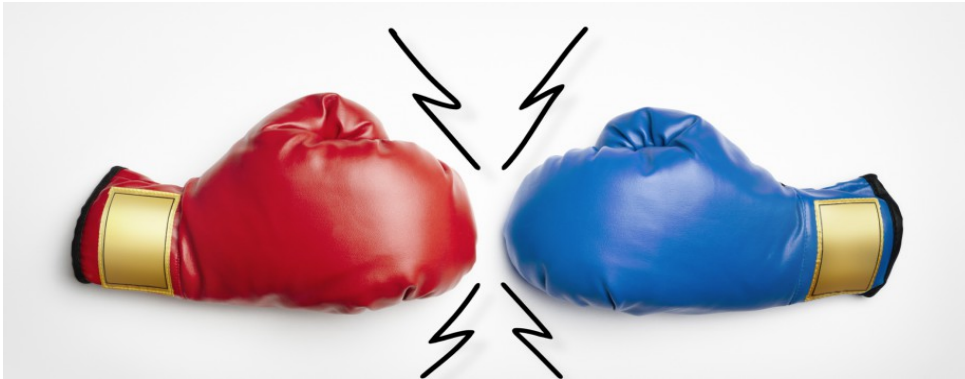


## TRABAJO PRÁCTICO ADMINISTRACIÓN GNU/LINUX

# GPT VS MBR



TRABAJO CREADO POR DESIO GUSTAVO YAMIL

PROFESOR A CARGO DEL CURSO: EXEQUIEL ARAMBURU

TEMA DEL TRABAJO: TABLA DE PARTICIONES “GPT Y MBR”

AÑO LECTIVO 2018

## Indice

INTRODUCCIÓN.....	3
MBR.....	4
GPT.....	6
INSTALACIÓN DE GPT Y MBR DESDE DEBIAN GNU/LINUX.....	8
CONCLUSIÓN.....	11
BIBLIOGRAFÍA.....	12
GNU Free Documentation License.....	13

## INTRODUCCIÓN

¿**MBR** vs **GPT**? ¿Cuál usar? En este trabajo explico ambos sistemas para crear tablas de particiones, sus características, y como crear dichas tablas de particiones. A su vez espero que este trabajo pueda ayudar al lector/a a resolver problemas que surgen a la hora de instalar un sistema operativo y también a la hora de crear un medio de instalación del mismo. Soy un Técnico de computadoras que obtuvo su título del Laboratorio de Investigación GUGLER y me apasiona el tema de las computadoras. [Para instalar algunas distros de GNU/Linux es recomendable leer en sus páginas oficiales como instalarlo, ya que especifican si es conveniente usar MBR o GPT, y si es conveniente bootear o arrancar desde BIOS o UEFI.](#) Si en este trabajo usted encuentra algún error, algo mal explicado, o ve la necesidad de ampliar el contenido, me gustaría que se comunique conmigo a si me nutro de su sabiduría y puedo seguir aprendiendo.

## MBR

Master Boot Record y traducido al español Registro de Arranque Maestro, es una tabla para realizar particiones a partir de ella, se coloca en el sector 0 de algún disco de almacenamiento, dicho sector abarca 512 bytes, y se utiliza para albergar esta tabla de particiones llamada MBR y también aquí se instala un cargador de arranque para que se pueda iniciar un sistema operativo (de los más conocidos GRUB y LILO).

### CARACTERÍSTICAS DE MBR:

-Puede tener en su registro solo 4 particiones primarias. Si se desea tener mas de 4 particiones, es necesario crear una partición extendida, la misma se toma como partición primara, y dentro de ésta se pueden colocar particiones lógicas, que a su vez hay una limitante de n lógicas (esto depende del sistema operativo). Quedando así 3 particiones primarias y una cuarta extendida con n lógicas dentro de ella.

-Solo puede haber una partición extendida.

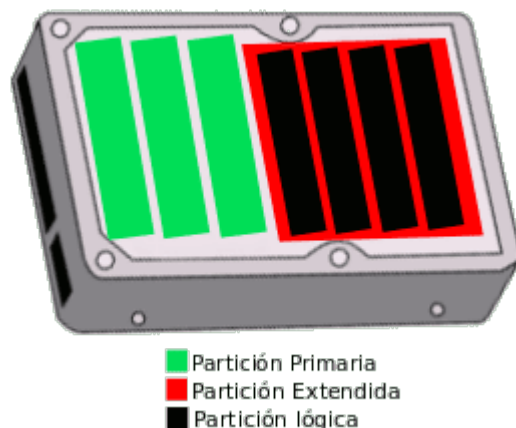


IMAGEN OBTENIDA DEL SIGUIENTE LINK

<https://usuariodebian.blogspot.com/2017/09/diferencia-entre-bios-y-uefi-mbr-y-gpt.html>

-El límite de tamaño por partición es de 2.2TB.

-El MBR puede ser leído tanto por sistemas BIOS y UEFI, pero no siempre esto es así, algunas UEFI pueden no llevarse bien con el MBR, y en especial las UEFI con el Secure Boot activo, para resolver dicho problema, hay que entrar en la UEFI y desactivar el Secure Boot. En lo personal recomiendo NO usar UEFI si tenemos MBR (en realidad las PC con sistema UEFI, solo poseen ese sistema, pero además incorporan compatibilidad para usar discos con MBR y nos dan la opción de usar UEFI o no. A este tipo de compatibilidad se lo puede encontrar como BIOS Legacy o UEFI con CSM).

-El MBR tiene una lista de los distintos sistemas operativos instalados y nos permite ejecutar alguno de ellos, también posee información sobre cada partición contenida en el disco de almacenamiento (en que sector comienza cada una y en que sector termina, si la misma es arrancable o no y el formato de la misma).

-MBR trabaja con CHS cylinder-head-sector / cilindro-cabezal-sector, y también con LBA logical-block-addressing / direccionamiento de bloque lógico. CHS es una manera de comunicar detalles físicos del disco directamente al sistema operativo. Y utilizando LBA el sistema operativo no se comunica directamente con los detalles físicos del disco, sino que el sistema operativo se comunica con el controlador de la unidad y es este el que se encarga de comunicar el disco con el sistema operativo.

## ESQUEMA DEL MBR DENTRO DEL SECTOR 0 DE 512 BYTES



IMAGEN OBTENIDA DEL SIGUIENTE LINK  
[https://es.wikipedia.org/wiki/Tabla\\_de\\_particiones](https://es.wikipedia.org/wiki/Tabla_de_particiones)

## GPT

GUID Partition Table y traducido al español, Tabla de Particiones GUID. Esta tabla de particiones se crea junto con EFI/UEFI, por lo que GPT y UEFI van de la mano, junto con la arquitectura de 64 bit de procesadores. Para una buena compatibilidad se recomienda GPT con UEFI y 64 en la arquitectura de procesadores. GPT nace para ampliar las características de MBR y para mejorar la seguridad, un conjunto de compañías se reunieron y desarrollaron GPT y UEFI. Esta tabla de particiones abarca generalmente 2048 o 4096 bytes, por lo que el sector 0 del disco de almacenamiento puede ser mayor a 512 bytes (en realidad los sectores de los discos estaban establecidos en 512 bytes como estándar, pero hoy en día esto puede ser diferente, y podemos encontrarnos con discos de 4096 bytes por sector; CON SECTOR DE 512 BYTES O 4096 BYTES ME REFIERO A SECTORES FÍSICOS EN CUANTO A SU FABRICACIÓN, pero si hablamos de sectores “lógicos”, pueden variar dependiendo a las necesidades del usuario. Un ejemplo de sector “lógico” sería si tenemos un disco fabricado con sector físico de 512 bytes, pero colocamos en el mismo GPT, esta tabla de particiones ocupa mas de 512 bytes, por lo que se creará un sector lógico para albergar la tabla GPT).

La tabla de particiones GPT al igual que la tabla de particiones MBR poseen un registro e información de las particiones albergadas en el disco de almacenamiento. El GPT almacena los gestores de arranque como por ejemplo GRUB y LILO en un área denominada “protective MBR”. Pero también, usando esta tabla de particiones en un dispositivo de almacenamiento, si el sistema operativo lo permite, se puede almacenar el grub en una partición diferente, denominada “BIOS Boot Partition”. Para crear esta partición se debe tener la herramienta adecuada, GNU Parted es una de ellas. Si desea mas información sobre esto dejo el link del manual de GNU GRUB donde se explica esto.

<https://www.gnu.org/software/grub/manual/grub/grub.html#BIOS-installation>

### CARACTERÍSTICAS DE GPT:

-Esta tabla de particiones puede tener mas de 4 particiones primarias (en realidad en la tabla GPT ya no existen particiones primarias, extendidas ni lógicas, simplemente son particiones).

-El límite de particiones está como standar en 128, pero se puede editar la tabla y colocar mas particiones y por partición se acepta 2 ZiB.

2 ZiB = 2361183241.4348 TB conversión hecha en <https://www.unitjuggler.com/memory-convertir-ZiB-a-TB.html?val=2>

-La tabla de partición GPT en teoría puede ser reconocida por BIOS antiguas ya que la misma posee en LBA0 un MBR protector, el cual es un área de compatibilidad para BIOS que no poseen UEFI, pero su principal función es la de proteger el área GPT de software viejo que no reconoce dicha tabla.

-Teniendo la tabla GPT se puede crear una partición exclusiva para el gestor de arranque GRUB. GRUB se puede incrustar en esa partición sin el riesgo de ser sobrescrito por otro software y sin estar contenido en un sistema de archivos que pueda mover sus bloques. (más arriba dejé link hacia el manual de GNU GRUB para realizar esto).

-La tabla GPT esta hecha para trabajar en conjunto con el Secure Boot de UEFI.

-Ofrece mayor seguridad que MBR. Tanto en temas de malware como en tema de integridad de los datos que posee la tabla, ya que posee un backup de la información de las particiones al final del disco.

-GPT trabaja solamente con LBA, estos son direccionamientos de bloques lógicos que se comunican con el sistema operativo. Es más seguro trabajar con estos bloques lógicos que trabajar con CHS cylinder-head-sector que usa el MBR.

## ESQUEMA DE LA TABLA GPT DENTRO DE SU SECTOR

### GUID Partition Table Scheme

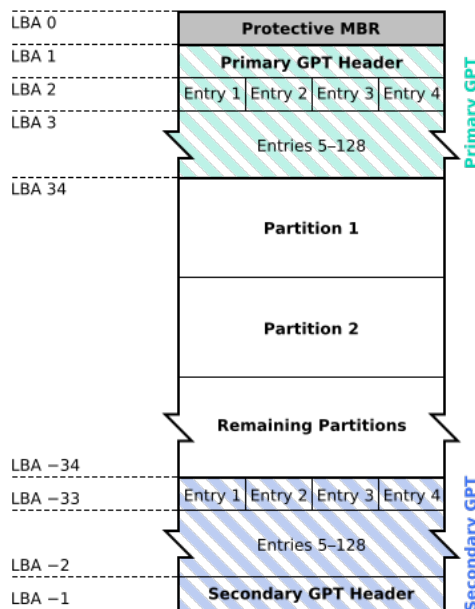


IMAGEN OBTENIDA DEL SIGUIENTE LINK  
[https://es.wikipedia.org/wiki/Tabla\\_de\\_particiones\\_GUI](https://es.wikipedia.org/wiki/Tabla_de_particiones_GUI)

# INSTALACIÓN DE GPT Y MBR DESDE DEBIAN GNU/LINUX

A la hora de instalar una tabla de particiones debemos tener presente que dicha instalación borrará todos los datos contenidos en el disco, ya que lo primero a realizar en un disco de almacenamiento es crear la tabla de particiones, de allí se procede a crear las particiones que albergarán el o los sistemas operativos y los datos y archivos del usuario.

Para instalar GPT o MBR desde una interfaz gráfica se puede usar el Gparted, es un programa para manipular discos de almacenamiento y posee una interfaz gráfica (hago una aclaración, menciono muchas veces la palabra “disco” pero es solo cuestión de costumbre, hoy en día los SSD, las memoras m.2, las memorias m.2 con interfaz NVMe están lejos de tener forma de disco, jaja). Y para instalar desde consola se puede usar fdisk.

## Instalación interfaz gráfica

**-Primer paso** para la creación de tabla GPT o MBR, es conseguir el Gparted, se puede descargar desde su página oficial <https://gparted.org/> o desde Debian, apt install gparted, se puede usar desde una distro de GNU/Linux o desde un live usb.

**-Segundo paso** abrimos el Gparted, nos situamos sobre dispositivo y luego vamos a Crear tabla de particiones.



**-Tercer paso** elegimos el tipo de tabla de particiones, en el Gparted el MBR figura como “msdos”. Damos aplicar y esperamos.

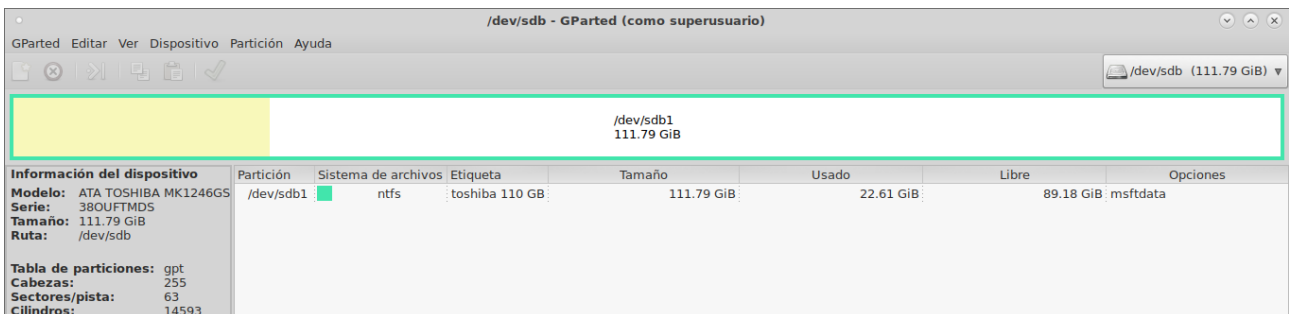




Para ver el tipo de partición que posee un dispositivo, dentro del Gparted nos situamos en la parte superior derecha, elegimos nuestro dispositivo, y luego nos situamos en la pestaña ver, y elegimos información de dispositivo.

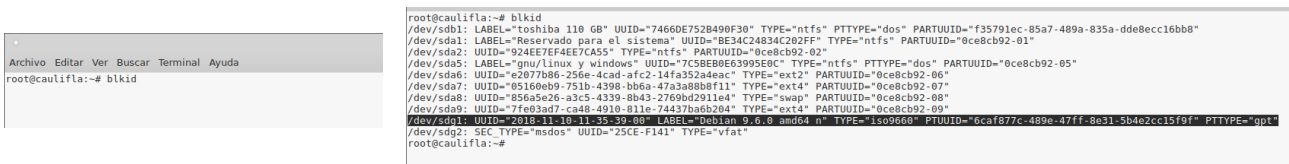


A la izquierda se mostrarán detalles del dispositivo, entre ellos “la tabla de particiones” que posee.



## Instalación desde consola

**Primer paso** con el comando “blkid” identificamos nuestro dispositivo de almacenamiento. (se necesita estar logueados como root para usar blkid)



**Segundo paso** como ayuda extra para asegurarnos de que es el dispositivo correcto podemos pasar un “cfdisk /dev/sdg” (siempre que hagamos una tabla de partición o queramos referirnos a la misma, se especifica /dev/sdX, esto hace referencia al sector 0). Ahora se nos muestra el dispositivo y se puede ver su tamaño, si es el dispositivo correcto proseguimos.



**Tercer paso** (siempre que hagamos una tabla de partición o queramos referirnos a la misma, se especifica /dev/sdX, esto hace referencia al sector 0) Usamos el programa fdisk “fdisk /dev/sdg” y presionamos la letra m. De todas maneras fdisk es súper fácil de usar, es solo cuestión de leer lo que nos dice el programa.

```
Archivo Editar Ver Buscar Terminal
root@caulifla:~# fdisk /dev/sdg
```

```
Archivo Editar Ver Buscar Terminal Ayuda
root@caulifla:~# fdisk /dev/sdg
Bienvenido a fdisk (util-linux 2.29.2).
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.
Tenga cuidado antes de utilizar la orden de escritura.

Orden (m para obtener ayuda): m
```

**Cuarto paso** hay que crear una nueva etiqueta, vemos en el menú de fdisk dicha opción, y elegimos la letra correspondiente a la tabla de particiones que queremos. En este caso voy a usar MBR, que en fdisk se especifica como DOS. Se coloca la letra “o” y damos enter(1). Luego en el apartado **GUARDAR Y SALIR**, elegimos la letra “w” la cual es para escribir la tabla en el disco y salir(2). Luego podemos volver a entrar a fdisk al mismo dispositivo y con la letra “p” podemos ver la tabla de particiones que posee el dispositivo que se especifica como “tipo de etiqueta de disco”(3). (abajo las imágenes del cuarto paso en su respectivo orden)

```
q sale sin guardar los cambios
Crea una nueva etiqueta
g crea una nueva tabla de particiones GPT vacía
G crea una nueva tabla de particiones SGI (IRIX) vacía
o crea una nueva tabla de particiones DOS vacía
s crea una nueva tabla de particiones Sun vacía

Orden (m para obtener ayuda): o
```

```
I carga la estructura del disco de un fichero de scri
O vuelca la estructura del disco a un fichero de scri
Guardar y Salir
w escribe la tabla en el disco y sale
q sale sin guardar los cambios
```

```
General
d borra una partición
F lista el espacio libre no particionado
l lista los tipos de particiones conocidos
n añade una nueva partición
p muestra la tabla de particiones
t cambia el tipo de una partición
v verifica la tabla de particiones
i imprime información sobre una partición
```

```
Orden (m para obtener ayuda): p
Disco /dev/sdg: 3,7 GiB, 400560320 bytes, 7823360 sectores
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0x5edf5a2d
```

## CONCLUSIÓN

A modo de conclusión comento que realizar este trabajo me ha gustado mucho, he aprendido muchas cosas.

Hoy en día la tabla de particiones MBR al igual que la tabla de particiones GPT son muy usadas ambas, pero puede ser que en un futuro solo nos quedemos con la tabla de particiones GPT debido a su mayor seguridad y a su mejor soporte en cuanto a cantidad de particiones y almacenamiento.

## BIBLIOGRAFÍA

Si desea investigar mas por su cuenta o indagar en mas detalles acerca de las tablas de particiones y todo lo que esta relacionado con las mismas, dejo los enlaces a las páginas que he visitado para realizar este trabajo, también he recolectado información del actual curso que estoy realizando el cual es Administración en GNU/Linux.

Una buena página para comenzar a leer es esta:

[https://wiki.archlinux.org/index.php/Partitioning#GUID\\_Partition\\_Table](https://wiki.archlinux.org/index.php/Partitioning#GUID_Partition_Table)

<https://www.gnu.org/software/grub/manual/grub/grub.html#BIOS-installation>

[https://es.wikipedia.org/wiki/Direccionamiento\\_de\\_bloque\\_l%C3%B3gico](https://es.wikipedia.org/wiki/Direccionamiento_de_bloque_l%C3%B3gico)

<https://superuser.com/questions/1165557/how-grub2-works-on-a-mbr-partitioned-disk-and-gpt-partitioned-disk>

<https://usuariodebian.blogspot.com/2017/09/diferencia-entre-bios-y-uefi-mbr-y-gpt.html>

[https://es.wikipedia.org/wiki/Tabla\\_de\\_particiones](https://es.wikipedia.org/wiki/Tabla_de_particiones)

[https://es.wikipedia.org/wiki/Tabla\\_de\\_particiones\\_GUID](https://es.wikipedia.org/wiki/Tabla_de_particiones_GUID)

[https://es.wikipedia.org/wiki/Extensible\\_Firmware\\_Interface](https://es.wikipedia.org/wiki/Extensible_Firmware_Interface)

[https://es.wikipedia.org/wiki/Sector\\_\(inform%C3%A1tica\)](https://es.wikipedia.org/wiki/Sector_(inform%C3%A1tica))

<https://www.debian.org/releases/stable/amd64/apcs05.html.es#idm4122>

[https://es.wikipedia.org/wiki/Unidad\\_de\\_disco\\_duro#cite\\_note-2](https://es.wikipedia.org/wiki/Unidad_de_disco_duro#cite_note-2)

[https://wiki.debian.org/UEFI#Booting\\_a\\_UEFI\\_machine\\_normally](https://wiki.debian.org/UEFI#Booting_a_UEFI_machine_normally)

[https://wiki.archlinux.org/index.php/Partitioning\\_\(Espa%C3%B1ol\)#GUID\\_Partition\\_Table](https://wiki.archlinux.org/index.php/Partitioning_(Espa%C3%B1ol)#GUID_Partition_Table)

## **GNU Free Documentation License**

Copyright (C) 2018 GUSTAVO YAMIL DESIO.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".