

SCRIPTING



COMANDOS DENTRO DEL SCRIPT

Recordatorio: Sintaxis de comandos



orden [**-modificadores**] [**argumentos**]

- parámetros
- opciones

Orden: acción a realizar

Modificadores: precedidos del carácter – guion

Argumentos: entrada del comando (archivo o un directorio)



cp: Copiar archivos y directorios.

mv: Mover archivos y directorios.

rm: Eliminar archivos y directorios.

mkdir: Crear directorios.

rmdir: Eliminar directorios vacíos.

touch: Crear archivos vacíos *o actualizar la fecha de modificación de archivos existentes.*

umask: establece los permisos a eliminar de los predeterminados al crear un fichero o directorio

find: Buscar archivos y directorios que cumplan ciertos criterios.

grep: Buscar patrones de texto dentro de archivos.

sed: Editar archivos de texto de forma no interactiva, *utiliza expresiones regulares (patrones complejos) para buscar y manipular/transformar texto de forma precisa.*



umask

Determina qué permisos serán **eliminados** del conjunto completo de permisos al crear nuevos archivos y directorios

```
umask 022
```

permisos predeterminados

RESTAR

NUEVOS*

ARCHIVOS
DIRECTORIOS

666
777

022
022

644
755

*touch / mkdir



echo: Mostrar texto en la salida estándar.

read: Leer entrada del usuario y asignarla a una variable.

sleep: Pausar la ejecución del script durante un período de tiempo especificado.

exit: Salir del script con un código de salida específico.

trap: Capturar y manejar señales del sistema, como SIGINT (Ctrl+C) y SIGTERM.



ps: Mostrar información sobre los procesos en ejecución.

kill: Enviar una señal a un proceso especificado.

wait: Esperar a que finalicen los procesos secundarios lanzados por el script.



- >: Redireccionar la salida estándar a un archivo (sobrescribir).
- >>: Redireccionar la salida estándar a un archivo (agregar al final).
- <: Redireccionar la entrada estándar desde un archivo.
- |: Crear una tubería para enviar la salida de un comando como entrada a otro comando.



useradd: Crear un nuevo usuario.

userdel: Eliminar un usuario.

passwd: Cambiar la contraseña de un usuario.

usermod: Modificar las propiedades de un usuario existente.

id: Mostrar información sobre un usuario, incluido su ID de usuario (UID) y sus grupos.

who: Mostrar información sobre los usuarios que están actualmente conectados.

w: Mostrar quién está conectado y qué están haciendo.

last: Mostrar el historial de inicio de sesión de los usuarios.

chown: Cambiar el propietario de un archivo o directorio.

groupadd: Crear un nuevo grupo.

groupdel: Eliminar un grupo.

groupmod: Modificar las propiedades de un grupo existente.

groups: Mostrar los grupos a los que pertenece un usuario.

chgrp: Cambiar el grupo propietario de un archivo o directorio.



su: Cambiar de usuario (por defecto, al superusuario **root**).

sudo: Ejecutar comandos con privilegios de superusuario.

visudo: Editar el archivo `sudoers` de forma segura.

adduser: Interfaz más amigable para agregar nuevos usuarios.

deluser: Interfaz más amigable para eliminar usuarios.

addgroup: Interfaz más amigable para agregar nuevos grupos.

delgroup: Interfaz más amigable para eliminar grupos.

setacl: configuración de permisos para usuarios y/o grupos específicos

getacl: Verifica la configuración de la ACL



El fichero ***/etc/sudoers*** es un archivo de configuración que define qué usuarios y grupos tienen permiso para ejecutar comandos con privilegios de superusuario utilizando el comando sudo

```
$ sudo visudo /etc/sudoers
```

```
usuario(s)    host=(usuario_a_ejecutar)    comando(s)
```

```
$ sudo visudo -c
```

Verificación de la sintaxis



ACL - "Access Control List" (Lista de Control de Acceso)

Conjunto de reglas asociadas a un objeto (como un archivo o un directorio) que especifica los **permisos** de acceso **individuales** para **usuarios y grupos específicos**, complementando o extendiendo los permisos tradicionales de UNIX (lectura, escritura y ejecución).

```
setfacl -m u:usuario1:rwx,g:grupo1:rwx dir1
```

```
getfacl dir1
```



Si `mount | grep -i acl` NO da ninguna ocurrencia:

1. Verificar que ACL está habilitado en el kernel

```
grep CONFIG_FS_POSIX_ACL /boot/config-$(uname -r)
```

Si el comando muestra **CONFIG_FS_POSIX_ACL=y**, significa que el soporte para ACL está habilitado en el kernel.

2. Instalar el paquete ACL

3. Montar los sistemas de archivos con soporte ACL

Para
automáticamente

```
sudo nano /etc/fstab  
UUID=XXX /mount/point ext4 defaults,acl 0 2
```

4. Reiniciar o remontar el sistema de archivos

```
sudo mount -o remount /mount/point
```



hwinfo: Proporciona una visión general detallada del hardware del sistema, incluyendo CPU, memoria, discos, tarjetas de red, etc. (Puede requerir instalación adicional en algunas distribuciones).

dmidecode: Proporciona información del Sistema de Gestión de Datos (DMI) del sistema, incluyendo detalles sobre la BIOS, la placa base, la memoria, etc.

uptime: Muestra cuánto tiempo ha estado funcionando el sistema y carga promedio.

uname -r: Proporciona información sobre la versión del kernel



lscpu: Muestra información detallada sobre la **CPU del sistema**, incluyendo el número de núcleos, la arquitectura, la velocidad del reloj y más.

lsblk: Muestra información sobre los **dispositivos de bloques** del sistema, como discos duros y unidades USB, así como sus particiones.

lsusb: Muestra una lista de todos los **dispositivos USB** conectados al sistema.

lspci: Muestra información sobre los **dispositivos PCI** conectados al sistema, como tarjetas gráficas, controladores de red, etc.

lsdev: Muestra una lista de todos los **dispositivos en el sistema**.



free: Muestra información sobre el uso de memoria del sistema, incluyendo memoria total, usada y libre.

df: Muestra información sobre el espacio en disco disponible en todas las particiones montadas del sistema.

du: muestra el uso del espacio en disco

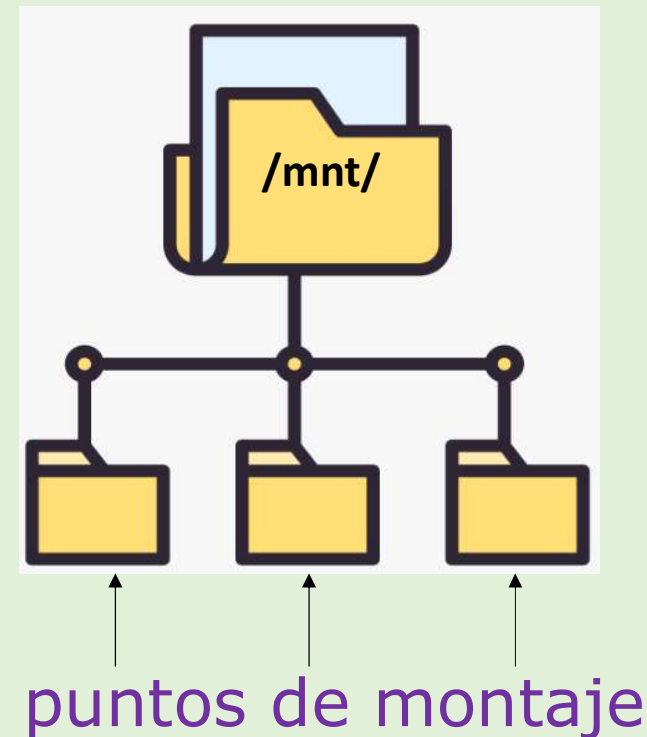
mount: para montar sistemas de fichero en el árbol de directorios del sistema

mdadm: crea sistema RAID

fdisk: administración de particiones

rsync: copias de seguridad

“incorporar un sistema de archivos almacenado en un dispositivo de almacenamiento en el árbol de directorios del sistema”



```
lsblk
```

```
mkdir /mnt/usb32
```

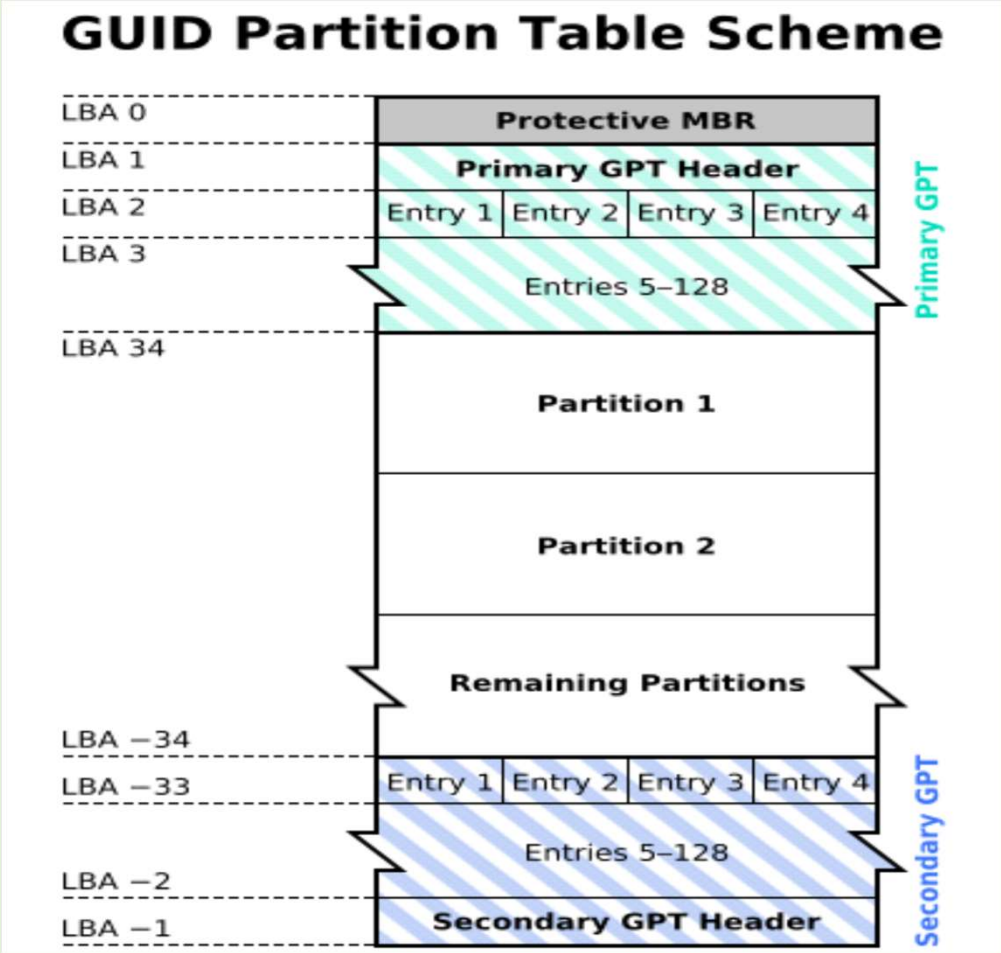
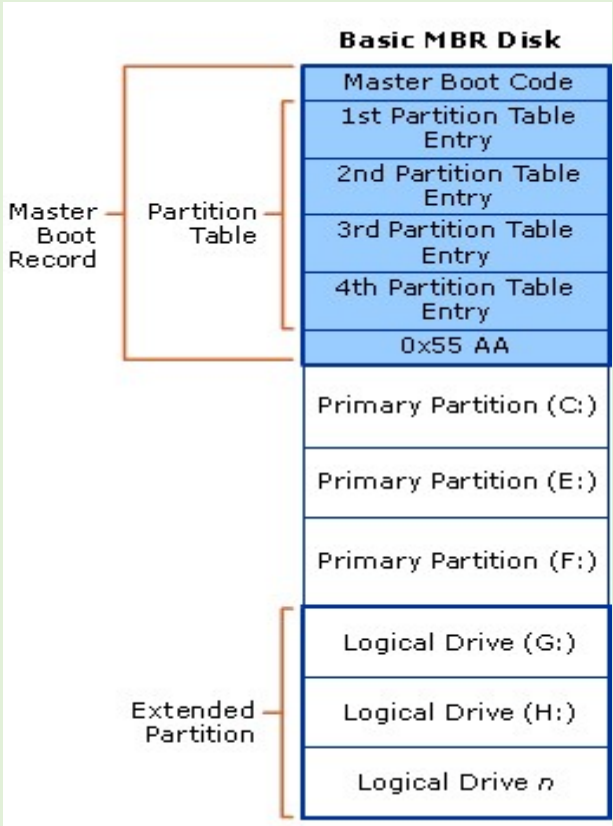
```
mount /dev/sda1 /mnt/usb32
```

```
/etc/fstab
```

AMPLIACIÓN – PARTICIONADO

¿Por qué necesitamos particionar un disco?

- Por organización de datos
- Para alojar varios sistemas operativos en un solo disco



Comando fdisk - para crear, eliminar, cambiar, mostrar y administrar particiones en unidades de almacenamiento

- ✓ Creación y gestión de particiones en discos nuevos
- ✓ Redimensionamiento de particiones para hacer espacio para nuevos sistemas operativos o datos
- ✓ Eliminación de particiones antiguas o no utilizadas para liberar espacio en el disco

fdisk opciones dispositivo

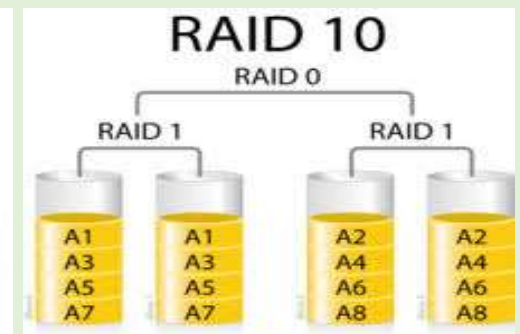
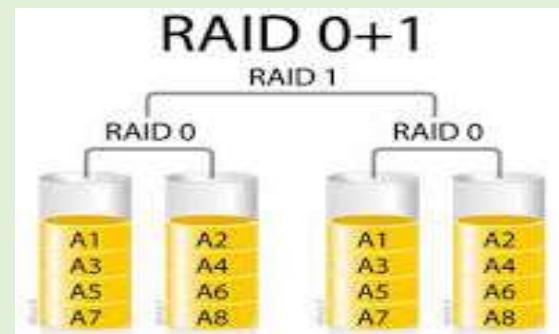
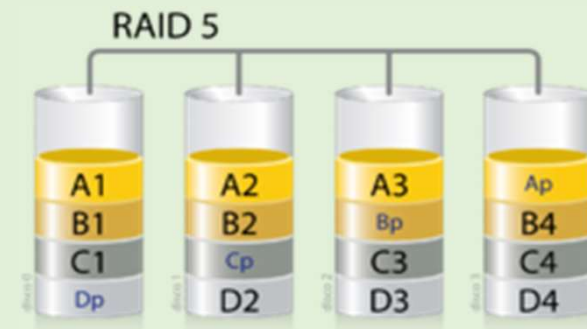
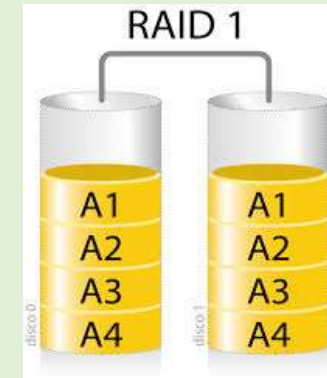
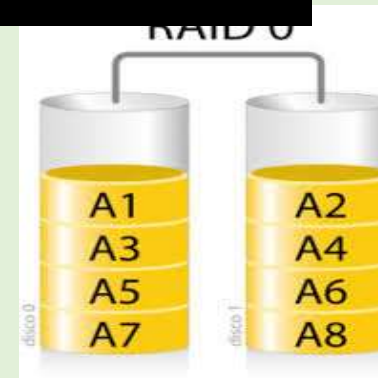
- l** para listar la tabla de particiones
- s** para mostrar el tamaño de una partición
- t** para cambiar el tipo de una partición
- n** para crear una partición nueva
- d** para borrar una partición
- v** modo verbose

AMPLIACIÓN – RAID – NIVELES

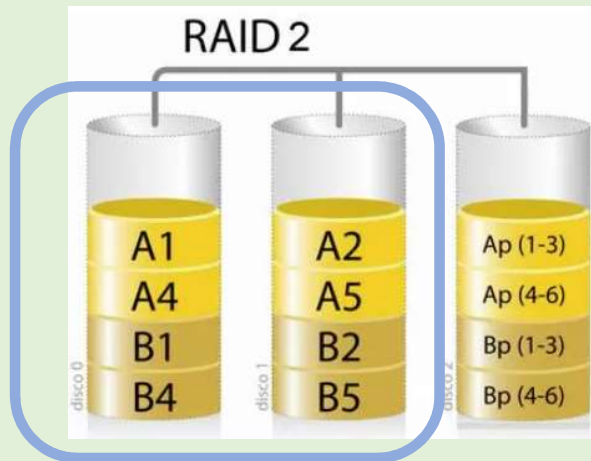
- Volumen seleccionado o **RAID 0** (División)
- Volumen reflejado o **RAID 1** (Espejo)
- Volumen **RAID 5** (Distribuido con paridad)
- “Anidados”, para incrementar el rendimiento y la redundancia:

✓ Espejo de divisiones **RAID 0+1**

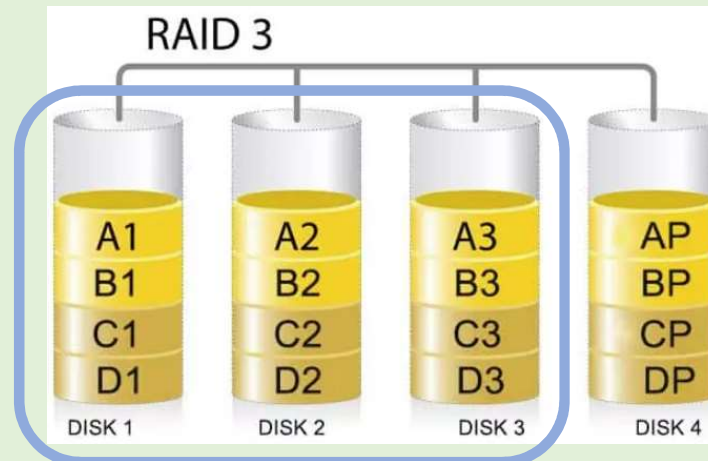
✓ División de espejos **RAID 1+0**



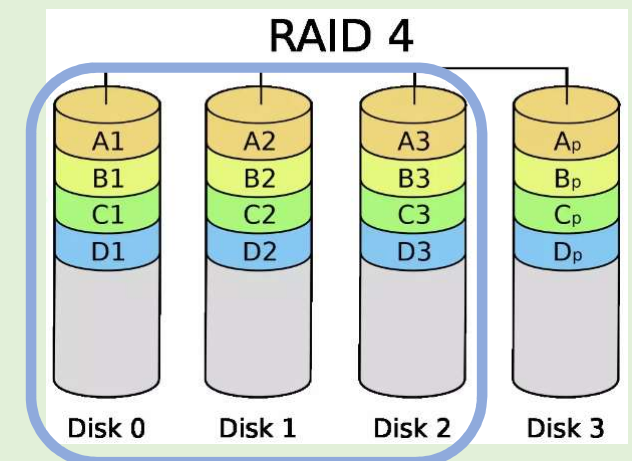
Paridad dedicada



bits

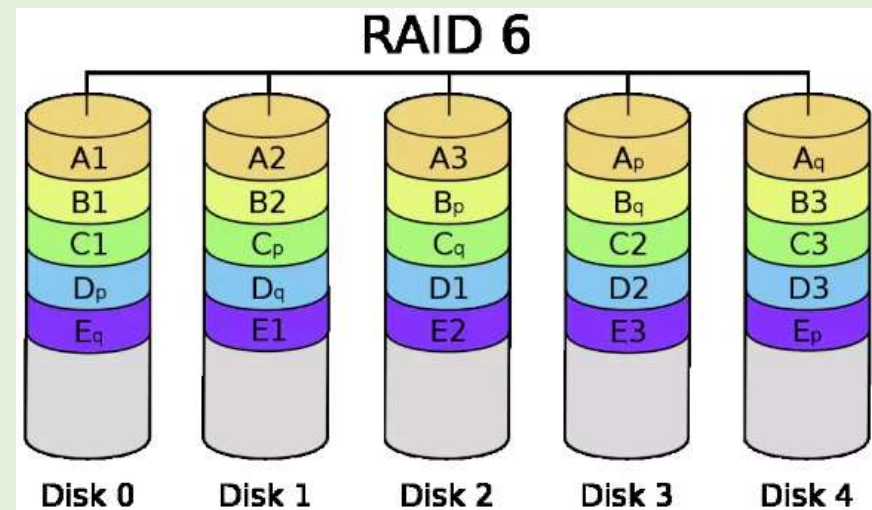
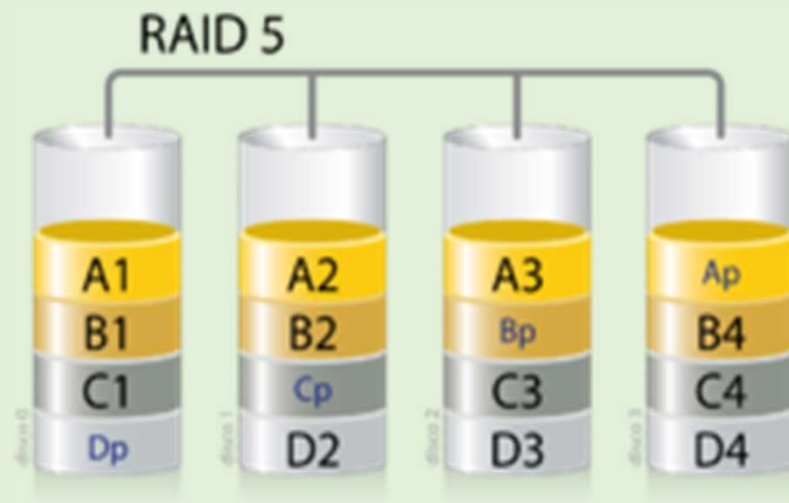


Byte

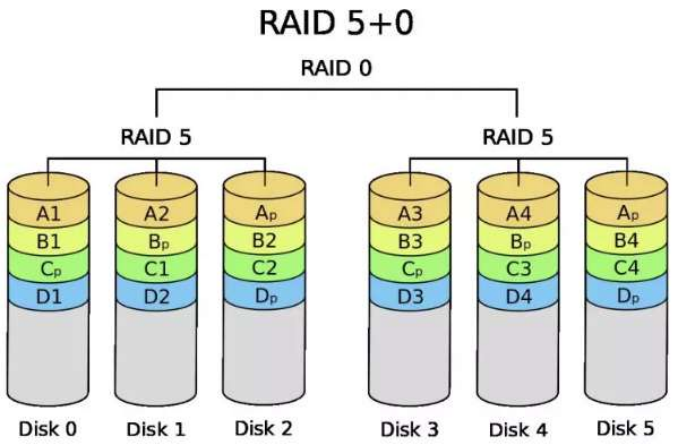
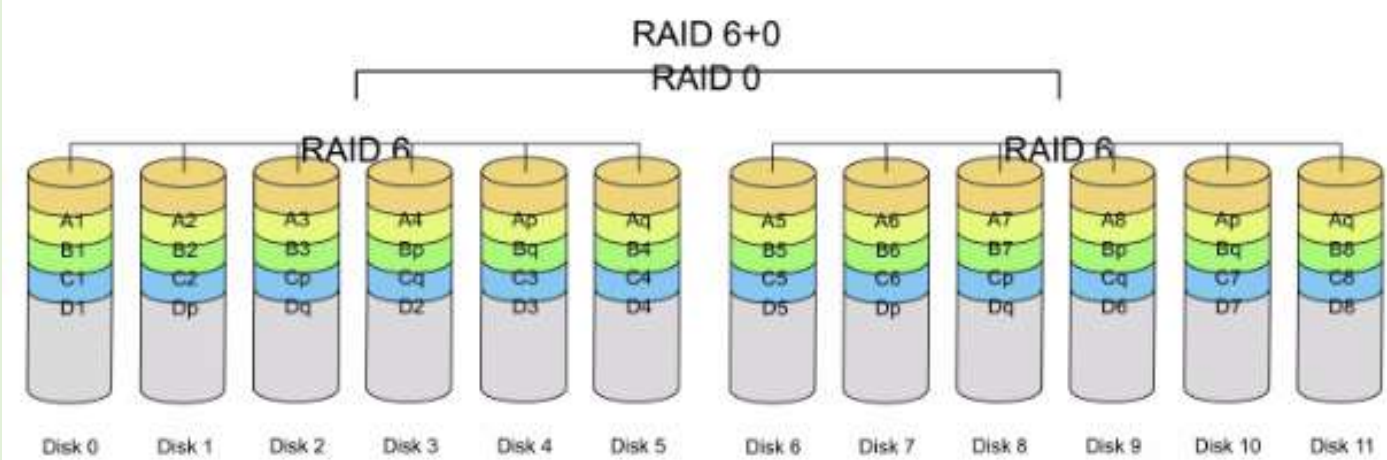
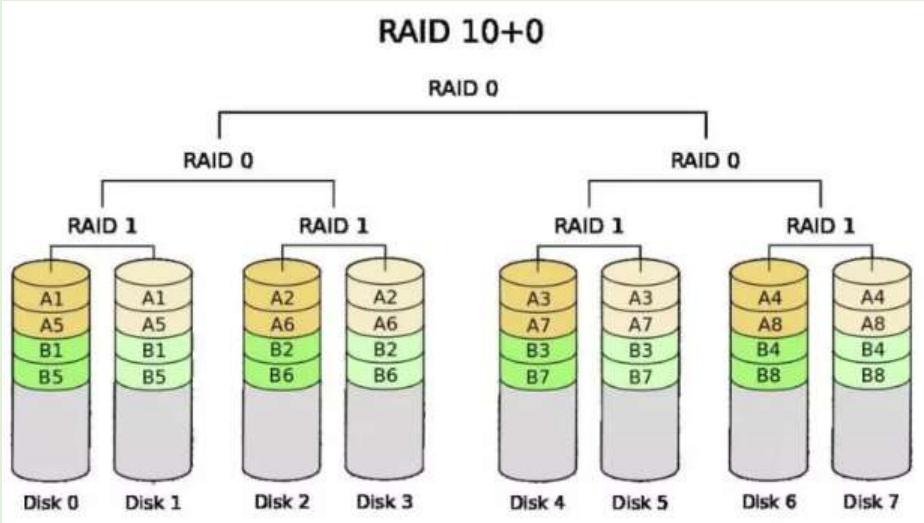
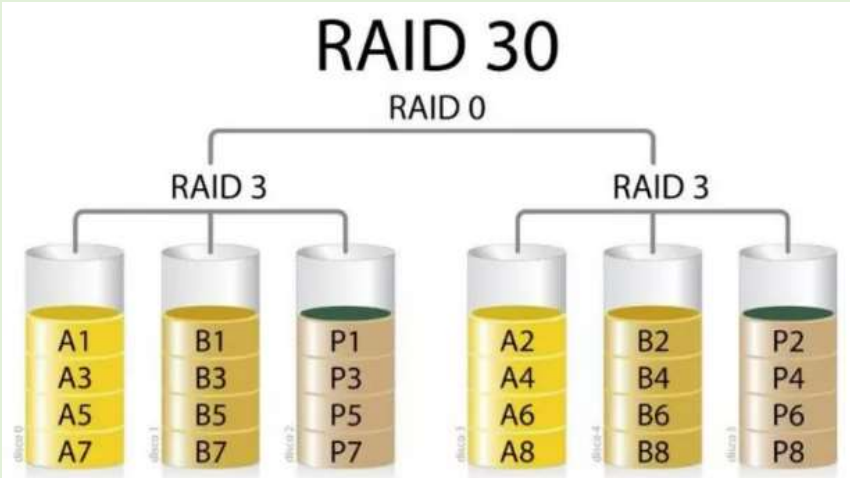


Bloque

Paridad distribuida



Volúmenes anidados





Comando mdadm, para crear la matriz RAID

Algunas opciones del comando mdadm:

--create se utiliza para crear una nueva matriz

--level=nombre (ó número) especifica que nivel de RAID se usará

--raid-devices indica el número de dispositivos en la matriz

\${dispositivos[@]} se utiliza para proporcionar la lista de dispositivos

RAID 0 (Striping): Distribuye los datos entre múltiples dispositivos de almacenamiento para mejorar el rendimiento, pero no ofrece redundancia.

```
mdadm --create --level=0 ...
```

RAID 1 (Mirroring): Duplica los datos en dos o más dispositivos, proporcionando redundancia completa.

```
mdadm --create --level=1 ...
```

RAID 5 (Striping with Distributed Parity): Distribuye los datos y la paridad a través de tres o más dispositivos para proporcionar redundancia y mejorar el rendimiento.

```
mdadm --create --level=5 ...
```

RAID 6 (Striping with Double Distributed Parity): Similar a RAID 5 pero con dos bloques de paridad distribuidos para mayor redundancia frente a fallos de disco.

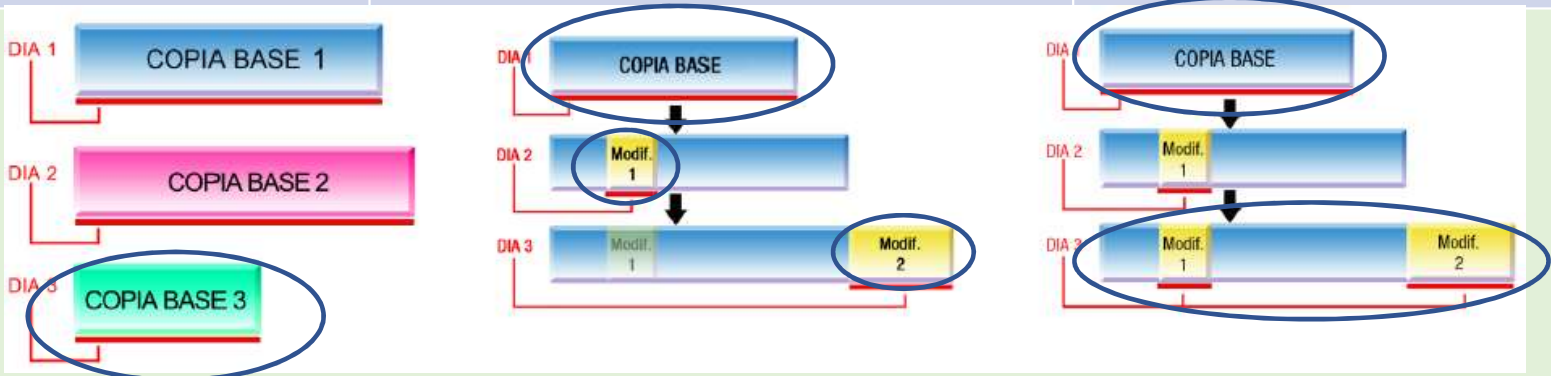
```
mdadm --create --level=6 ...
```

RAID 10 (Nested RAID 1+0): Combina RAID 1 (espejo) y RAID 0 (striping) para ofrecer redundancia y rendimiento.

```
mdadm --create --level=10 ...
```

AMPLIACIÓN – COPIAS DE SEGURIDAD - TIPOS

COMPLETA	INCREMENTAL	DIFERENCIAL
Copia simple de los datos: "copia base"	Parte de la copia base y genera un archivo con los datos se han modificado	Crea un archivo con todos los datos modificados desde la copia base
Tiempo de recuperación breve. Técnica de almacenamiento simple.	Tiempo de generación más rápido	Tiempo de generación más lento. Almacena más información.
Varias copias base implica gran cantidad de almacenamiento con información redundante	Para la recuperación se necesita la copia base y todos los archivos generados en copias de seguridad incremental anteriores	Recuperación más rápida , ya que no emplea archivos intermedios entre la copia base y la copia diferencial.



rsync:

Permite copias de seguridad de archivos y directorios en sistemas de archivos **locales o remotos**.

Tipo de copia de seguridad: **Completa**, pero puede realizar copias de seguridad **incrementales** cuando se combina con otros comandos o scripts de shell que identifican y copian solo los cambios desde la última copia de seguridad.

tar:

Empaqueta archivos y directorios en un archivo de respaldo **comprimido**.

Tipo de copia de seguridad: **Completa**. Sin embargo, puedes utilizar scripts de shell y comandos como find para realizar copias de seguridad **incrementales o diferenciales** copiando solo los archivos modificados desde la última copia de seguridad.

rsnapshot:

Utiliza **rsync y enlaces simbólicos** para realizar copias de seguridad **incrementales** y crear **instantáneas** de copias de seguridad.

Tipo de copia de seguridad: **Incremental**. Crea instantáneas de copias de seguridad incrementales utilizando enlaces simbólicos para archivos idénticos entre instantáneas.

Duplicity:

Realiza copias de seguridad en almacenamientos **remotos con cifrado** y soporte para copias de seguridad incrementales y diferenciales.

Tipo de copia de seguridad: **Incremental o diferencial**.

Utilidad de sincronización de archivos que se utiliza para **copiar y sincronizar** archivos y directorios de forma eficiente, tanto **localmente como entre sistemas remotos**. A diferencia de simplemente copiar archivos, **rsync** es capaz de **detectar diferencias entre los archivos en la fuente y en el destino, y solo copiará las partes que han cambiado**. Esto hace que sea ideal para mantener copias de seguridad actualizadas y para sincronizar grandes cantidades de datos de manera eficiente

rsync [opciones] origen destino

Algunas opciones para configurar el comportamiento de la copia de seguridad:

- a para mantener atributos
- v para mostrar la salida detallada
- r para copiar recursivamente
- z para comprimir
- P para ver el progreso de la transferencia
- e *ssh* para establecer la conexión remota, en este caso SSH

```
rsync -avz -e ssh usuario@servidor:/ruta/de/origen /ruta/de/destino
```