



Sistemas de arranque y Particiones

Objetivo: Este documento está diseñado para que los alumnos de la Facultad de Ciencias de la UNAM puedan tener un concepto un poco más claro de cómo es el funcionamiento de los Sistemas de arranque que tiene una computadora pues el que se encarga de **inicializar** el hardware, **cargar** el Sistema Operativo que tiene la computadora para después **ejecutarlo**.

Otro tema importante que se menciona en este documento es la **elaboración de particiones** en el disco duro de una manera segura.

Índice

1. Sistema de arranque de Linux	2
1.1. ¿Cómo es el arranque en Linux?	2
2. Sistema de arranque de Windows	2
2.1. ¿Cómo es el arranque en Windows?	2
3. BIOS	3
4. UEFI	4
5. Diferencias entre BIOS y UEFI	5
6. GRUB	6
6.1. ¿Qué es el GRUB?	6
6.2. ¿Cómo funciona el GRUB?	6
7. Particiones	7
7.1. ¿Qué es una partición?	7
7.2. Tipos de discos duros.	7
8. Particiones de disco duro	8
8.1. ¿Cómo particionar un disco en Windows?	9
8.2. ¿Cómo particionar un disco en Linux?	14
8.3. ¿Cómo hacer una partición de algún tipo en específico en Linux?	19
8.4. ¿Cómo formatear una partición en Linux?	22



1. Sistema de arranque de Linux

1.1. ¿Cómo es el arranque en Linux?

El proceso de arranque del Sistema Linux considera los siguientes pasos:

1. Al encender la computadora el CPU empieza a ejecutar **código máquina** alojado en una dirección de memoria ya antes predefinida.
2. Dicho código máquina se le conoce como **BIOS** que significa 'Basic Input Output System' o 'Sistema Básico de Entrada y Salida' en español.
3. La BIOS busca un programa llamado **GRUB** el cual es un gestor de arranque.
Esto se conoce como **bootstrapping** pues la BIOS no se puede ejecutar sin ejecutar el GRUB es decir, un programa no puede ejecutarse sin haber ejecutado a otro.
4. El GRUB instala un gestor de arranque en el registro MBR (Registro Maestro de Arranque o Sector Cero) el cual hace referencia al sector del disco duro que contiene al gestor de arranque.
5. Una vez cargado el GRUB lo que sigue será buscar el **Kernel** que es el núcleo del Sistema Linux, lo colocará en la memoria correspondiente para después indicarle al CPU que lo ejecute.
6. Al ejecutar lo que se hace es iniciar el Sistema Operativo, transferimos instrucciones especiales a los kernels durante el arranque o conocer parámetros del sistema (como la memoria RAM disponible) antes de arrancar.

2. Sistema de arranque de Windows

2.1. ¿Cómo es el arranque en Windows?

A diferencia de Linux, el arranque de Windows utiliza una herramienta llamada BCD store (Boot Configuration Data) un registro que contiene los datos suficientes para poder hacer el arranque del Sistema.

Los pasos de arranque son similares a los de Linux:

1. La BIOS carga el registro MBR (Registro Maestro de Arranque) el cual se encarga de verificar las particiones del equipo esto para verificar en qué partición está alojado el Sistema Operativo Windows.
2. Se localiza el **bootmgr** mediante el sector de arranque y se ejecuta.
El bootmgr es el que se encarga de que se ejecute el **boot loader** quien se encarga del arranque pues carga en la memoria RAM del equipo los Sistemas Operativos que se tengan instalados en un equipo.
3. Al ejecutarse busca el directorio \boot el cual contiene los programas necesarios para iniciar el Sistema.
4. Por último se ejecuta el BCD en el que se selecciona el sistema que se desea arrancar comenzando la carga del núcleo (kernel) del Sistema.



3. BIOS

Como ya vimos, el BIOS es un sistema de arranque que se utiliza para iniciar una computadora. Sin embargo, hay veces que es necesario entrar a dicho programa y poder solucionar algún problema que se tenga.

Para entrar a la BIOS existen diferentes maneras, esto depende de la computadora, la marca o características. Para saber como acceder lo que podemos hacer es:

- Apagar o reiniciar la computadora.
- Cuando encendamos de nuevo la computadora o esta se reinicie, lo que haremos será presionar la tecla **ESC** o en su defecto **F1**.
- Nos tiene que aparecer una pantalla parecida a lo siguiente:

```
Startup Menu

F1   System Information
F2   System Diagnostics
F9   Boot Device Options
F10  BIOS Setup
F11  System Recovery

Enter - Continue Startup

For more information, please visit:
www.hp.com/go/techcenter/startup
```

Como podemos observar, nos aparece un menú el cual nos dice ciertas utilidades que tienen algunas teclas al ser presionadas justo cuando estamos encendiendo la computadora o esta se reinicia.

En este caso nos dice la siguiente información:

- Con la tecla **F1** podemos encontrar información del Sistema.
- Con la tecla **F2** podemos ver el diagnóstico del Sistema.
- Con la tecla **F9** podemos acceder a las opciones del boot es decir, aquí podemos cambiar el modo de arranque de nuestra computadora si tenemos dos Sistemas Operativos. Aquí se decide que Sistema se arranca principalmente.
- Con la tecla **F10** podemos acceder a la BIOS de la computadora.
- Con la tecla **F11** podemos ver las opciones de recuperación del Sistema.



Entonces con la información anterior sabemos que con **F10** entramos a la BIOS de esta computadora. Al entrar se desplegará una pantalla similar a la siguiente:



4. UEFI

Bios UEFI o Unified Extensible Firmware Interface por sus siglas en inglés es una versión más actualizada que la BIOS por lo que también funciona como sistema de arranque de una computadora. Este sistema de arranque está disponible por lo general en sistemas Windows.

Fue creado para ser más intuitivo y fácil de usar pues hace uso de una interfaz gráfica para mayor comodidad al usuario.

Varias de las ventajas de UEFI son:

- Es compatible con equipos de 64 bits.
De hecho Bios UEFI no se instalará en equipos de 32 bits.
- Tiene tiempos de arranque más rápidos.
- Tiene la capacidad de utilizar Secure Boot.
- Utiliza una interfaz gráfica haciéndolo más fácil de usar.
- Tiene la capacidad de iniciar el arranque con discos duros con capacidad de más de 2TB.
- Acepta particiones GPT.

Para ingresar al UEFI de Windows debemos realizar los siguientes pasos:
Esto es valido pra versiones mayores o iguales de WIndows 8.1

1. Entramos al menú de Configuración de Windows.
2. Damos click en la opción de [Cambiar configuración de PC](#).
3. Damos click en la opción de [Actualizar y recuperar](#).
4. Debajo del ítem que dice [Inicio Avanzado](#) damos click en la opción de [Reiniciar ahora](#).
5. Finalmente entramos al UEFI de Windows.



La siguiente es una representación del UEFI en Windows:



En dado caso de que no podamos acceder al UEFI de Windows con los pasos anteriores podemos hacerlo reiniciando la computadora y hacer los pasos que hicimos con Linux.

5. Diferencias entre BIOS y UEFI

Mencionaremos las principales diferencias mediante el uso de la siguiente tabla:

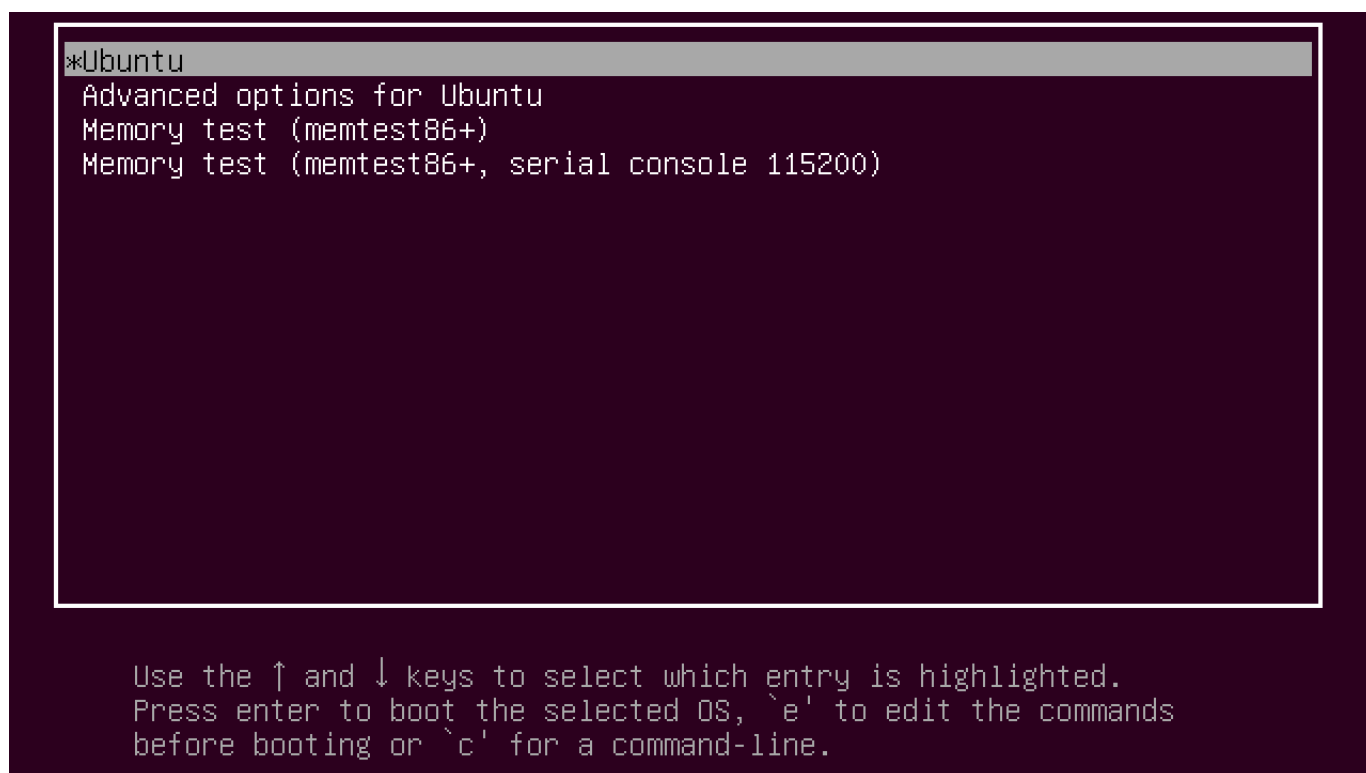
BIOS	UEFI
Tiene un diseño en el cual solo nos podemos mover haciendo uso de las teclas de dirección, de función y la tecla Enter.	Tiene una interfaz la cuál es fácil de usar para el usuario y aquí si podemos hacer uso del mouse.
Su actualización se hace a través de una USB booteable.	Puede conectarse a una red inalámbrica para hacer actualizaciones por su cuenta.
Puede ser ejecutada con equipos de 16 bits.	Solo se ejecuta en equipos de 32 o 64 bits.
	El arranque es más rápido que el de la BIOS.
	Evita que se inicie un Sistema Operativo que no esté autenticado, da más seguridad.
	Es independiente de cualquier Sistema Operativo.



6. GRUB

6.1. ¿Qué es el GRUB?

GRUB por sus siglas en inglés significa Grand Unified Bootloader, es un gestor de arranque. Cuando tenemos instalados dos Sistemas Operativos en nuestra computadora lo primero que se inicia es el GRUB pues este lo que hace es cargar de ambos sistemas y nos permite elegir en cuál iniciar.



6.2. ¿Cómo funciona el GRUB?

Su funcionamiento se define mediante los siguientes pasos:

1. El MBR busca que partición es la que vamos a activar es decir, que Sistema Operativo vamos a iniciar.
2. Al encontrar la partición lo que hace MBR es cargar el sistema desde su gestor de arranque en su memoria y da acceso a la misma para que se encargue de ello.
3. Como MBR no tiene la capacidad de memoria alta lo que pasa a continuación es que el GRUB sustituye a MBR por defecto y usa su propio código para poder hacer el arranque del sistema.

Nota: Más adelante se explicará con un poco más de detalle la memoria MBR.



7. Particiones

7.1. ¿Qué es una partición?

Sabemos que la computadora contiene un disco el cual es un almacenamiento de memoria y justamente ahí es donde se aloja nuestro sistema operativo así como archivos, aplicaciones, etc. En dicho disco podemos hacer **particiones** es decir, podemos asignar un espacio de memoria para almacenar diferentes archivos o sistemas.

Entonces podemos ver a una partición como el nombre que se le da a cada división que se le da a un disco duro de una computadora en donde se almacenan los datos, dicha división sirve para que de forma intuitiva se puedan tener varios discos duros en uno solo es decir, cada división trabaja como si fuera un solo disco duro independiente.

Otra forma de verlo es como tener varios discos duros en un solo disco, los cuales trabajan de manera independiente pues cada uno tiene su propio sistema de archivos y trabajan de manera diferente.

Existen tres tipos de particiones de discos:

- **Partición primaria.**

Son las que detecta la computadora al momento de hacer el arranque por lo que en este tipo de particiones es donde se instalan los Sistemas Operativos. El máximo número de particiones en un disco duro es de 4.

- **Partición extendida o secundaria.**

Aquí podemos tener mas de 4 particiones en el disco sin embargo, no se puede alojar ningun Sistema Operativo en este tipo de particiones por lo que solo podremos almacenar datos en ella.

- **Partición lógica.**

Estas particiones se hacen dentro de las particiones extendidas, de esta manera solo basta con asignarle un tamaño y un tipo de sistema de archivos por ejemplo FAT32, NTFS, ext2, etc.

7.2. Tipos de discos duros.

Los ordenadores pueden tener un disco duro SSD (de estado sólido), un disco VHD (discos virtuales) o discos HDD (Hard Drive Disk), entre otros. Sin embargo, solo mencionaremos estos pues son los más comunes.

- **SSD:** Es de las mejores opciones para elegir una computadora al momento de comprarla o armarla.

A continuación mencionaremos algunas de las ventajas que tiene este tipo de disco duro:

- Su velocidad es rápida pues el tiempo para obtener una respuesta siempre es el mismo.
- Tiene mayor resistencia a los golpes o vibraciones debido a que su estructura no contiene componentes que se muevan (es sólido) así que es más duradero.
- No hace ruido al trabajar por lo mismo de que su estructura no tiene piezas movibles.
- No consume mucha energía para poder funcionar, además debido a eso lo hace perfecto para ordenadores portátiles pues no genera tanto calor por lo que la computadora tardará mucho en calentarse lo que a su vez también es menor desgaste en el disco y lo hace más duradero.

Por otro lado, algunas de sus desventajas son las siguientes:

- Su precio es elevado.
- Tienen menor capacidad de memoria.

- **HDD:** Este disco a diferencia del SSD está compuesto por piezas movibles o mecánicas. Se compone por varios discos que se unen con engranajes y que giran a cierta velocidad.

Algunas de sus ventajas son las siguientes:



- Son más económicos que los discos SSD.
- Tienen mayor capacidad de almacenamiento.

Sus desventajas son las siguientes:

- Tienen menor velocidad de arranque.
 - Consumen más energía por lo que las computadoras suelen calentarse de más.
 - Debido a su estructura de discos giratorios suele hacer ruido al estar trabajando.
 - Es más susceptible a golpes o caídas pues los discos se pueden desalinearse y quedar dañados.
- **VHD:** Es un disco duro virtual el cual tiene funcionalidades parecidas a las de un disco duro ya sea SSD o HDD.

Algunas de sus ventajas:

- Tiene todas las funcionalidades de un disco duro.
- Puede alojar en él varios discos virtuales y no interfieren entre sí.
- Tienen tamaño fijo o flexible.

Sus desventajas:

- Cualquier usuario puede añadir archivos.

8. Particiones de disco duro

La partición **MBR** por sus siglas en inglés **Master Boot Record** se refiere al primer sector de memoria que tiene un disco duro.

Dicho sector de memoria solo cuenta con 512 bytes en donde almacena al **gestor de arranque primario** el cual hace uso de 446 bytes de la memoria del MBR.

Una de las desventajas de este sector de memoria es que al tener poco espacio de memoria solo es posible ser usado con un máximo de 2TB de memoria en el disco duro. Esto significa que si tenemos un disco más grande a 2TB con una partición MBR, sólo se puede usar como máximo el espacio de 2TB en él.

La partición **GPT** es introducida por UEFI. Podemos crear las particiones que queramos siempre y cuando el número de las particiones esté limitado por el Sistema Operativo.

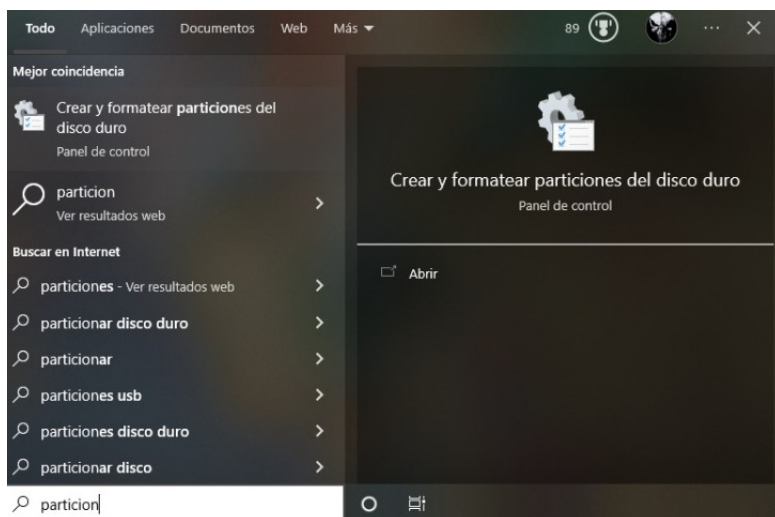
En GPT no existen las particiones lógicas o extendidas, esto debido a lo mencionado anteriormente (el número de particiones que se pueden hacer).

Una de las ventajas de GPT es la seguridad que tiene pues hace copias de seguridad de esta manera si el encabezado GPT o la tabla de particiones se daña, estas copias de seguridad serán útiles para restaurar los datos.



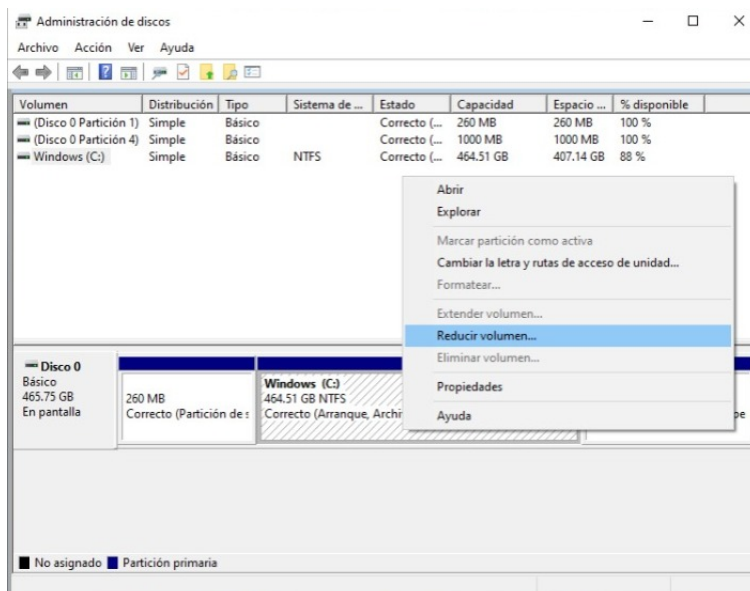
8.1. ¿Cómo particionar un disco en Windows?

Ahora bien, si lo que queremos es mantener dos Sistemas Operativos que en este caso sería Windows y alguna distribución de Linux, lo que debemos hacer es presionar la tecla de *Windows* y escribimos *Partición*.



Accedemos a la opción *Abrir* y lo siguiente será dar click derecho sobre el disco al cual le queremos hacer la partición.

En este caso, observemos que el disco con nombre *Windows* es el principal pues nunca se ha hecho una partición, además de que contiene más memoria que los demás. Entonces, damos click derecho sobre *Windows (C:)* y seleccionamos la opción de *Reducir volumen*.

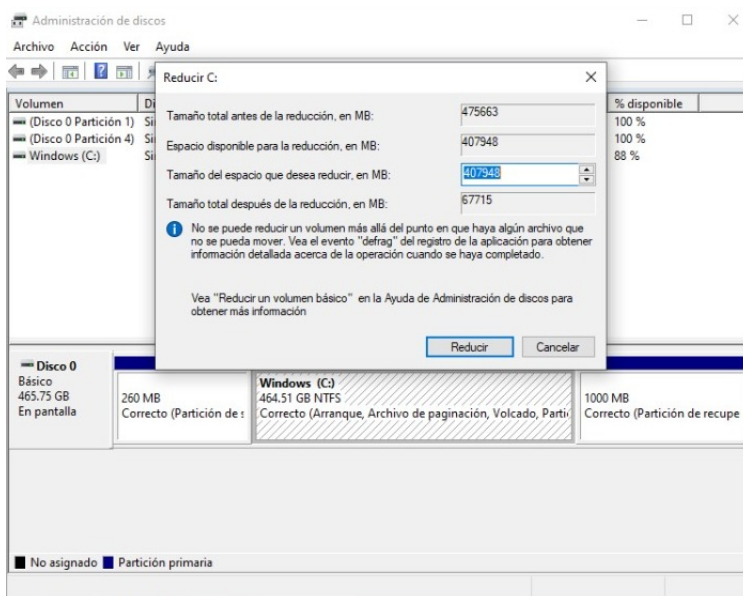


Hay que tener en cuenta que también existe la opción de *Eliminar volumen* en donde podemos borrar una partición que ya no nos sirva y así tener más memoria o inclusive podemos *Formatear* un disco para así 'limpiarlo' antes de usarlo o de hacer una partición.

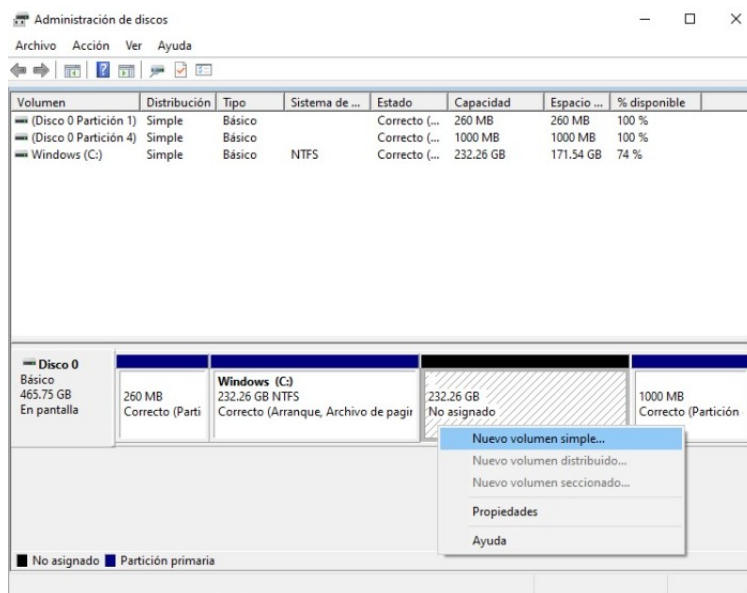
Lo siguiente es asignar el espacio de memoria en el disco que vamos a crear a partir de la partición.



Esto se hace a nivel de MB por lo que es recomendable multiplicar por 1024 la cantidad de memoria que deseamos asignar por ejemplo, si queremos asignar 200GB de memoria al nuevo disco haremos la multiplicación $200 \times 1024 = 204,800$ MB es lo que pondremos en la casilla y daremos click en [Reducir](#).



Al terminar el proceso de reducción del disco duro aparecerá en pantalla un espacio en negro (los demás son azules) con el nombre de [No Asignado](#) el cual hace referencia al espacio que tenemos disponible.

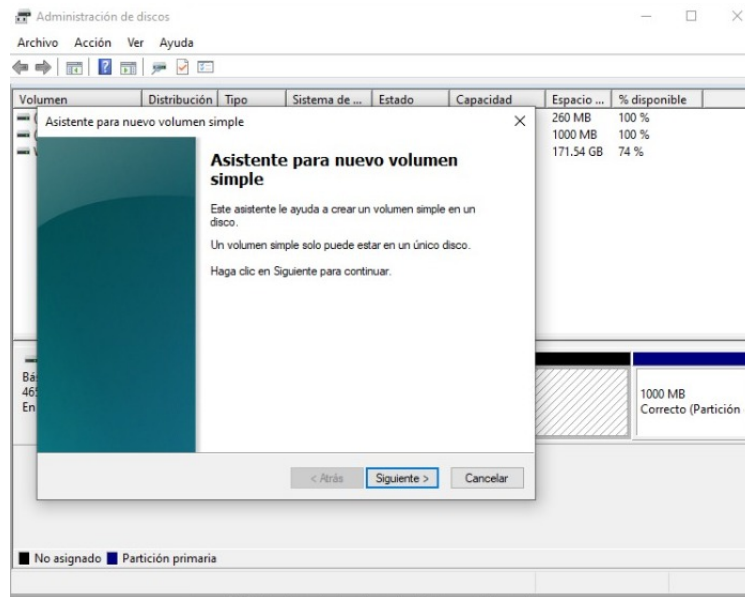


Nota: Puedes dejar así la partición con la etiqueta [No asignado](#). La diferencia de seguir los siguientes pasos es hacer una partición NTFS.
Para más información, puedes consultar el siguiente [link](#).



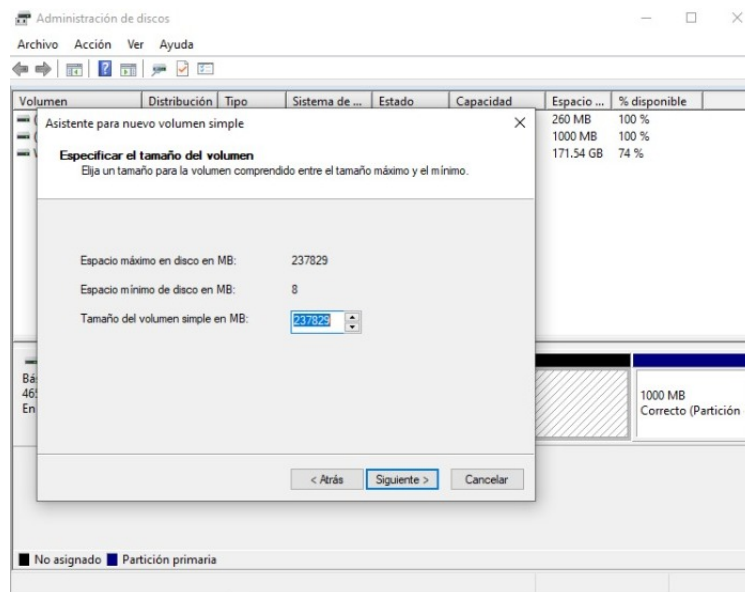
Damos click derecho sobre el espacio no asignado y elegimos la opción de *Nuevo volumen simple* para crear la nueva partición.

Lo siguiente será seguir las instrucciones que nos dirá el Asistente para el nuevo volumen. Damos click en *Siguiente*.



Nuevamente debemos poner el tamaño de la nueva partición. Por defecto nos muestra el tamaño que ya le hemos asignado con anterioridad, pero lo podemos editar si así lo deseamos.

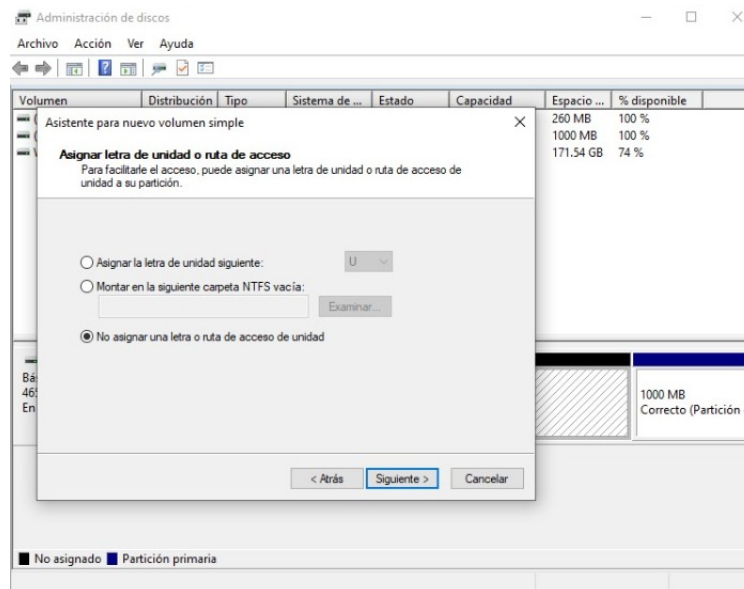
Al finalizar, damos click en *Siguiente*.



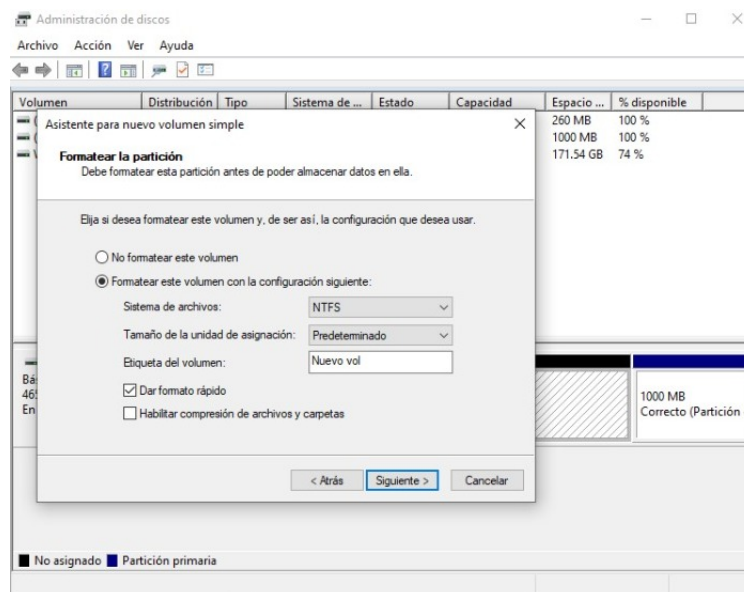
Algo importante que hay que mencionar en el siguiente paso es que Windows no sabe que la partición la queremos para instalar otro Sistema Operativo.



Así mismo, el siguiente paso es asignarle un nombre a la nueva partición que acabamos de crear y lo recomendable es poner la opción de *No asignar una letra o ruta de acceso de unidad* de esta manera se va a crear la partición pero Windows no la utilizará como unidad de almacenamiento.

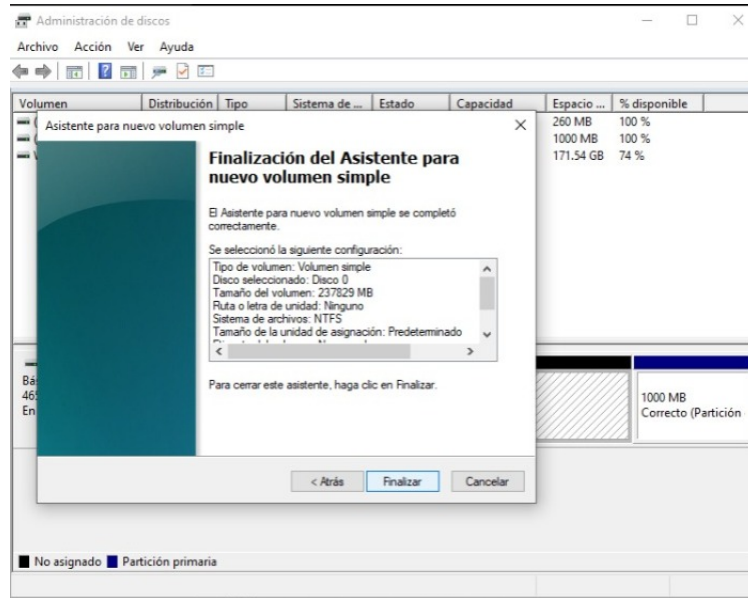


Elegimos si queremos formatear la partición lo cuál es recomendable hacer. Damos click en *Siguiente*.

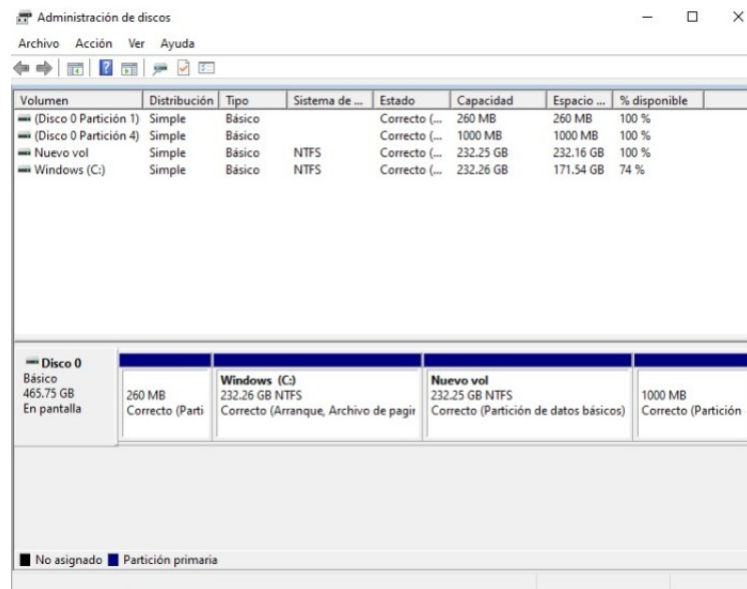




Termina el proceso del asistente. Damos click en [Finalizar](#).



Finalmente, se ha creado una nueva partición en el disco duro.





8.2. ¿Cómo particionar un disco en Linux?

En algunas ocasiones puede pasar que requerimos otro espacio de memoria ya sea para la instalación de un tercer Sistema Operativo o para guardar archivos más pesados y así no consumir tantos recursos en una partición.

Entonces, para poder hacer una partición desde Linux podemos usar [GParted](#) que es una herramienta la cual nos ayuda (de manera gráfica) hacer particiones de un disco duro.

Para su instalación, abrimos la terminal y ejecutamos los siguientes comandos:

```
$ sudo apt update
$ sudo apt install gparted
$ sudo apt upgrade
```

Después de la instalación podemos ir al buscador de aplicaciones y buscar GParted y abrirlo o bien, podemos escribir en la consola `gparted` y automáticamente se abrirá la aplicación.

Ahora, imaginemos que ya tenemos la partición de Windows y de Linux, las particiones se verán de la siguiente manera:

Partition	Name	File System	Size	Used	Unused	Flags
/dev/sda1	EFI system partition	fat32	100.00 MiB	100.00 MiB	1.00 KiB	boot, esp
/dev/sda2	Microsoft reserved partition	unknown	16.00 MiB	---	---	msftres
/dev/sda3	Basic data partition	ntfs	46.45 GiB	28.64 GiB	17.81 GiB	msftdata
unallocated		unallocated	3.54 GiB	---	---	
/dev/sda4	Basic data partition	ext4	98.43 GiB	7.76 GiB	90.67 GiB	msftdata
unallocated		unallocated	1.25 MiB	---	---	
/dev/sda5		ntfs	522.00 MiB	436.03 MiB	85.97 MiB	hidden, diag
unallocated		unallocated	1.84 MiB	---	---	

La información relevante es la siguiente:

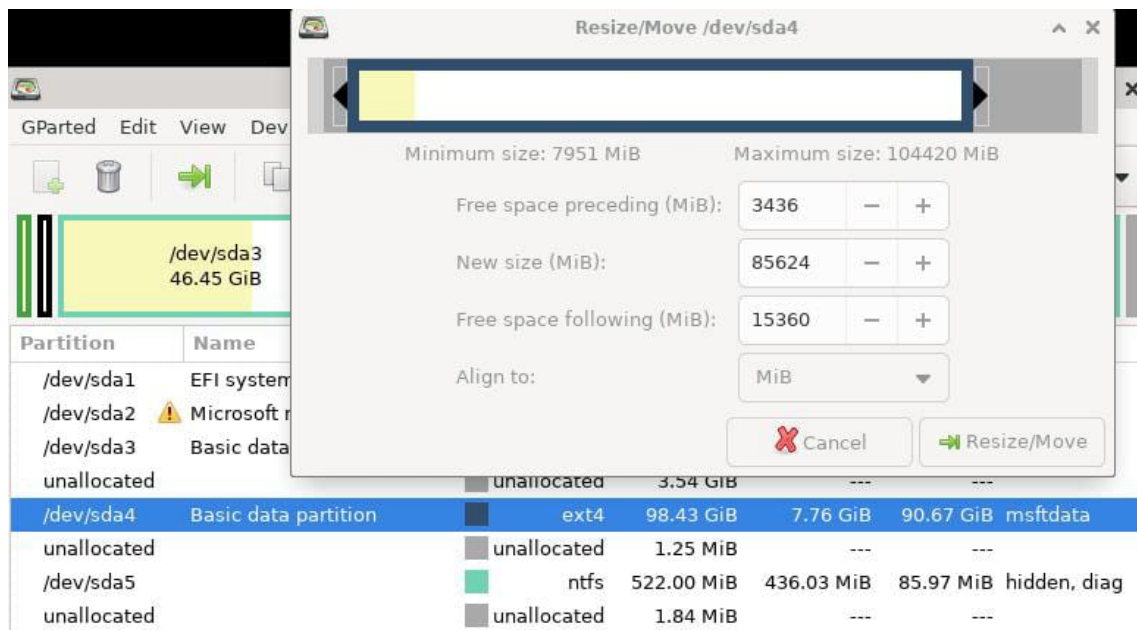
- `/dev/sda1`: Corresponde al espacio en memoria del EFI.
- `/dev/sda2`: Corresponde al espacio en memoria de reserva que utiliza Windows.
- `/dev/sda3`: Es el espacio que ocupa el Sistema Operativo Windows.
- `/dev/sda4`: Es el espacio que ocupa el Sistema Operativo Linux.
- `unallocated`: Es un espacio de memoria que no ha sido asignado.

Nota: Antes de hacer una partición hay que considerar el espacio de memoria para evitar errores o inconsistencias es decir, hay que ver que disco nos conviene reducir su tamaño.



En este caso, nos conviene hacer la partición tomando espacio en memoria del disco `/dev/sda4` pues tiene un tamaño de 98.43GiB.

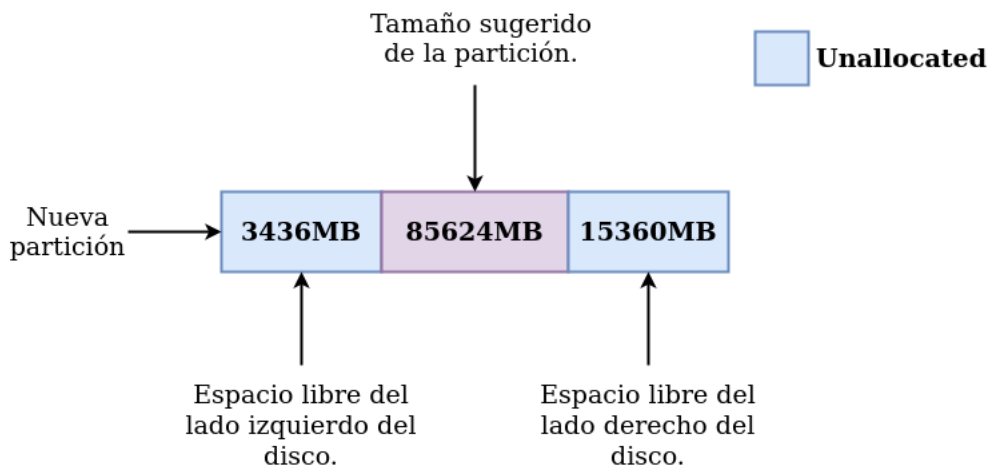
Para hacer la partición, nos colocamos en el disco a reducir y damos click derecho. Seleccionamos la opción **Resize/Move**, nos tiene que aparecer una pantalla en donde vamos a poder mover o recorrer la partición (sólo si hay espacio disponible) y asignarle el espacio que se le va a quitar.



GParted nos da valores por default. Notemos las siguientes observaciones:

- **Free space preceding:** Es el espacio que tenemos libre del lado izquierdo del disco `/dev/sda4`. Este espacio corresponde a la etiqueta `unallocated` de tamaño 3.54GB
- **New size:** Es el espacio en MB (mega byte) que GParted sugiere asignar al nuevo espacio en memoria.
- **Free space following:** Es el espacio libre que tenemos del lado derecho de la nueva partición que vamos a crear. Este espacio corresponde a la suma del espacio libre del tamaño máximo que GParted nos recomienda que tenga el disco con el espacio libre `unallocated` de espacio 1.25MB

De manera gráfica, la información anterior es como el siguiente diagrama:





Nota: Si dejamos los valores predeterminados que nos ofrece GParted puede que tengamos espacio de memoria sin asignar a la derecha o a la izquierda del nuevo disco y es desperdiciar recursos. Por lo tanto, es importante no dejar espacios de memoria y fijarnos en el tamaño que le asignamos a la partición así como la memoria libre por la izquierda y derecha del disco.

Asignamos el nuevo espacio. En este caso le asignaremos 15GB.

Para esto, recordemos que 1GB contiene 1024MB por lo que, si queremos asignar 15GB deberemos hacer la conversión de GB a MB es decir, $15\text{GB} \times 1024\text{MB} = 15360\text{MB}$.

Para no dejar espacio libre de memoria sin asignar del lado izquierdo de nuestra nueva partición y tomar la memoria libre que tenemos del lado derecho del disco, los valores serian los siguientes:

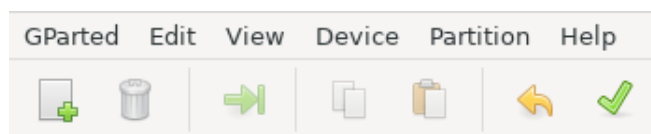
```
Free space preceding (MiB): 0
New size (MiB):             15360
Free space following (MiB): 0
                           Resize/Move
```

Damos click en *Resize/Move* y verificamos si se hizo de forma correcta la particion:

/dev/sda4	Basic data partition	ext4	83.62 GiB	7.76 GiB	75.85 GiB	msftdata
unallocated	unallocated	15.00 GiB	---	---	---	---

Observemos que efectivamente se hizo bien la partición además, tomamos el espacio libre que quedaba del lado derecho del disco /dev/sda4.

En dado caso de que no se haya hecho bien la partición porque el espacio fue el incorrecto o quedo espacio libre sin asignar por otro lado del disco, podemos revertir la acción usando las opciones del lado superior derecho del panel de GParted. En este caso, damos click a la flecha de color amarillo y revertimos la acción para corregir los errores.



Caso contrario, damos click en la flecha verde para seguir con el proceso.

GParted nos preguntará de nuevo si los datos son correctos mostrándonos la siguiente pantalla:

Create new Partition

Minimum size: 1 MiB Maximum size: 15360 MiB

Free space preceding (MiB): 0 Create as: Primary Partition

New size (MiB): 15360 Partition name: Prueba

Free space following (MiB): 0 File system: ext4

Align to: MiB Label:

Cancel Add



Si lo deseamos podemos cambiar el tamaño de la partición así como elegir si dejar espacio libre y sin asignar de ambos lados del disco. En este caso NO lo hacemos.

Podemos asignarle un nombre a nuestra nueva partición, en nuestro caso le ponemos 'Prueba'.

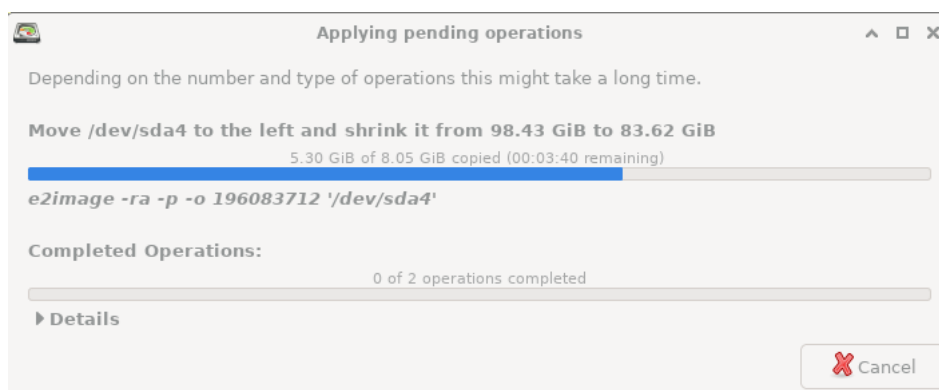
Si lo deseamos, podemos cambiar el tipo de partición, en este caso esta asignada como `ext4`, esto porque el disco `/dev/sda4` es de este tipo sin embargo, podemos cambiarlo a formato `ntfs` por ejemplo, entre otras opciones que nos ofrece GParted.

Finalmente, damos click en `+Add`.

<code>/dev/sda4</code>	Basic data partition		ext4	83.62 GiB	7.76 GiB	75.85 GiB	msftdata
New Partition #1	Prueba		ext4	15.00 GiB	---	---	

Seguido de esto, GParted se encargará de asignarle el número de partición correspondiente es decir, convertir esa partición en formato `/dev/sdaN` donde `N` es el número de partición.

Esperamos hasta que termine el proceso.



Finalmente, podemos verificar que la partición quedo de manera correcta pues no dejamos espacios de memoria sin asignar por la derecha o izquierda de la partición que creamos además, de asignarle el espacio deseado:

Partition	Name	File System	Size	Used	Unused	Flags
<code>/dev/sda1</code>	EFI system partition	fat32	100.00 MiB	100.00 MiB	1.00 KiB	boot, esp
<code>/dev/sda2</code>	⚠ Microsoft reserved partition	unknown	16.00 MiB	---	---	msftres
<code>/dev/sda3</code>	Basic data partition	ntfs	46.45 GiB	28.64 GiB	17.81 GiB	msftdata
unallocated		unallocated	3.36 GiB	---	---	
<code>/dev/sda4</code>	Basic data partition	ext4	83.62 GiB	7.73 GiB	75.88 GiB	msftdata
<code>/dev/sda6</code>	Prueba	ext4	15.00 GiB	118.86 MiB	14.88 GiB	
<code>/dev/sda5</code>		ntfs	522.00 MiB	436.03 MiB	85.97 MiB	hidden, diag
unallocated		unallocated	1.84 MiB	---	---	



Otra manera de verificar como quedo nuestro disco duro con esta última partición es utilizando el comando [fdisk](#). Abrimos la consola en Linux con Ctrl+Alt+T y escribimos el siguiente comando:

```
$ sudo fdisk -l
```

Ingresamos nuestra contraseña y nos mostrara información de nuestro disco duro, en este caso nos imprime en consola los siguientes datos:

Device	Start	End	Sectors	Size	Type
/dev/sda1	2048	206847	204800	100M	EFI System
/dev/sda2	206848	239615	32768	16M	Microsoft reserved
/dev/sda3	239616	97656250	97416635	46.5G	Microsoft basic data
/dev/sda4	104693760	280051711	175357952	83.6G	Microsoft basic data
/dev/sda5	311508992	312578047	1069056	522M	Windows recovery environment
/dev/sda6	280051712	311508991	31457280	15G	Linux filesystem

Observemos que [/dev/sda6](#) es nuestra nueva partición con tamaño 15GB y su tipo es [Linux filesystem](#) esto es porque, esa partición se hizo de una partición de Linux.



8.3. ¿Cómo hacer una partición de algún tipo en específico en Linux?

En algunas ocasiones podemos necesitar más espacio de memoria en el EFI por ejemplo, o necesitar una partición de tipo MBR para la instalación de algún sistema que solo sea compatible con ese tipo de partición de disco.

Supongamos que necesitamos una nueva partición de tipo [EFI](#) porque hubo algún problema de espacio en el efi y necesitamos asignarle más memoria.

Checamos el disco de nuestra computadora:

```
$ sudo fdisk -l /dev/sda
```

Observamos el contenido:

Device	Start	End	Sectors	Size	Type
/dev/sda1	2048	4095	2048	1M	BIOS boot
/dev/sda2	4096	2101247	2097152	1G	Linux filesystem
/dev/sda3	2101248	31455231	29353984	14G	Linux filesystem

```
[liveuser@localhost-live ~]$
```

Ahora, vamos a hacer una nueva partición del disco [/dev/sda3](#) de tipo [EFI](#) para almacenar nuevos datos. En la terminal, entramos como [superusuario](#) y volvemos a checar el disco:

```
$ sudo su -  
$ fdisk -l
```

Podemos dar la opción [m](#) para ver el listado de opciones que nos dan para las particiones que queramos hacer:

```
[liveuser@localhost-live ~]$ sudo su -  
[root@localhost-live ~]# fdisk /dev/sda  
  
Welcome to fdisk (util-linux 2.38.1).  
Changes will remain in memory only, until you decide to write them.  
Be careful before using the write command.  
  
Command (m for help):
```

En este caso damos la opción [n](#) la cuál nos dice [add a new partition](#) y damos Enter.

Nota: Si nos sale un mensaje que nos diga [No enough free sectors available](#) lo que podemos hacer es darle otro volumen al disco.

Esto lo podemos hacer usando [GParted](#) o desde la terminal con [parted](#).

Al indicarle que queremos hacer una nueva partición, lo siguiente será especificarle los sectores de inicio y final de dicha partición.

En este caso, la nueva partición quedará enseguida de la partición donde tenemos a Linux pues es inconveniente dejar espacios vacíos a menos que sepamos que se pueden ocupar en un futuro.



En el primer sector damos Enter, después damos la opción **p** que es el último tipo de partición y finalmente de nuevo damos Enter.

Nos tiene que aparecer un mensaje que nos diga el tipo de partición y su tamaño.

```
Command (m for help): n
Partition number (4-128, default 4):
First sector (31250432-31457246, default 31250432):
Last sector, +/-sectors or +/-size[K,M,G,T,P] (31250432-31457246, default 31455231): p
Last sector, +/-sectors or +/-size[K,M,G,T,P] (31250432-31457246, default 31455231):
Created a new partition 4 of type 'Linux filesystem' and of size 100 MiB.
Command (m for help):
```

En este caso, el tamaño de la nueva partición es de 100MB que es el espacio ideal para una partición EFI.

Verificamos que la nueva partición aparezca en el disco duro como **/dev/sda4** pues es nueva partición, poniendo como opción la letra **p** que nos dice **print the partition table**.

```
Command (m for help): p
Disk /dev/sda: 15 GiB, 16106127360 bytes, 31457280 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 7D80BEEF-E464-483A-928D-1CBEE8D37DFE

Device            Start      End          Sectors      Size Type
/dev/sda1          2048       4095         2048         1M BIOS boot
/dev/sda2          4096     2101247    2097152         1G Linux filesystem
/dev/sda3     2101248    31250431    29149184     13.9G Linux filesystem
/dev/sda4     31250432    31455231     204800       100M Linux filesystem
Command (m for help):
```

Observemos que, la nueva partición está como tipo **Linux filesystem**.

Lo que sigue es cambiar el tipo de partición, para esto damos la opción **t** que nos ofrece **change a partition type**. Después nos pedirá cuál número de partición es la que queremos cambiar, en este caso es la cuatro, ingresamos **4** y si queremos ver la lista de los tipos de particiones que nos ofrece **fdisk** ponemos **l** en la terminal.



Las opciones que nos da [fdisk](#) para cambiar el tipo de una partición son las siguientes:

```
Command (m for help): t
Partition number (1-4, default 4): 4
Partition type or alias (type L to list all): l
 1 EFI System                                C12A7328-F81F-11D2-BA4B-00A0C93EC93B
 2 MBR partition scheme                     024DEE41-33E7-11D3-9D69-0008C781F39F
 3 Intel Fast Flash                         D3BFE2DE-3DAF-11DF-BA40-E3A556D89593
 4 BIOS boot                               21686148-6449-6E6F-744E-656564454649
 5 Sony boot partition                     F4019732-066E-4E12-8273-346C5641494F
 6 Lenovo boot partition                   BFBFAFE7-A34F-448A-9A5B-6213EB736C22
 7 PowerPC PReP boot                       9E1A2D38-C612-4316-AA26-8B49521E5A8B
 8 ONIE boot                               7412F7D5-A156-4B13-81DC-867174929325
 9 ONIE config                             D4E6E2CD-4469-46F3-B5CB-1BFF57AFC149
10 Microsoft reserved                     E3C9E316-0B5C-4DB8-817D-F92DF00215AE
11 Microsoft basic data                   EBD0A0A2-B9E5-4433-87C0-68B6B72699C7
12 Microsoft LDM metadata                 5808C8AA-7E8F-42E0-85D2-E1E90434CFB3
13 Microsoft LDM data                     AF9B60A0-1431-4F62-BC68-3311714A69AD
14 Windows recovery environment           DE94BBA4-06D1-4D40-A16A-BFD50179D6AC
15 IBM General Parallel Fs                 37AFFC90-EF7D-4E96-91C3-2D7AE055B174
16 Microsoft Storage Spaces               E75CAF8F-F680-4CEE-AFA3-B001E56EFC2D
17 HP-UX data                             75894C1E-3AEB-11D3-B7C1-7B03A0000000
18 HP-UX service                         E2A1E728-32E3-11D6-A682-7B03A0000000
19 Linux swap                             0657FD6D-A4AB-43C4-84E5-0933C84B4F4F
20 Linux filesystem                       0FC63DAF-8483-4772-8E79-3D69D8477DE4
21 Linux server data                     3B8F8425-20E0-4F3B-907F-1A25A76F98E8
22 Linux root (x86)                       44479540-F297-41B2-9AF7-D131D5F0458A
23 Linux root (x86-64)                   4F68BCE3-E8CD-4DB1-96E7-FBCAF984B709
24 Linux root (Alpha)                     6523F8AE-3EB1-4E2A-A05A-18B695AE656F
25 Linux root (ARC)                       D27F46ED-2919-4CB8-BD25-9531F3C16534
```

Regresando al proceso, tipeamos el número **1** pues queremos una partición de tipo [EFI](#).

```
Command (m for help): t
Partition number (1-4, default 4): 4
Partition type or alias (type L to list all): 1
Changed type of partition 'Linux filesystem' to 'EFI System'.
Command (m for help): p
```

Después verificamos que si se hizo el cambio en la partición correspondiente, en este caso en [/dev/sda4](#) escribiendo en la terminal la opción [p](#).

```
Device          Start      End        Sectors    Size Type
/dev/sda1        2048      4095        2048      1M BIOS boot
/dev/sda2        4096    2101247   2097152    1G Linux filesystem
/dev/sda3       2101248  31250431  29149184  13.9G Linux filesystem
/dev/sda4       31250432  31455231    204800    100M EFI System
Command (m for help):
```




Finalmente, damos la opción **w** que nos dice **write table to disk and exit** para guardar los cambios que hicimos.

```
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
[root@localhost-live ~]#
```

De esta manera podemos hacer una partición y ponerle el tipo que nos sea conveniente usar dependiendo de las necesidades que tengamos.

8.4. ¿Cómo formatear una partición en Linux?

Así como podemos crear particiones también las podemos formatear. Esto lo podemos hacer para reparar los siguientes inconvenientes:

- Cambiar una partición de tipo **fat32** por ejemplo a una partición de tipo **ntfs** o viceversa. Esto más que nada porque pueden haber errores del sistema debido a la incompatibilidad de archivos.
- Formateamos para limpiar el disco y hacer una instalación de algún Sistema.
- El disco está dañado.

Podemos hacer el formateo de manera manual sin necesidad de una interfaz gráfica. Primero verificamos las particiones del disco para ver cuál disco queremos formatear y después lo formateamos.

Supongamos que la partición del disco duro se llama **/dev/sda4** y la queremos formatear a **ext**. Abrimos la terminal y escribimos la siguiente línea de comandos:

```
$ sudo fdisk -l
$ sudo mkfs.ext4 /dev/sda4
```

Si quisieramos formatear a **ntfs**, entonces el comando sería:

```
$ sudo mkfs.ntfs /dev/sda4
```

Por último, si lo que queremos es **limpiar** el disco duro podemos usar **wipe**.
Lo instalamos si es que no lo tenemos disponible:

```
$ sudo apt-get install wipe
```

Para usarlo solo sería usando el siguiente comando:

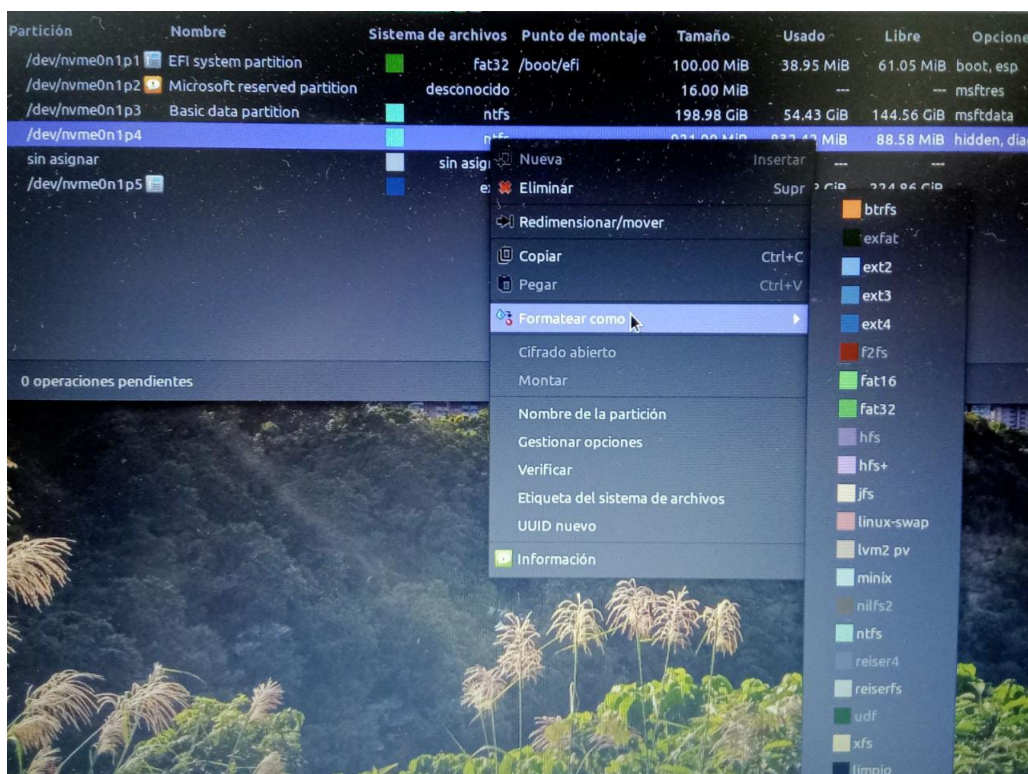
```
$ sudo wipe /dev/sda4
```

Para hacer el formateo de una partición del disco duro en Linux de manera gráfica, podemos usar **GParted**.
Lo único que debemos hacer es abrir la terminal y escribir en la línea de comandos:

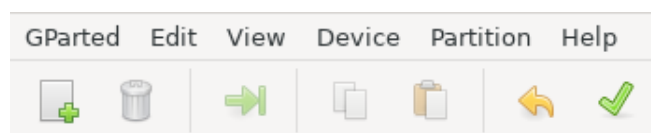
```
$ gparted
```



Nos posicionamos en la partición que queremos formatear, damos click derecho y vamos a la opción de [Formatear como](#) y nos da las diferentes opciones que tenemos para hacer el formateo del disco.



Al finalizar, damos click en la fecha verde para que [GParted](#) proceda a hacer los cambios.



Nuestra partición ha sido formateada de manera correcta.

Nota: Si quieres formatear una partición de disco en Windows, puedes hacerlo desde el siguiente [link](#).



**Facultad de
Ciencias**
UNAM

Sistemas de arranque y Particiones.
Universidad Nacional Autónoma de México
Sistemas Operativos
Autor: Miriam Torres Bucio.

Referencias

- [1] Información sobre el GRUB.
<http://web.archive.org/web/20100102212345/http://www.ubuntu-es.org/?q=node/60473>
- [2] Instalación de Windows.
<https://learn.microsoft.com/es-es/windows-hardware/manufacture/desktop/windows-setup-installing-using-the-mbr-or-gpt-partition-style?view=windows-11>
- [3] GParted.
<https://itslinuxfoss.com/how-to-install-and-use-gparted-on-ubuntu-20-04/>