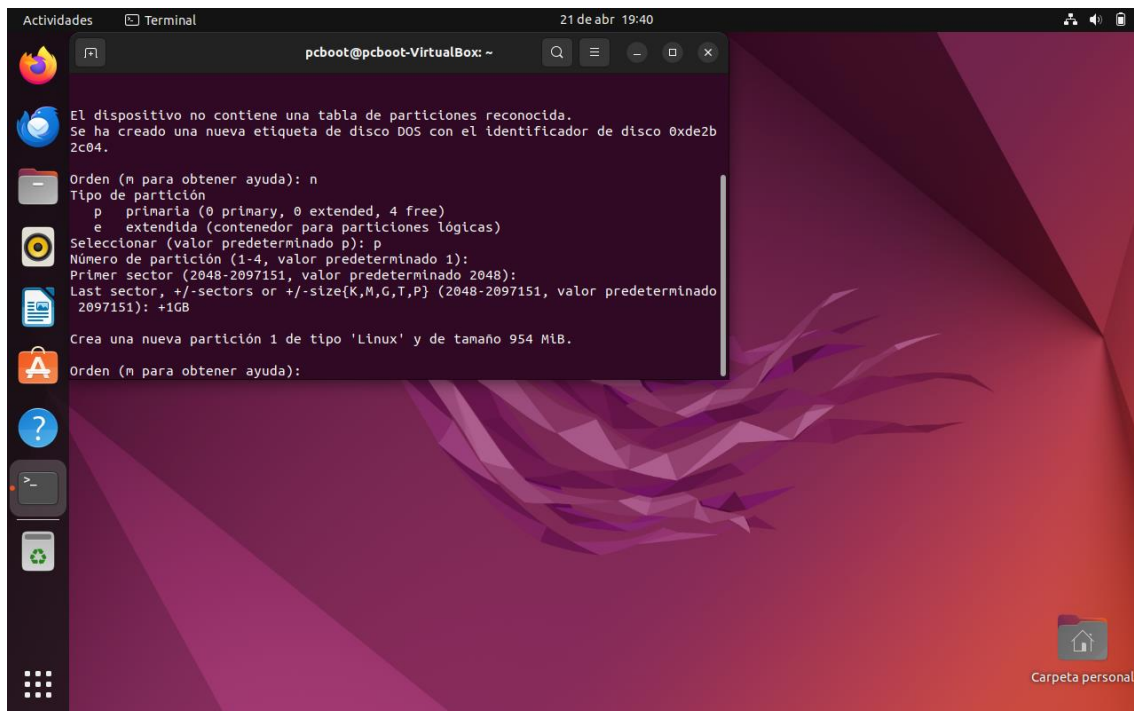
En esta pantalla, se observa que se ha añadido el disco duro de 1GB a la máquina virtual llamada Disco1Gb.vdi. Después de verificar esto, se hace clic en el botón "Aceptar" para confirmar la configuración.



En esta pantalla, se ejecuta el comando `sudo fdisk -l` para listar los discos duros disponibles en la máquina virtual. Tras la ejecución del comando, se identifica que el disco de 1GB se encuentra en la ubicación `/dev/sdb`.

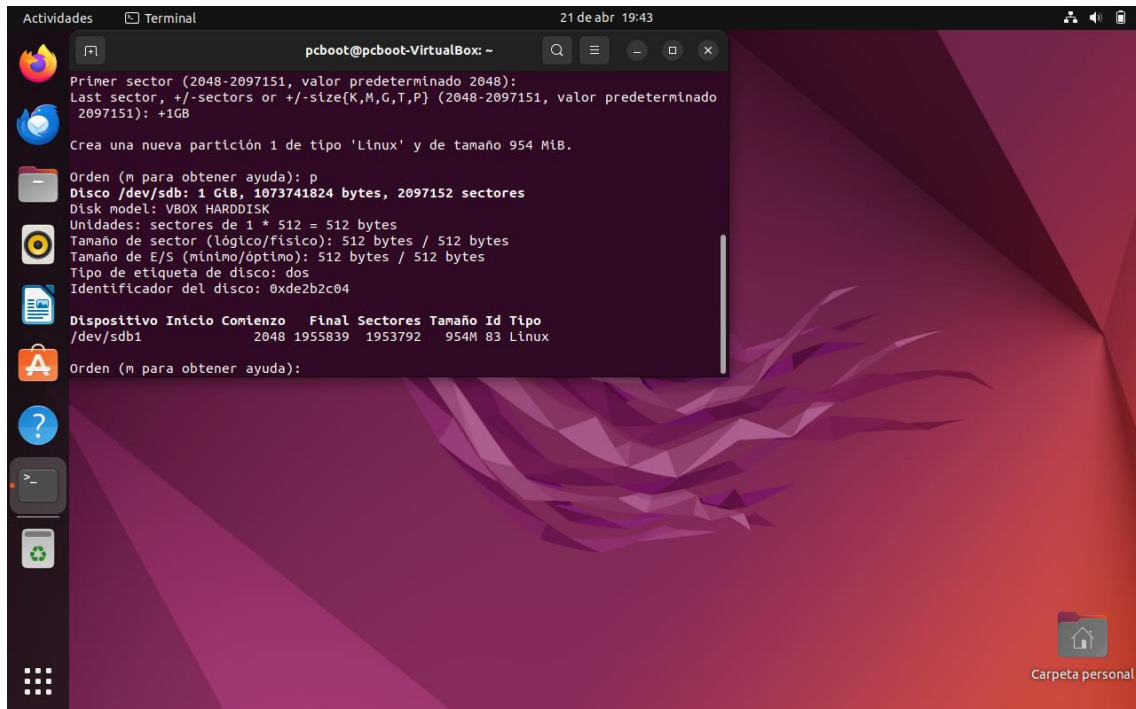


En esta pantalla, se procede a ejecutar el comando `sudo fdisk /dev/sdb` con el fin de formatear y montar la unidad de 1GB.

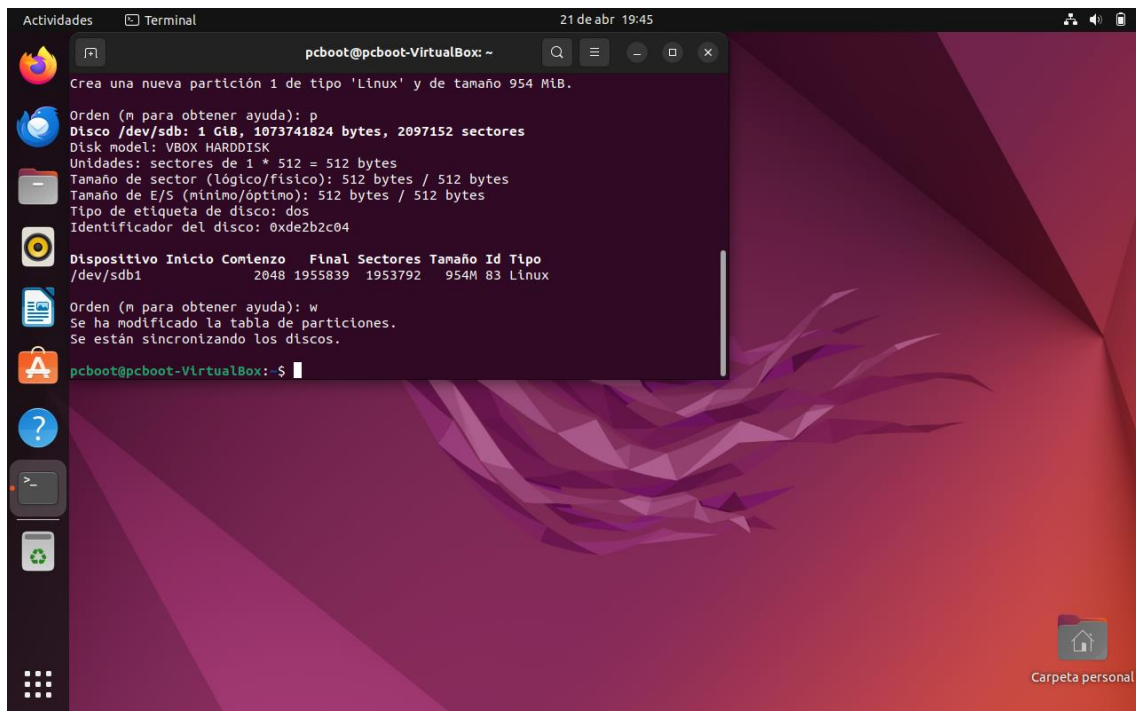


En este paso, se crea la partición utilizando la opción "n" para añadir una partición nueva. Se mantienen los parámetros por defecto, eligiendo la opción "p" para partición primaria, el número de partición se deja como "1", el primer sector se establece en "2048" y se asigna un tamaño de partición de "+1GB". Después de crearla, se observa que el tamaño de la partición es de 954 MiB.

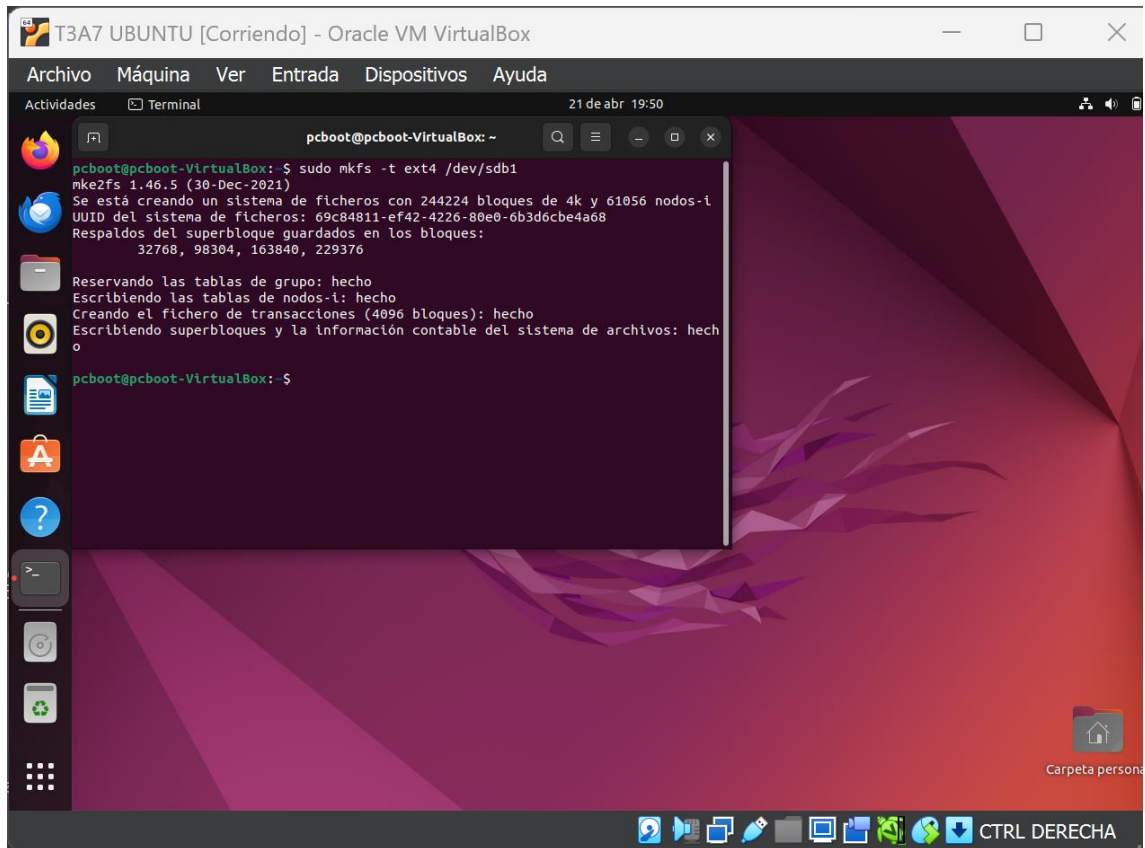




En esta pantalla, se procede a visualizar los dispositivos añadidos utilizando la opción "p".

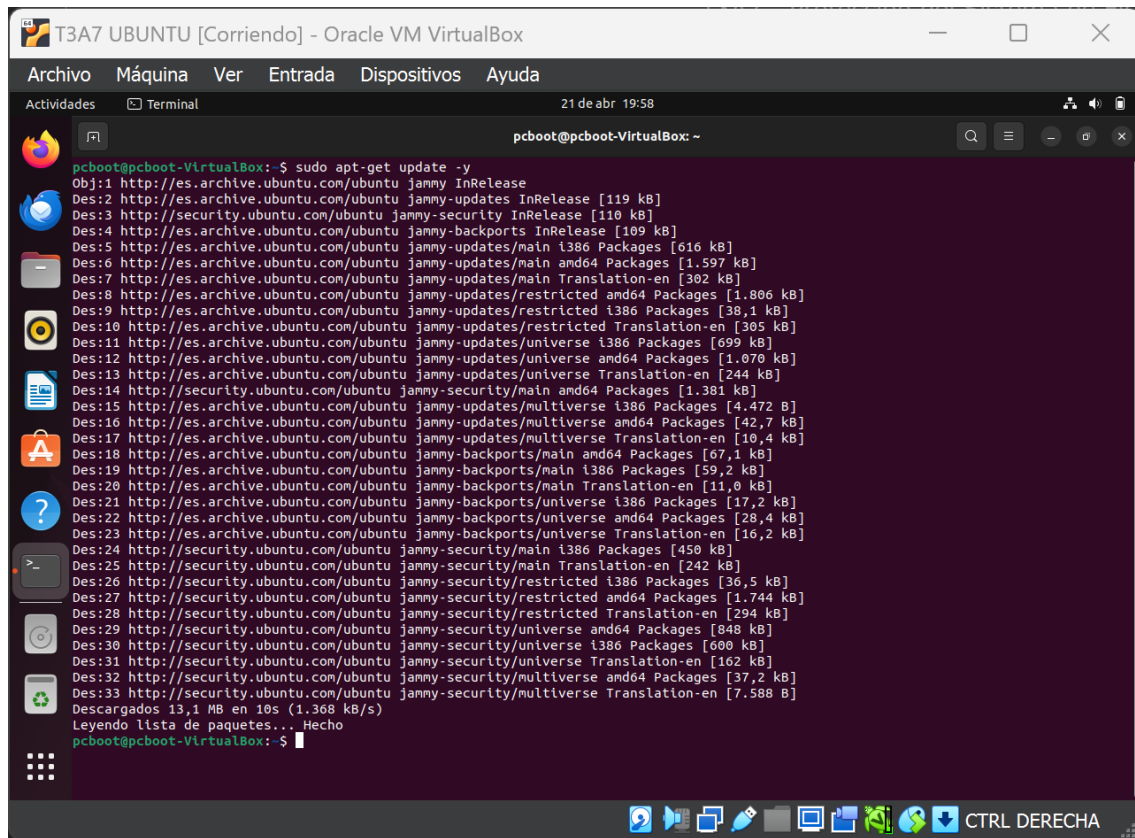


En esta pantalla, se procede a guardar todos los cambios realizados utilizando la opción "w".



En esta pantalla, se procede a formatear la unidad utilizando el comando `sudo mkfs -t ext4 /dev/sdb1`. En la parte derecha, se puede observar el disco duro montado y formateado con éxito, listo para su uso.

## 2.- Instalación y preparación.



```
T3A7 UBUNTU [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Actividades Terminal 21 de abr 19:58
pcboot@pcboot-VirtualBox: ~

pcboot@pcboot-VirtualBox: $ sudo apt-get update -y
Obj:1 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Des:2 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Des:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu jammy-updates/main i386 Packages [616 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1.597 kB]
Des:7 http://es.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [302 kB]
Des:8 http://es.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1.806 kB]
Des:9 http://es.archive.ubuntu.com/ubuntu jammy-updates/restricted i386 Packages [38,1 kB]
Des:10 http://es.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [305 kB]
Des:11 http://es.archive.ubuntu.com/ubuntu jammy-updates/universe i386 Packages [699 kB]
Des:12 http://es.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1.070 kB]
Des:13 http://es.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [244 kB]
Des:14 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1.381 kB]
Des:15 http://es.archive.ubuntu.com/ubuntu jammy-updates/multiverse i386 Packages [4.472 B]
Des:16 http://es.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [42,7 kB]
Des:17 http://es.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [10,4 kB]
Des:18 http://es.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [67,1 kB]
Des:19 http://es.archive.ubuntu.com/ubuntu jammy-backports/main i386 Packages [59,2 kB]
Des:20 http://es.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [11,0 kB]
Des:21 http://es.archive.ubuntu.com/ubuntu jammy-backports/universe i386 Packages [17,2 kB]
Des:22 http://es.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [28,4 kB]
Des:23 http://es.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16,2 kB]
Des:24 http://security.ubuntu.com/ubuntu jammy-security/main i386 Packages [450 kB]
Des:25 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [242 kB]
Des:26 http://security.ubuntu.com/ubuntu jammy-security/restricted i386 Packages [36,5 kB]
Des:27 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1.744 kB]
Des:28 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [294 kB]
Des:29 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [848 kB]
Des:30 http://security.ubuntu.com/ubuntu jammy-security/universe i386 Packages [600 kB]
Des:31 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [162 kB]
Des:32 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37,2 kB]
Des:33 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7.588 B]
Descargados 13,1 MB en 10s (1.368 kB/s)
Leyendo lista de paquetes... Hecho
pcboot@pcboot-VirtualBox: $
```

En esta pantalla, se lleva a cabo la actualización de Ubuntu utilizando el comando `sudo apt-get update -y`. Luego, se presiona la tecla Enter para ejecutar el comando y comenzar el proceso de actualización.

The screenshot shows a terminal window titled "T3A7 UBUNTU [Corriendo] - Oracle VM VirtualBox". The terminal output shows the command `sudo apt-get install cryptsetup -y` being executed. The output includes the following text:

```
pcboot@pcboot-VirtualBox:~$ sudo apt-get install cryptsetup -y
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
 cryptsetup-bin cryptsetup-initramfs
Paquetes sugeridos:
 keyutils
Se instalarán los siguientes paquetes NUEVOS:
 cryptsetup cryptsetup-bin cryptsetup-initramfs
0 actualizados, 3 nuevos se instalarán, 0 para eliminar y 273 no actualizados.
Se necesita descargar 365 kB de archivos.
Se utilizarán 1.245 kB de espacio de disco adicional después de esta operación.
Des:1 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 cryptsetup-bin amd64 2:2.4.3-1ubuntu1.2 [145 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 cryptsetup amd64 2:2.4.3-1ubuntu1.2 [193 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 cryptsetup-initramfs all 2:2.4.3-1ubuntu1.2 [26,2 kB]
Descargados 365 kB en 1s (248 kB/s)
Preconfigurando paquetes ...
Seleccionando el paquete cryptsetup-bin previamente no seleccionado.
(Leyendo la base de datos ... 204394 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../cryptsetup-bin_2%3a2.4.3-1ubuntu1.2_amd64.deb ...
Desempaquetando cryptsetup-bin (2:2.4.3-1ubuntu1.2) ...
Seleccionando el paquete cryptsetup previamente no seleccionado.
Preparando para desempaquetar .../cryptsetup_2%3a2.4.3-1ubuntu1.2_amd64.deb ...
Desempaquetando cryptsetup (2:2.4.3-1ubuntu1.2) ...
Seleccionando el paquete cryptsetup-initramfs previamente no seleccionado.
Preparando para desempaquetar .../cryptsetup-initramfs_2%3a2.4.3-1ubuntu1.2_all.deb ...
Desempaquetando cryptsetup-initramfs (2:2.4.3-1ubuntu1.2) ...
Configurando cryptsetup-bin (2:2.4.3-1ubuntu1.2) ...
Configurando cryptsetup (2:2.4.3-1ubuntu1.2) ...
update-initramfs: deferring update (trigger activated)
Procesando disparadores para man-db (2.10.2-1) ...
Procesando disparadores para initramfs-tools (0.140ubuntu13.2) ...
update-initramfs: Generating /boot/initrd.img-6.5.0-17-generic
pcboot@pcboot-VirtualBox:~$
```

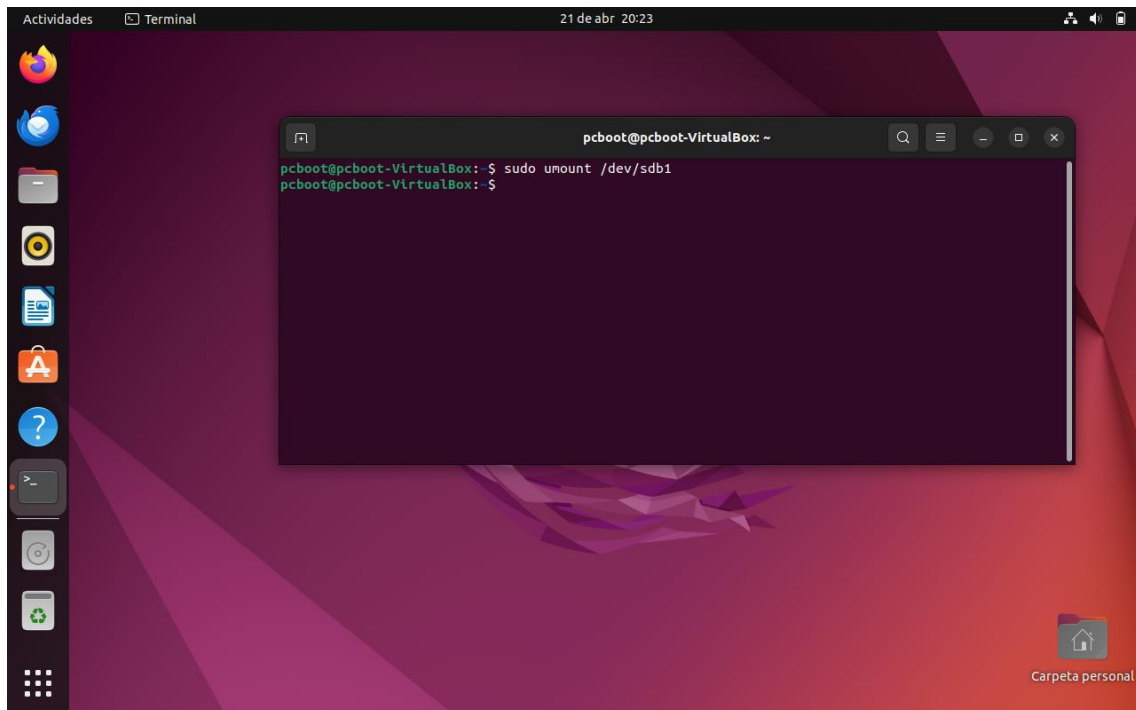
En esta pantalla, se procede a instalar el programa "cryptsetup" utilizando el comando `sudo apt-get install cryptsetup -y`. Después de ingresar el comando, se presiona la tecla Enter para continuar con la instalación del programa.

The screenshot shows a terminal window titled "T3A7 UBUNTU [Corriendo] - Oracle VM VirtualBox". The terminal output shows the command `sudo dd if=/dev/urandom of=/dev/sdb1 bs=2k` being executed. The output includes the following text:

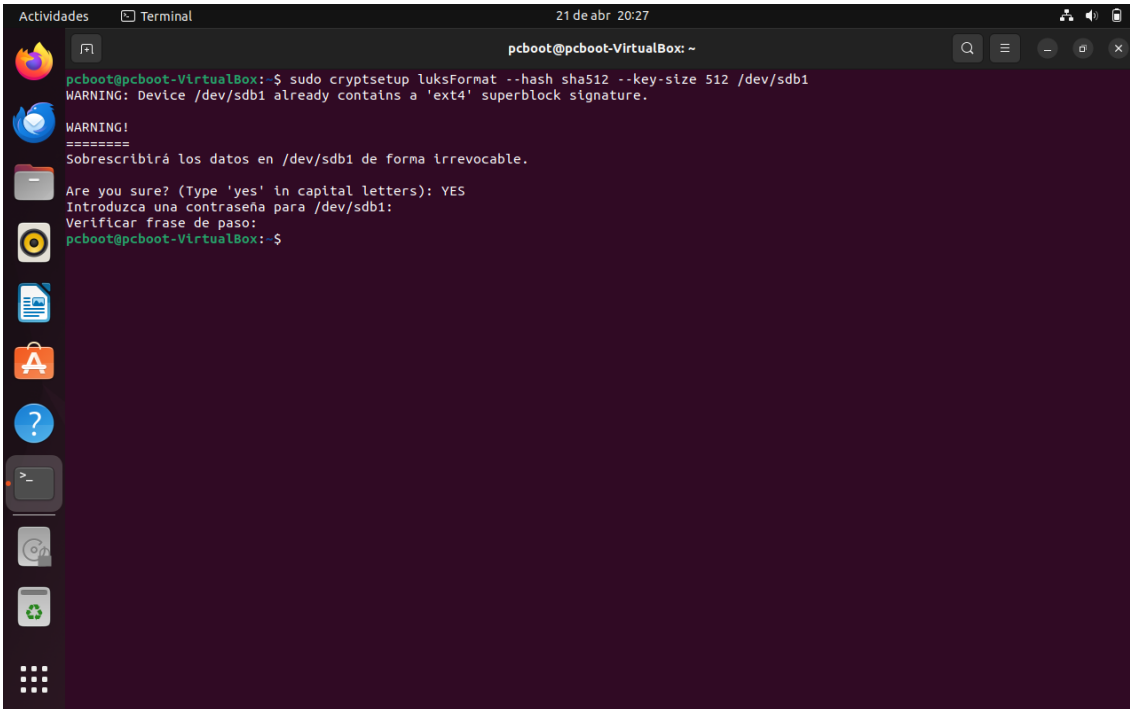
```
pcboot@pcboot-VirtualBox:~$ sudo dd if=/dev/urandom of=/dev/sdb1 bs=2k
dd: error al escribir en '/dev/sdb1': No queda espacio en el dispositivo
488449+0 registros leídos
488448+0 registros escritos
1000341504 bytes (1,0 GB, 954 MiB) copied, 278,557 s, 3,6 MB/s
pcboot@pcboot-VirtualBox:~$
```

En esta pantalla, con el dispositivo identificado y el paquete "cryptsetup" instalado, se procede a preparar el disco para su cifrado. Para ello, se asigna información aleatoria a nivel de bajo nivel que dificulte un criptoanálisis en caso de que el disco caiga en manos no autorizadas. Esto se logra utilizando el comando `dd if=/dev/urandom of=/dev/sdb1 bs=2k`. Luego, se presiona la tecla Enter para ejecutar el comando y llevar a cabo el proceso de asignación de información aleatoria al disco.

### 3.- Cifrado del disco



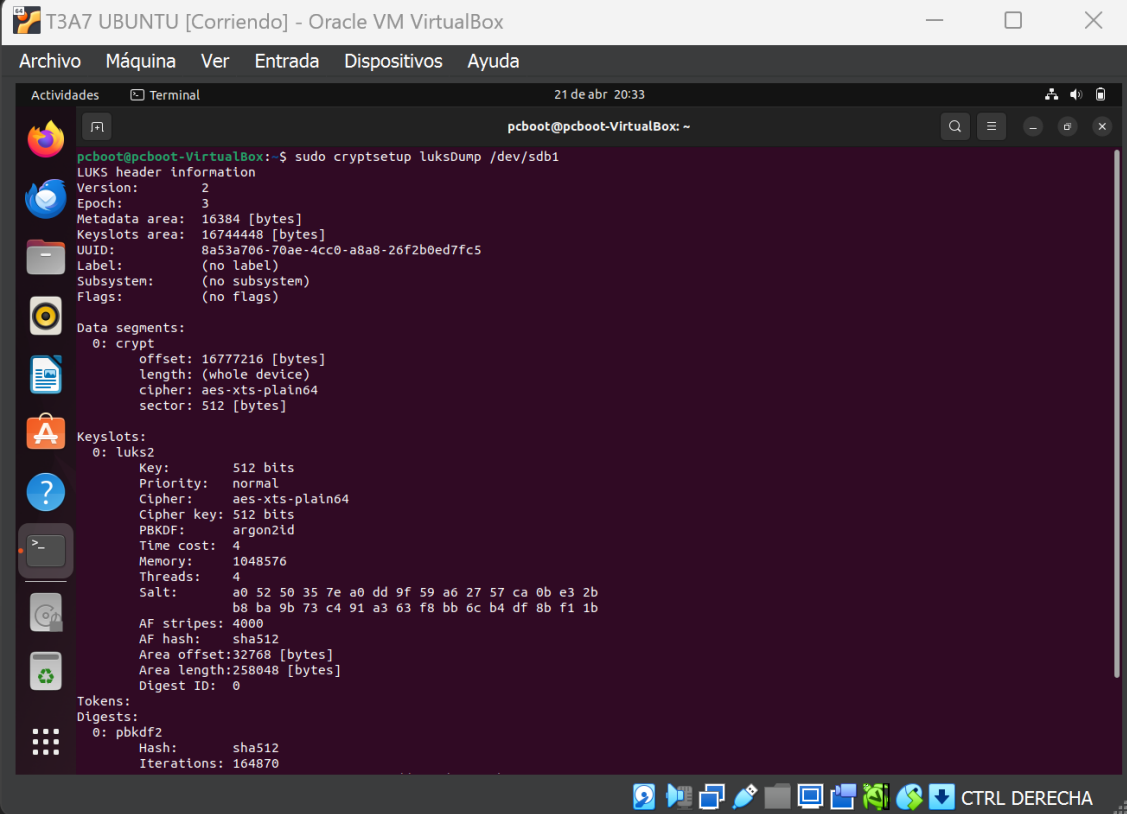
En esta pantalla, se procede a desmontar la unidad `/dev/sdb1` utilizando el comando `sudo umount /dev/sdb1`. Después de ingresar el comando, se presiona la tecla Enter para ejecutarlo y desmontar la unidad correctamente.



```
Actividades Terminal 21 de abr 20:27 pcboot@pcboot-VirtualBox: ~
pcboot@pcboot-VirtualBox:~$ sudo cryptsetup luksFormat --hash sha512 --key-size 512 /dev/sdb1
WARNING: Device /dev/sdb1 already contains a 'ext4' superblock signature.
=====
Sobrescribirá los datos en /dev/sdb1 de forma irrevocable.
Are you sure? (Type 'yes' in capital letters): YES
Introduzca una contraseña para /dev/sdb1:
Verificar frase de paso:
pcboot@pcboot-VirtualBox:~$
```

En esta pantalla, se procede a preparar la unidad del dispositivo para que todos los datos que se almacenen en él se cifren utilizando el algoritmo indicado. Esto se realiza utilizando el comando `sudo cryptsetup luksFormat --hash sha512 --key-size 512 /dev/sdb1`. Luego, se presiona la tecla Enter para ejecutar el comando y comenzar el proceso de cifrado de la unidad con el algoritmo y la contraseña especificados.

## 4.- Ver información sobre sistema cifrado



```
pcboot@pcboot-VirtualBox: ~  
$ sudo cryptsetup luksDump /dev/sdb1  
LUKS header information  
Version: 2  
Epoch: 3  
Metadata area: 16384 [bytes]  
Keyslots area: 16744448 [bytes]  
UUID: 8a53a706-70ae-4cc0-a8a8-26f2b0ed7fc5  
Label: (no label)  
Subsystem: (no subsystem)  
Flags: (no flags)  
  
Data segments:  
0: crypt  
offset: 16777216 [bytes]  
length: (whole device)  
cipher: aes-xts-plain64  
sector: 512 [bytes]  
  
Keyslots:  
0: luks2  
Key: 512 bits  
Priority: normal  
Cipher: aes-xts-plain64  
Cipher key: 512 bits  
PBKDF: argon2id  
Time cost: 4  
Memory: 1048576  
Threads: 4  
Salt: a0 52 50 35 7e a0 dd 9f 59 a0 27 57 ca 0b e3 2b  
b8 ba 9b 73 c4 91 a3 63 f8 bb 6c b4 df 8b f1 1b  
AF stripes: 4000  
AF hash: sha512  
Area offset: 32768 [bytes]  
Area length: 258048 [bytes]  
Digest ID: 0  
  
Tokens:  
Digests:  
0: pbkdf2  
Hash: sha512  
Iterations: 164870
```

En esta pantalla, procedemos a visualizar información sobre la partición encriptada utilizando el comando `sudo cryptsetup luksDump /dev/sdb1`. Luego, presionamos la tecla Enter para ejecutar el comando y obtener los detalles del cifrado del sistema en esa partición.

El comando `sudo cryptsetup luksDump /dev/sdb1` proporcionará información detallada sobre la partición encriptada, incluyendo:

**Encabezado LUKS:** Proporciona información básica sobre el formato de la partición encriptada.

**Versión del formato:** Indica la versión del formato LUKS utilizado.

**UUID (Identificador único universal):** Identifica de forma única la partición encriptada.

**Tamaño de la clave:** Especifica el tamaño de la clave utilizada para el cifrado.

**Función de derivación de clave (KDF):** Describe el algoritmo utilizado para derivar la clave de cifrado a partir de la contraseña proporcionada.

**Parámetros de la función de derivación de clave:** Detalla los parámetros específicos del algoritmo de KDF.



**Algoritmo de cifrado:** Indica el algoritmo de cifrado utilizado para proteger los datos en la partición.

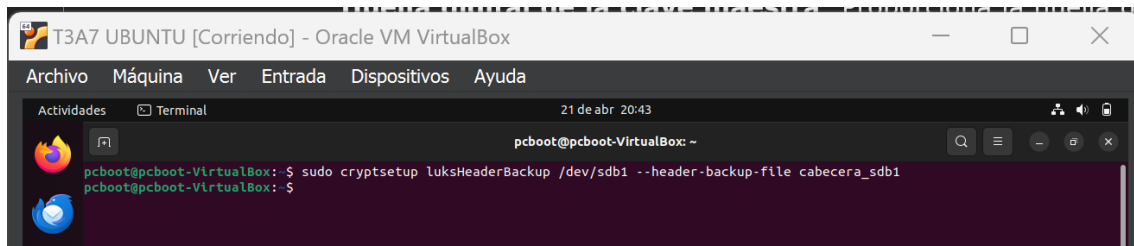
**Modo de operación:** Especifica el modo de operación del algoritmo de cifrado.

**Huella digital de la clave maestra:** Proporciona la huella digital de la clave maestra utilizada para el cifrado.

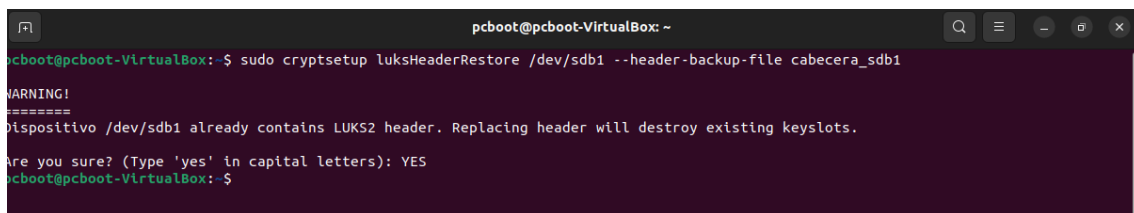
**Huella digital de la clave de derivación de la contraseña:** Ofrece la huella digital de la clave derivada de la contraseña utilizada para el cifrado.

Estos son algunos de los datos clave que se pueden ver al ejecutar el comando `sudo cryptsetup luksDump /dev/sdb1`. La información exacta puede variar dependiendo de la configuración específica de la partición encriptada.

## 5.- Copia de seguridad de la cabecera



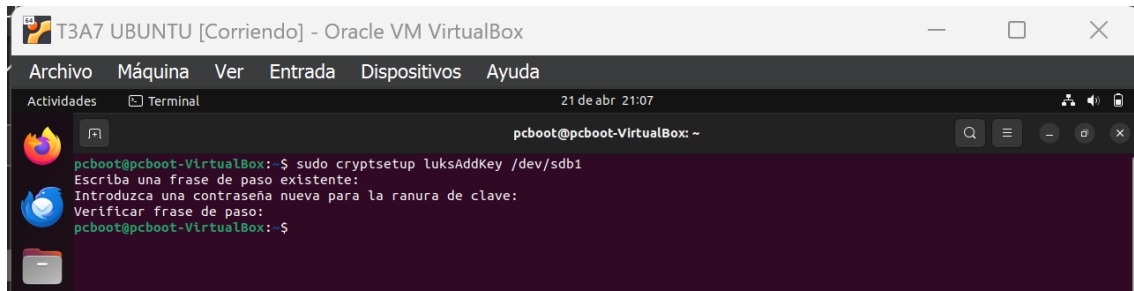
En esta pantalla, procedemos a realizar una copia de seguridad del encabezado utilizando el comando `sudo cryptsetup luksHeaderBackup /dev/sdb1 --header-backup-file cabecera_sdb1`. Luego, presionamos la tecla Enter para ejecutar el comando y crear el archivo de copia de seguridad de la cabecera.



En esta pantalla, procedemos a restaurar la copia del encabezado utilizando el comando `sudo cryptsetup luksHeaderRestore /dev/sdb1 --header-backup-file cabecera_sdb1`. Luego, presionamos la tecla Enter para ejecutar el comando y restaurar el encabezado desde el archivo de copia de seguridad.

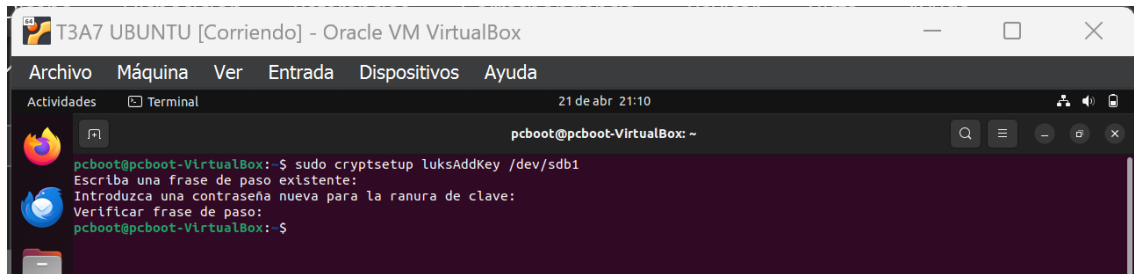


## 6.- Añadir nuevas contraseñas (slots)



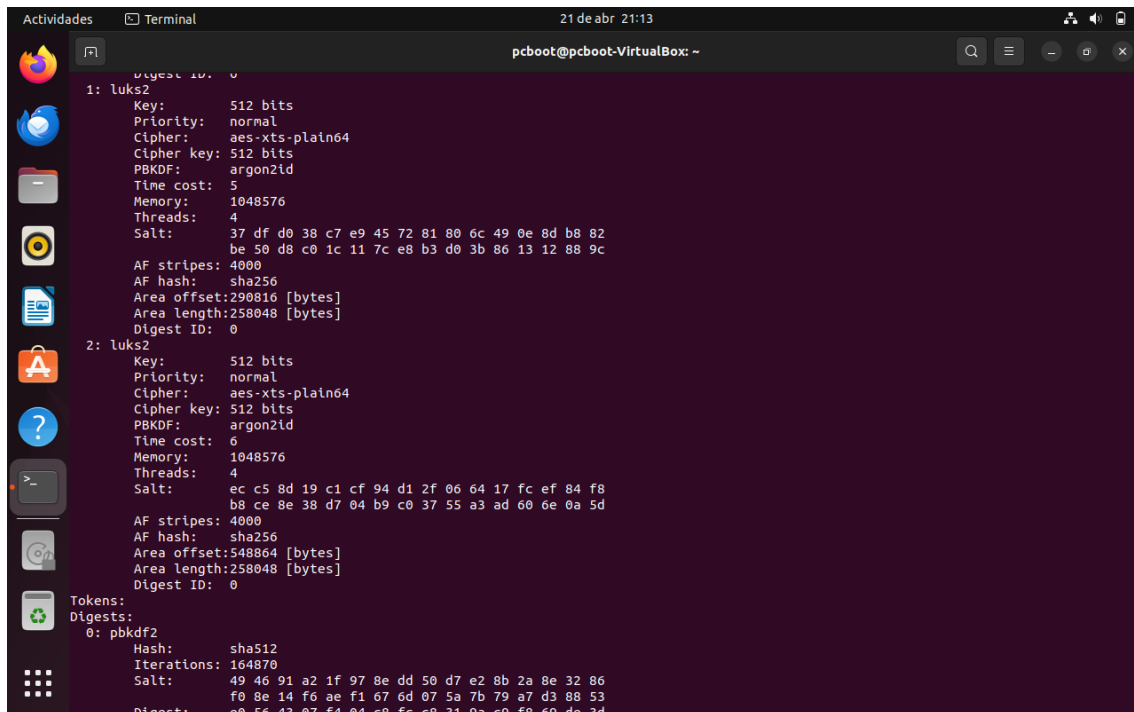
```
T3A7 UBUNTU [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Actividades  Terminal
21 de abr 21:07
pcboot@pcboot-VirtualBox: ~
pcboot@pcboot-VirtualBox:~$ sudo cryptsetup luksAddKey /dev/sdb1
Escriba una frase de paso existente:
Introduzca una contraseña nueva para la ranura de clave:
Verificar frase de paso:
pcboot@pcboot-VirtualBox:~$
```

En esta pantalla, procedemos a añadir un nuevo slot con una nueva clave utilizando el comando `sudo cryptsetup luksAddKey /dev/sdb1`. Luego, presionamos la tecla Enter para ejecutar el comando y añadir el nuevo slot con la clave correspondiente.



```
T3A7 UBUNTU [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Actividades  Terminal
21 de abr 21:10
pcboot@pcboot-VirtualBox: ~
pcboot@pcboot-VirtualBox:~$ sudo cryptsetup luksAddKey /dev/sdb1
Escriba una frase de paso existente:
Introduzca una contraseña nueva para la ranura de clave:
Verificar frase de paso:
pcboot@pcboot-VirtualBox:~$
```

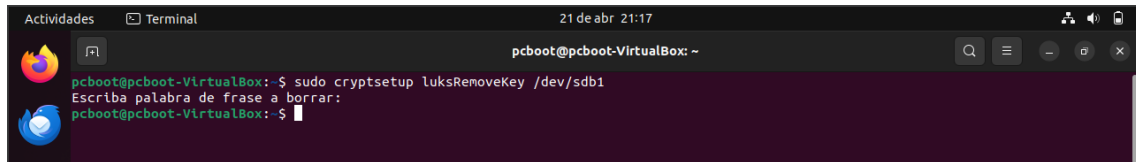
n esta pantalla, procedemos a añadir el segundo slot con una nueva clave utilizando el comando `sudo cryptsetup luksAddKey /dev/sdb1`. Luego, presionamos la tecla Enter para ejecutar el comando y añadir el nuevo slot con la clave correspondiente.



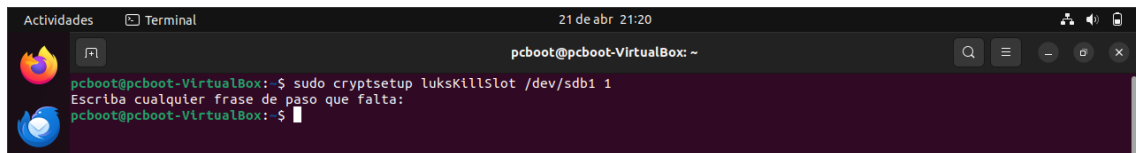
```
Actividades  Terminal
21 de abr 21:13
pcboot@pcboot-VirtualBox: ~
pcboot@pcboot-VirtualBox:~$ sudo cryptsetup luksAddKey /dev/sdb1
Digest ID: 0
1: luks2
  Key: 512 bits
  Priority: normal
  Cipher: aes-xts-plain64
  Cipher key: 512 bits
  PBKDF: argon2id
  Time cost: 5
  Memory: 1048576
  Threads: 4
  Salt: 37 df d0 38 c7 e9 45 72 81 80 6c 49 0e 8d b8 82
        be 50 d8 c0 1c 11 7c e8 b3 d0 3b 86 13 12 88 9c
  AF stripes: 4000
  AF hash: sha256
  Area offset: 290816 [bytes]
  Area length: 258048 [bytes]
  Digest ID: 0
2: luks2
  Key: 512 bits
  Priority: normal
  Cipher: aes-xts-plain64
  Cipher key: 512 bits
  PBKDF: argon2id
  Time cost: 6
  Memory: 1048576
  Threads: 4
  Salt: ec c5 8d 19 c1 cf 94 d1 2f 06 64 17 fc ef 84 f8
        b8 ce 8e 38 d7 04 b9 c0 37 55 a3 ad 60 6e 0a 5d
  AF stripes: 4000
  AF hash: sha256
  Area offset: 548864 [bytes]
  Area length: 258048 [bytes]
  Digest ID: 0
Tokens:
Digests:
0: pbkdf2
  Hash: sha512
  Iterations: 164870
  Salt: 49 46 91 a2 1f 97 8e dd 50 d7 e2 8b 2a 8e 32 86
        f0 8e 14 f6 ae f1 67 6d 07 5a 7b 79 a7 d3 88 53
  Digest: e0 56 43 07 f4 04 c8 fc c8 31 9a c9 f8 69 de 3d
```

En esta pantalla, procedemos a visualizar la información de los dos slots que hemos creado utilizando el comando `sudo cryptsetup luksDump /dev/sdb1`. Luego, presionamos la tecla Enter para ejecutar el comando y obtener los detalles de los slots de clave en la partición encriptada.

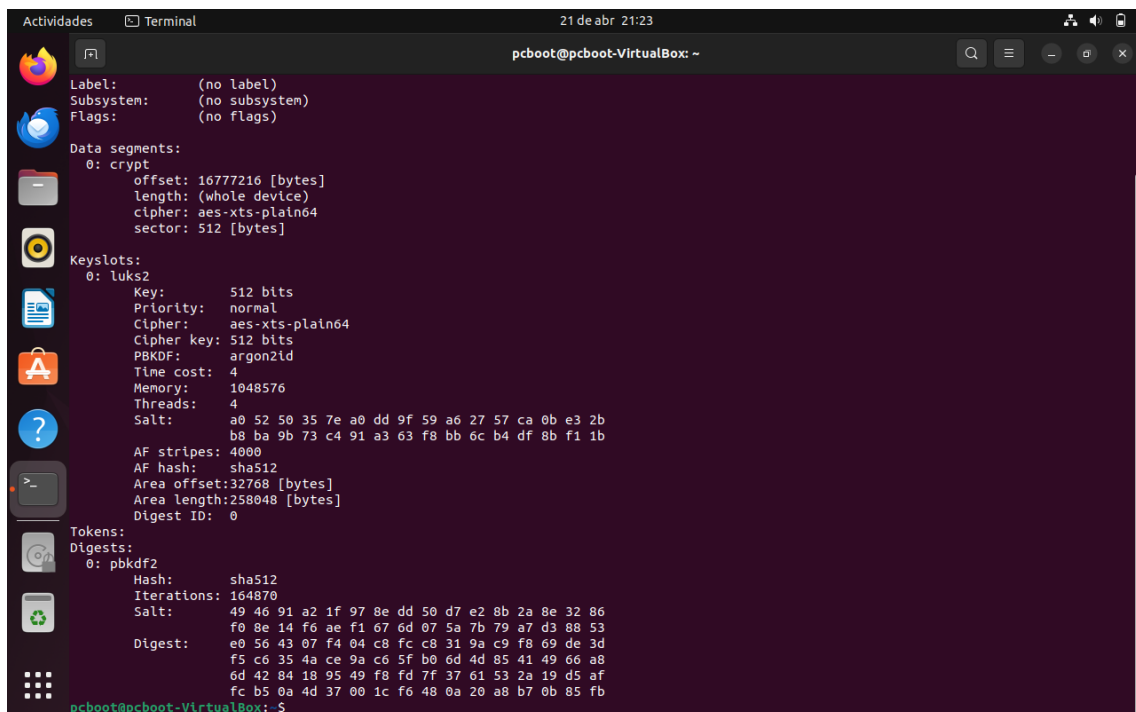
## 7.- Eliminar contraseña (slots)



En esta pantalla, procedemos a eliminar un slot de clave utilizando el comando `sudo cryptsetup luksKillSlot /dev/sdb1`. Luego, presionamos la tecla Enter para ejecutar el comando y eliminar el slot de clave especificado.

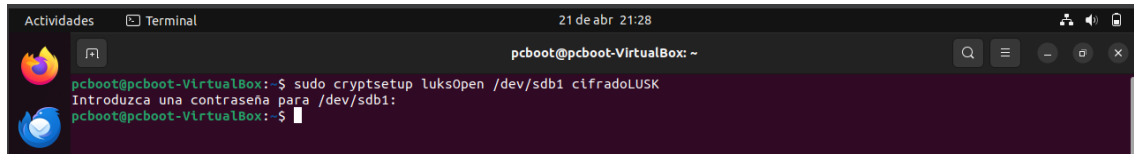


En esta pantalla, procedemos a eliminar un slot de clave utilizando el número de slot específico con el comando `cryptsetup luksKillSlot /dev/sdb1 1`. Luego, presionamos la tecla Enter para ejecutar el comando y eliminar el slot de clave correspondiente.



En esta pantalla, procedemos a verificar si los slots se han eliminado utilizando el comando `cryptsetup luksDump /dev/sdb1`. Luego, presionamos la tecla Enter para ejecutar el comando y observamos que los slots que habíamos creado se han eliminado correctamente.

## 8.- Abrir el contenedor cifrado



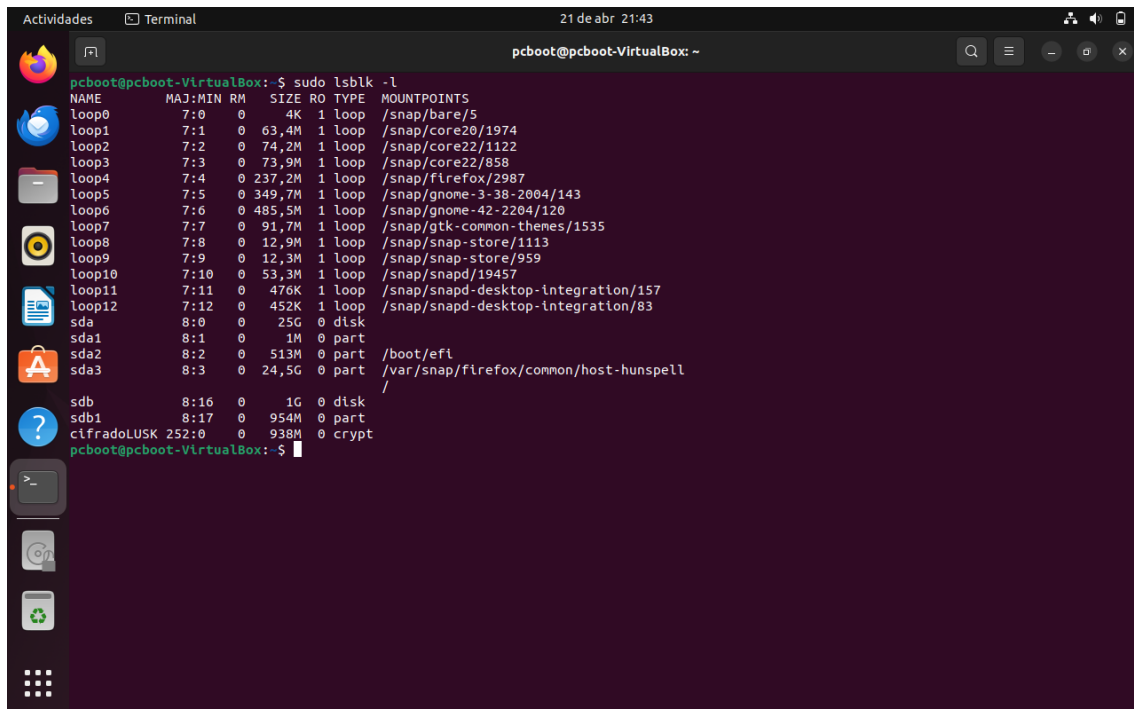
```
pcboot@pcboot-VirtualBox: ~  
$ sudo cryptsetup luksOpen /dev/sdb1 cifradoLUSK  
Introduzca una contraseña para /dev/sdb1:  
pcboot@pcboot-VirtualBox: ~
```

En esta pantalla, procedemos a asignar un nombre para mapear a la unidad cifrada utilizando el comando `sudo cryptsetup luksOpen /dev/sdb1 cifradoLUSK`. Luego, presionamos la tecla Enter para ejecutar el comando y asignar el nombre al dispositivo encriptado.



```
pcboot@pcboot-VirtualBox: ~  
$ sudo mkfs.ext4 /dev/mapper/cifradoLUSK  
mke2fs 1.46.5 (30-Dec-2021)  
Se está creando un sistema de ficheros con 240128 bloques de 4k y 60032 nodos-i  
UUID del sistema de ficheros: f9b722de-761f-43ab-9e64-393aa82ec4c2  
Respalos del superbloque guardados en los bloques:  
32768, 98304, 163840, 229376  
Reservando las tablas de grupo: hecho  
Escribiendo las tablas de nodos-i: hecho  
Creando el fichero de transacciones (4096 bloques): hecho  
Escribiendo superbloques y la información contable del sistema de archivos: hecho  
pcboot@pcboot-VirtualBox: ~
```

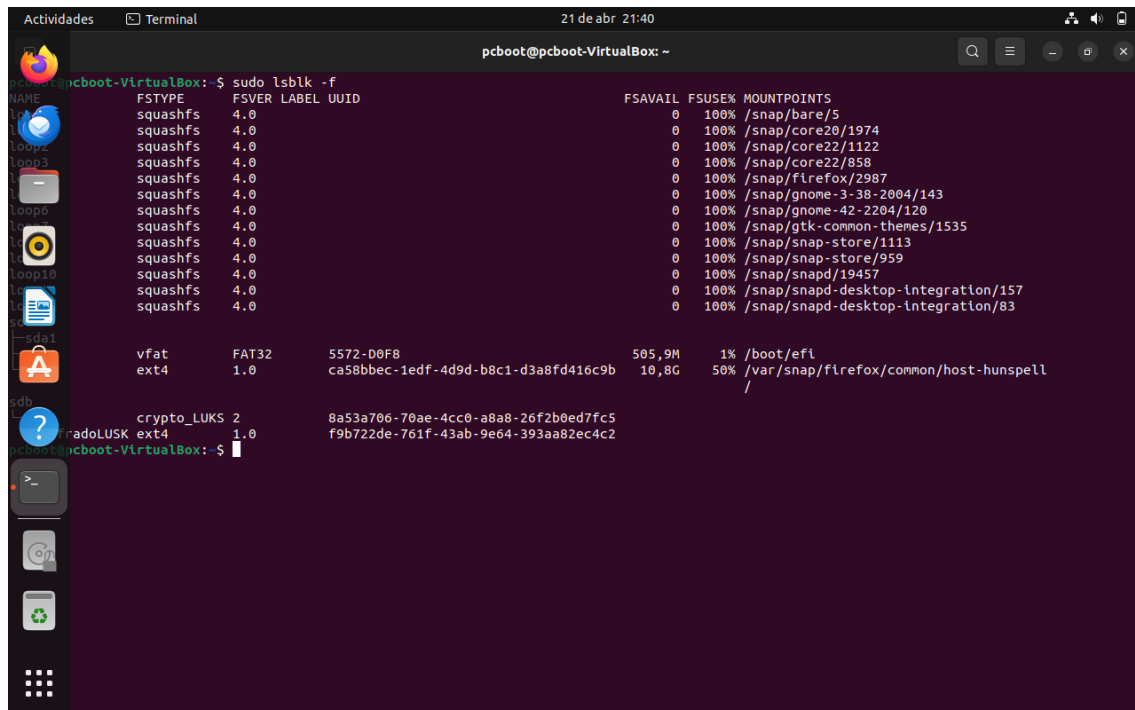
En esta pantalla, procedemos a asignar el nombre que la máquina virtual pueda utilizar para mapear la unidad con el comando `sudo mkfs.ext4 /dev/mapper/cifradoLUSK`. Luego, presionamos la tecla Enter para ejecutar el comando y formatear la unidad encriptada con el sistema de archivos ext4.



```
pcboot@pcboot-VirtualBox: ~  
$ sudo lsblk -l  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS  
loop0        7:0    0     4K  1 loop /snap/bare/5  
loop1        7:1    0    63,4M  1 loop /snap/core20/1974  
loop2        7:2    0    74,2M  1 loop /snap/core22/1122  
loop3        7:3    0    73,9M  1 loop /snap/core22/858  
loop4        7:4    0   237,2M  1 loop /snap/firefox/2987  
loop5        7:5    0   349,7M  1 loop /snap/gnome-3-38-2004/143  
loop6        7:6    0   485,5M  1 loop /snap/gnome-42-2204/120  
loop7        7:7    0    91,7M  1 loop /snap/gtk-common-themes/1535  
loop8        7:8    0    12,9M  1 loop /snap/snap-store/1113  
loop9        7:9    0    12,3M  1 loop /snap/snapd/959  
loop10       7:10   0    53,3M  1 loop /snap/snapd/19457  
loop11       7:11   0    476K  1 loop /snap/snapd-desktop-integration/157  
loop12       7:12   0    452K  1 loop /snap/snapd-desktop-integration/83  
sda          8:0    0     25G  0 disk  
sda1         8:1    0      1M  0 part  
sda2         8:2    0    513M  0 part /boot/efi  
sda3         8:3    0    24,5G  0 part /var/snap/firefox/common/host-hunspell  
sdb          8:16   0      1G  0 disk  
sdb1         8:17   0    954M  0 part  
cifradoLUSK 252:0   0    938M  0 crypt  
pcboot@pcboot-VirtualBox: ~
```

En esta pantalla, procedemos a visualizar la información, incluyendo el nombre que hemos creado para la unidad cifrada, utilizando el comando `sudo lsblk -l`. Luego, presionamos la tecla Enter para ejecutar el comando y

listar los discos disponibles, incluyendo el nombre asignado a la unidad cifrada.

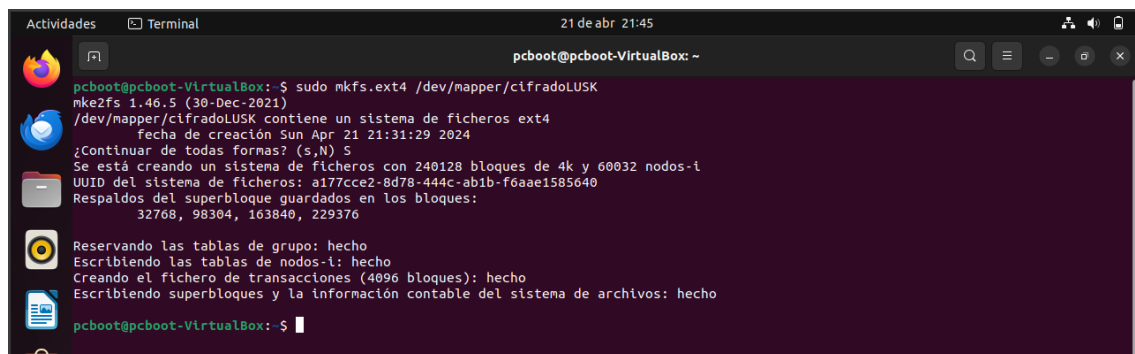


```
pcboot@pcboot-VirtualBox: ~$ sudo lsblk -f
```

NAME	FSTYPE	FSVER	LABEL	UUID	FS-AVAIL	FS-USE%	MOUNTPOINTS
squashfs	4.0				0	100%	/snap/bare/5
squashfs	4.0				0	100%	/snap/core20/1974
squashfs	4.0				0	100%	/snap/core22/1122
squashfs	4.0				0	100%	/snap/core22/858
squashfs	4.0				0	100%	/snap/firefox/2987
squashfs	4.0				0	100%	/snap/gnome-3-38-2004/143
squashfs	4.0				0	100%	/snap/gnome-42-2204/120
squashfs	4.0				0	100%	/snap/gtk-common-themes/1535
squashfs	4.0				0	100%	/snap/snap-store/1113
squashfs	4.0				0	100%	/snap/snap-store/959
squashfs	4.0				0	100%	/snap/snapd/19457
squashfs	4.0				0	100%	/snap/snapd-desktop-integration/157
squashfs	4.0				0	100%	/snap/snapd-desktop-integration/83
vfat	FAT32			5572-D0F8	505,9M	1%	/boot/efi
ext4	1.0			ca58bbec-1edf-4d9d-b8c1-d3a8fd416c9b	10,8G	50%	/var/snap/firefox/common/host-hunspell
crypto_LUKS	2			8a53a706-70ae-4cc0-a8a8-26f2b0ed7fc5			
ext4	1.0			f9b722de-761f-43ab-9e64-393aa82ec4c2			

En esta pantalla, procedemos a ejecutar el comando `sudo lsblk -f` para visualizar las unidades del sistema. Luego, presionamos la tecla Enter para ejecutar el comando y obtener la lista de unidades, incluyendo información sobre el sistema de archivos de cada una.

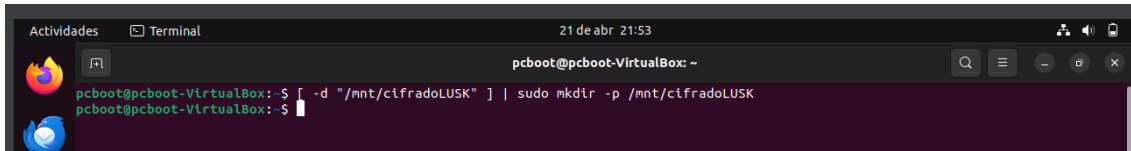
## 9.- Formateo y montaje de la unidad



```
pcboot@pcboot-VirtualBox: ~$ sudo mkfs.ext4 /dev/mapper/cifradoLUKS
```

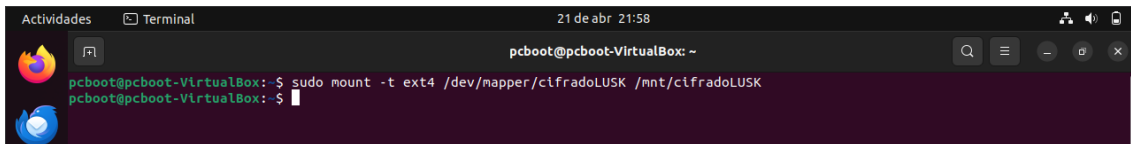
mkfs2fs 1.46.5 (30-Dec-2021)  
/dev/mapper/cifradoLUKS contiene un sistema de ficheros ext4  
fecha de creación Sun Apr 21 21:31:29 2024  
¿Continuar de todas formas? (s,n) S  
Se está creando un sistema de ficheros con 240128 bloques de 4k y 60032 nodos-l  
UUID del sistema de ficheros: a177cce2-8d78-444c-ab1b-f6aae1585640  
Respaldo del superbloque guardados en los bloques:  
32768, 98304, 163840, 229376  
Reservando las tablas de grupo: hecho  
Escribiendo las tablas de nodos-l: hecho  
Creando el fichero de transacciones (4096 bloques): hecho  
Escribiendo superbloques y la información contable del sistema de archivos: hecho

En esta pantalla, procedemos a formatear la unidad utilizando el comando `mkfs.ext4 /dev/mapper/cifradoLUKS`. Luego, presionamos la tecla Enter para ejecutar el comando y formatear la unidad encriptada con el sistema de archivos ext4.



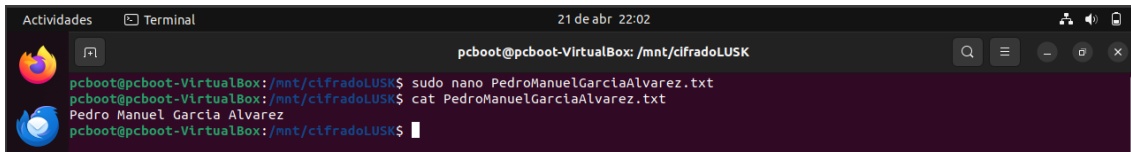
```
pcboot@pcboot-VirtualBox: ~  
pcboot@pcboot-VirtualBox:~$ [ -d "/mnt/cifradoLUSK" ] || sudo mkdir -p /mnt/cifradoLUSK  
pcboot@pcboot-VirtualBox:~$
```

En esta pantalla, procedemos a ejecutar el comando [ `"-d /mnt/cifradoLUSK"` ] || `sudo mkdir -p /mnt/cifradoLUSK` para verificar si el directorio ya existe. Luego, presionamos la tecla Enter para ejecutar el comando, si no existe el directorio no daría falso.



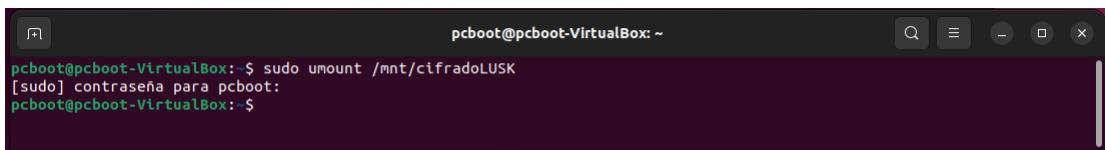
```
pcboot@pcboot-VirtualBox:~$ sudo mount -t ext4 /dev/mapper/cifradoLUSK /mnt/cifradoLUSK  
pcboot@pcboot-VirtualBox:~$
```

En esta pantalla, procedemos a montar la unidad utilizando el comando `sudo mount -t ext4 /dev/mapper/cifradoLUSK /mnt/cifradoLUSK`. Luego, presionamos la tecla Enter para ejecutar el comando y montar la unidad cifrada en el directorio `/mnt/cifradoLUSK`.



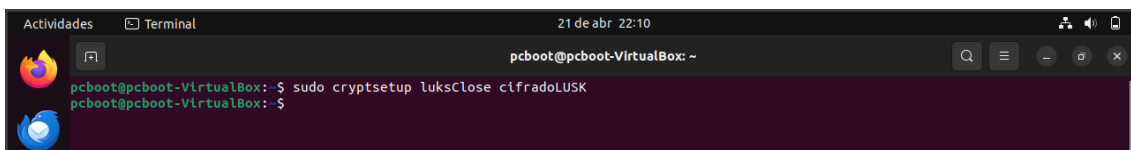
```
pcboot@pcboot-VirtualBox: /mnt/cifradoLUSK  
pcboot@pcboot-VirtualBox:/mnt/cifradoLUSK$ sudo nano PedroManuelGarciaAlvarez.txt  
pcboot@pcboot-VirtualBox:/mnt/cifradoLUSK$ cat PedroManuelGarciaAlvarez.txt  
Pedro Manuel Garcia Alvarez  
pcboot@pcboot-VirtualBox:/mnt/cifradoLUSK$
```

En esta pantalla, procedemos a crear un archivo dentro de la unidad cifrada utilizando el comando `sudo nano /mnt/cifradoLUSK/PedroManuelGarciaAlvarez.txt`. Luego, visualizamos el contenido del archivo utilizando el comando `cat /mnt/cifradoLUSK/PedroManuelGarciaAlvarez.txt`. Finalmente, presionamos la tecla Enter para ejecutar el comando y visualizar el contenido del archivo.



```
pcboot@pcboot-VirtualBox: ~  
pcboot@pcboot-VirtualBox:~$ sudo umount /mnt/cifradoLUSK  
[sudo] contraseña para pcboot:  
pcboot@pcboot-VirtualBox:~$
```

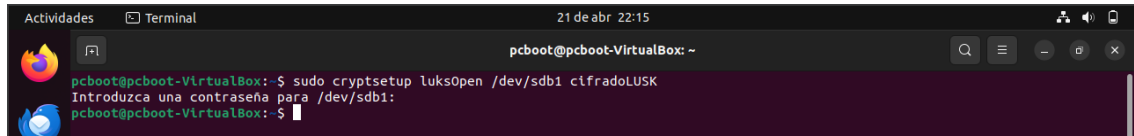
En esta pantalla, procedemos a desmontar la unidad cifrada utilizando el comando `sudo umount /mnt/cifradoLUSK`. Luego, presionamos la tecla Enter para ejecutar el comando y desmontar la unidad correctamente.



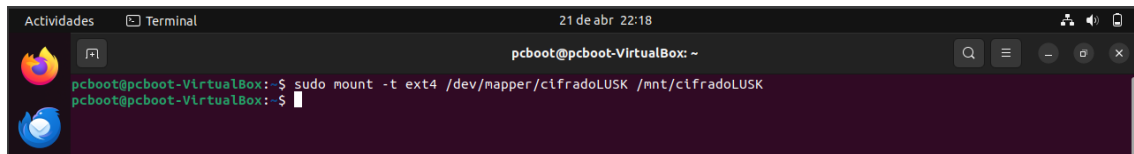
```
pcboot@pcboot-VirtualBox:~$ sudo cryptsetup luksClose cifradoLUSK  
pcboot@pcboot-VirtualBox:~$
```

En esta pantalla, procedemos a cerrar el volumen cifrado utilizando el comando `sudo cryptsetup luksClose cifradoLUSK`. Luego, presionamos la tecla Enter para ejecutar el comando y cerrar correctamente el volumen cifrado.

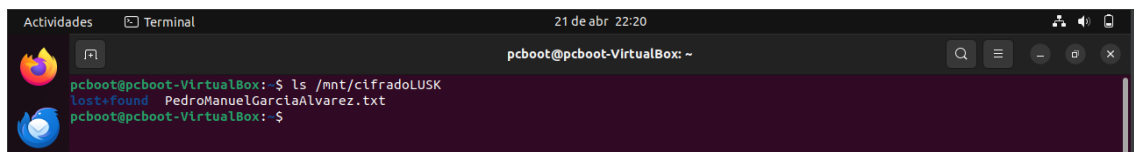
## 11.- Reapertura del disco



En esta pantalla, procedemos a iniciar un nuevo uso del dispositivo utilizando el comando `sudo cryptsetup luksOpen /dev/sdb1 cifradoLUSK`. Luego, presionamos la tecla Enter para ejecutar el comando y comenzar el uso del dispositivo con el nombre cifradoLUSK.

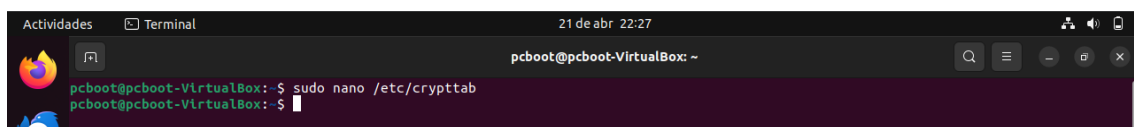


En esta pantalla, procedemos a montar la unidad utilizando el comando `sudo mount -t ext4 /dev/mapper/cifradoLUSK /mnt/cifradoLUSK`. Luego, presionamos la tecla Enter para ejecutar el comando y montar la unidad cifrada en el directorio /mnt/cifradoLUSK.

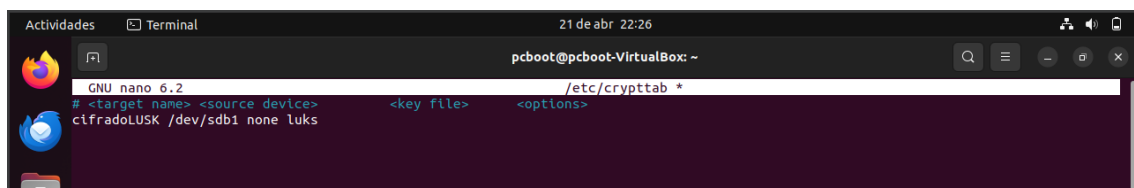


En esta pantalla, procedemos a verificar y comprobar si el archivo ya creado está presente utilizando el comando `ls /mnt/cifradoLUSK`. Luego, presionamos la tecla Enter para ejecutar el comando y listar los archivos dentro del directorio /mnt/cifradoLUSK.

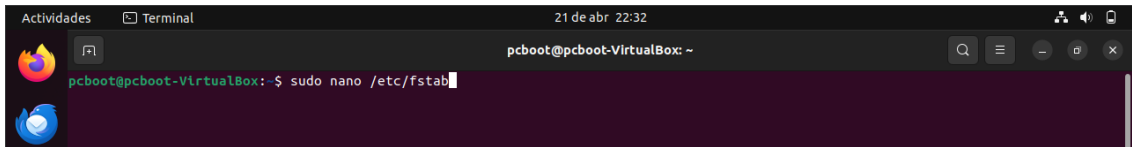
## 12.- Montaje automático del volumen en el arranque



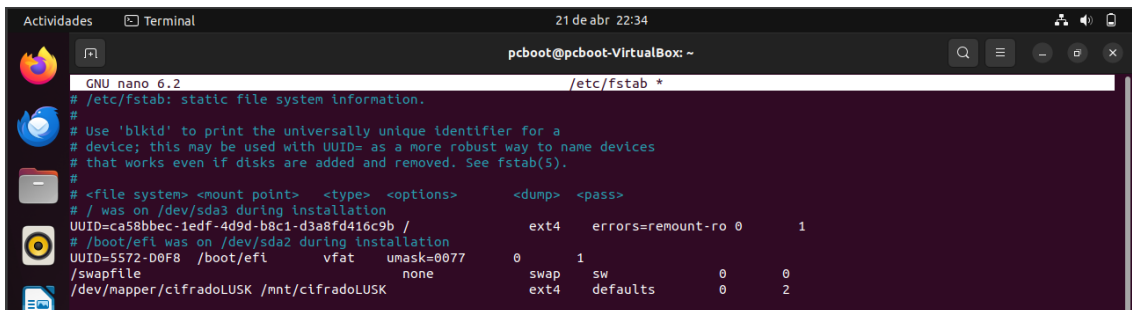
En esta pantalla, procedemos a editar el archivo /etc/crypttab utilizando el comando `sudo nano /etc/crypttab`. Luego, presionamos la tecla Enter para ejecutar el comando y abrir el archivo en el editor de texto Nano para realizar las modificaciones necesarias.



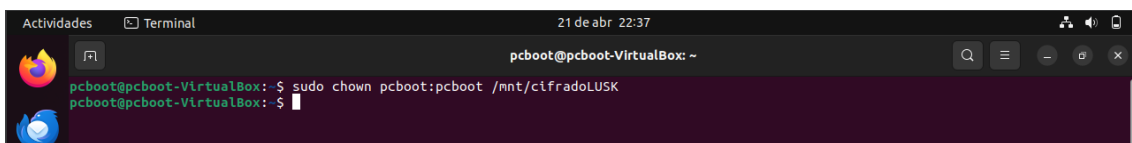
En esta pantalla, procedemos a editar el archivo crypttab. Escribimos la línea cifradoLUSK /dev/sdb1 none luks dentro del archivo y luego presionamos las teclas "Ctrl + O" para guardar los cambios y "Ctrl + X" para salir del editor Nano. De esta manera, hemos actualizado el archivo crypttab con la configuración adecuada.



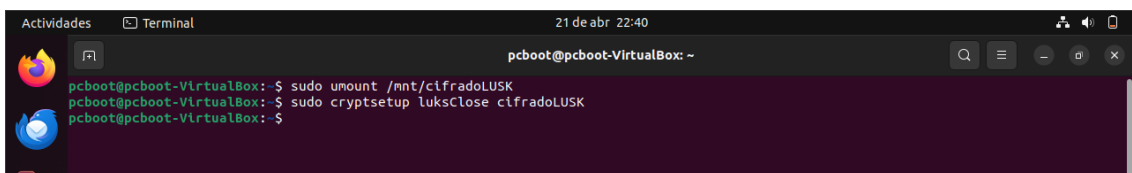
En esta pantalla, procedemos a editar el archivo /etc/fstab utilizando el comando sudo nano /etc/fstab. Luego, presionamos la tecla Enter para ejecutar el comando y abrir el archivo en el editor de texto Nano para realizar las modificaciones necesarias.



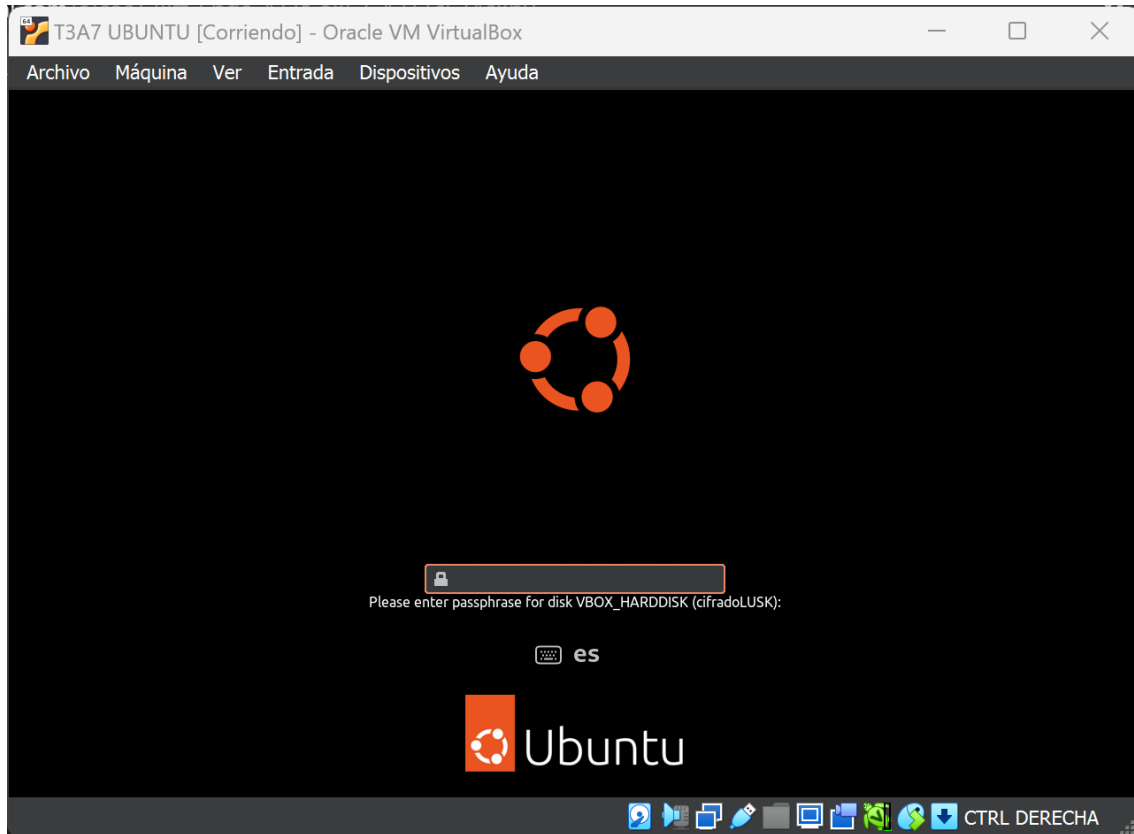
En esta pantalla, procedemos a escribir al final del archivo /etc/fstab la línea /dev/mapper/cifradoLUSK /mnt/cifradoLUSK ext4 defaults 0 2. Luego, presionamos las teclas "Ctrl + O" para guardar los cambios y "Ctrl + X" para salir del editor Nano. De esta manera, hemos añadido la configuración necesaria para montar automáticamente la unidad cifrada en el directorio /mnt/cifradoLUSK al iniciar el sistema.



En esta pantalla, para permitir que un usuario no root pueda almacenar información en la unidad cifrada, ejecutamos el comando sudo chown pcboot:pcboot /mnt/cifradoLUSK. Luego, presionamos la tecla Enter para ejecutar el comando y cambiar el propietario y grupo del directorio /mnt/cifradoLUSK al usuario pcboot. De esta manera, el usuario pcboot tendrá permisos para escribir en la unidad cifrada.

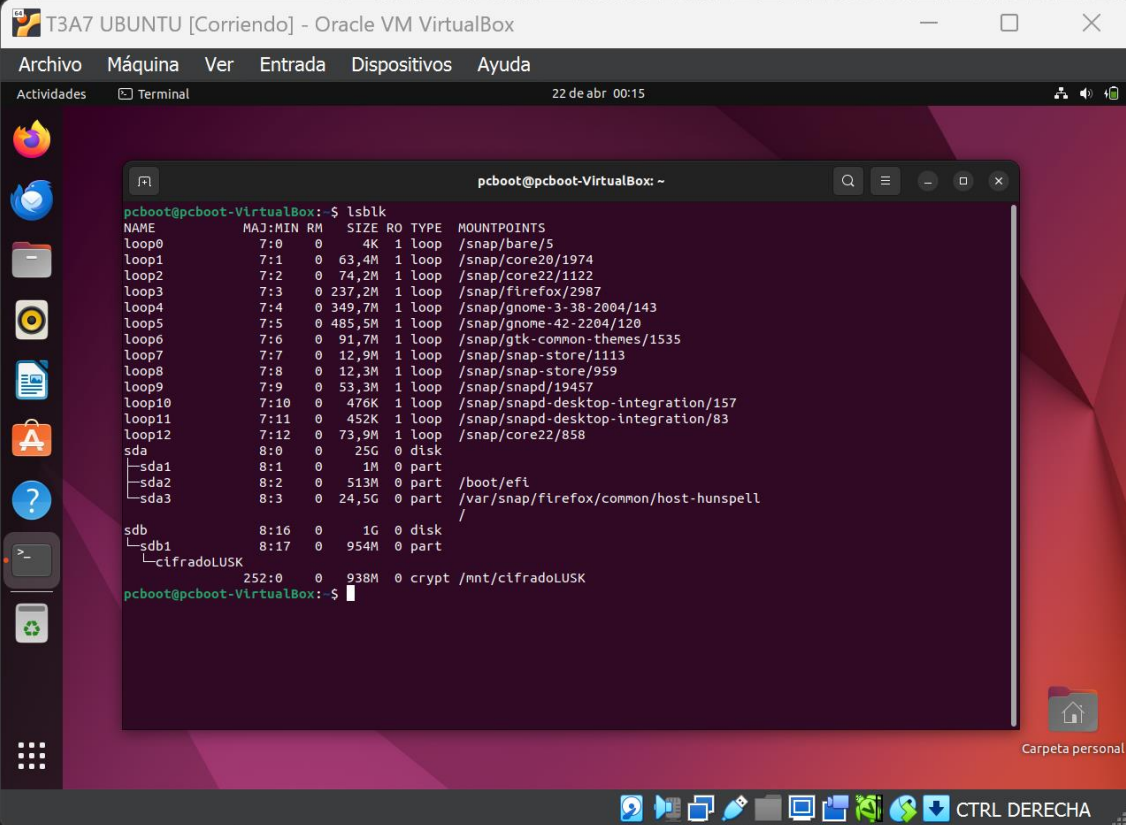


En esta pantalla, procedemos a desmontar la unidad utilizando el comando `sudo umount /mnt/cifradoLUSK` y luego bloquear la unidad con el comando `sudo cryptsetup luksClose cifradoLUSK`. Luego, presionamos la tecla Enter para ejecutar los comandos y desmontar y bloquear correctamente la unidad cifrada.



En esta pantalla, procedemos a simular lo que hace el sistema en el arranque utilizando el comando `sudo mount -a`. Esto intentará montar todos los sistemas de archivos definidos en `/etc/fstab`. Luego, al reiniciar el sistema, es posible que se te solicite la contraseña de cifrado para desbloquear la unidad cifrada y permitir que se monte automáticamente durante el arranque.

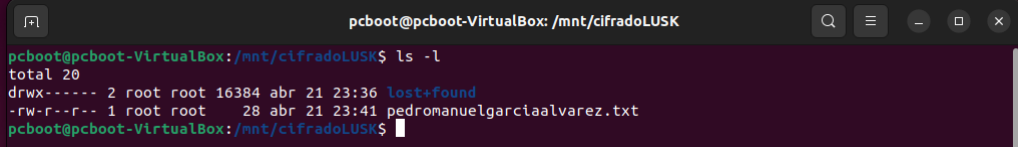




The screenshot shows a terminal window titled 'pcboot@pcboot-VirtualBox: ~' within a virtual machine environment. The terminal displays the output of the 'lsblk' command, which lists various storage devices and their mount points. The output is as follows:

```
pcboot@pcboot-VirtualBox: ~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINTS
loop0       7:0    0     4K  1 loop  /snap/bare/5
loop1       7:1    0   63,4M  1 loop  /snap/core20/1974
loop2       7:2    0   74,2M  1 loop  /snap/core22/1122
loop3       7:3    0  237,2M  1 loop  /snap/firefox/2987
loop4       7:4    0   349,7M  1 loop  /snap/gnome-3-38-2004/143
loop5       7:5    0   485,5M  1 loop  /snap/gnome-42-2204/120
loop6       7:6    0   91,7M  1 loop  /snap/gtk-common-themes/1535
loop7       7:7    0   12,9M  1 loop  /snap/snap-store/1113
loop8       7:8    0   12,3M  1 loop  /snap/snap-store/959
loop9       7:9    0   53,3M  1 loop  /snap/snapd/19457
loop10      7:10   0   476K  1 loop  /snap/snapd-desktop-integration/157
loop11      7:11   0   452K  1 loop  /snap/snapd-desktop-integration/83
loop12      7:12   0   73,9M  1 loop  /snap/core22/858
sda         8:0    0    25G  0 disk
├─sda1      8:1    0     1M  0 part
├─sda2      8:2    0   513M  0 part  /boot/efi
└─sda3      8:3    0   24,5G  0 part  /var/snap/firefox/common/host-hunspell
sdb         8:16   0     1G  0 disk
├─sdb1      8:17   0   954M  0 part
└─cifradoLUSK 252:0    0   938M  0 crypt /mnt/cifradoLUSK
pcboot@pcboot-VirtualBox: ~$
```

En esta pantalla, procedemos a ejecutar el comando `lsblk` para verificar que el volumen cifradoLUSK esté montado. Luego, presionamos la tecla Enter para ejecutar el comando y observar la lista de dispositivos de bloque, incluyendo las unidades montadas en el sistema.



The screenshot shows a terminal window titled 'pcboot@pcboot-VirtualBox: /mnt/cifradoLUSK'. The terminal displays the output of the 'ls -l' command, showing the contents of the mounted LUKS volume. The output is as follows:

```
pcboot@pcboot-VirtualBox: /mnt/cifradoLUSK$ ls -l
total 20
drwx----- 2 root root 16384 abr 21 23:36 lost+found
-rw-r--r-- 1 root root   28 abr 21 23:41 pedromanuelgarciaalvarez.txt
pcboot@pcboot-VirtualBox: /mnt/cifradoLUSK$
```

En esta pantalla, procedemos a ejecutar el comando `ls -l` dentro del directorio `/mnt/cifradoLUSK` para ver el contenido de la unidad cifrada. Luego, observamos el archivo `pedromanuelgarciaalvarez.txt` dentro de la unidad cifrada.

# Índice Alfabético

---

## A

actualización	10
aleatoria	12
Algoritmo	15
almacenamiento	2

---

## B

básica	14
bloques	2

---

## C

cifrarlo	2
comandos	2, 23
configuración	6, 15, 22
Copia	1, 15
creación	3, 4
Ctrl	22

---

## D

derivación	14, 15
detalles	14, 17
directorio	20, 21, 22, 24
Disk	4

---

## E

editor	21, 22
ejecución	6
encriptarlo	3
Enter	10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24
Escribimos	22
específica	15
específicos	14
explicación	2
extraíbles	2

---

## F

formato	2, 14
---------	-------

---

## H

Huella	15
--------	----

---

## I

Image	4
importantes	3
inaccesibles	2
información	1, 2, 12, 14, 15, 17, 18, 19, 22
instrucciones	2
interfaz	3

---

## K

KDF	14
-----	----

---

## L

Linux	2
lista	2, 19, 24

---

## M

máquina	5, 6, 18
MiB	7
modificaciones	21, 22
Muestra	2

---

## N

Nano	21, 22
necesarias	21, 22
nombres	2

---

## O

opción	4, 7, 8
operación	15

---

## P

pantalla	4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24
----------	---

Parámetros	14
partición	2, 7, 14, 15, 17
pendrives	2
permisos	22
precaución	2
primaria	7
propietario	22
puedes	2

---

## S

sector	7
seguridad	1, 3, 15
selección	5

---

## T

Tamaño	14
tecla	10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24

teclas	22
--------	----

---

## U

ubicación	6
Ubuntu	10
única	2, 14
únicos	2
unidad	1, 7, 9, 12, 13, 18, 19, 20, 21, 22, 23, 24
unidades	19, 24
usuario	22
utilidades	2
UUID	2, 14

---

## V

VDI	4
VirtualBox	3, 4