

Índice general

I	Apuntes de Clase	3
1.	Bloque 1	5
1.1.	Ping y SSH	5
1.2.	LVM y RAID	5
1.3.	LVM (Logical Volume Manager)	5
1.3.1.	Directorios base en Linux	5
1.4.	RAID (Redundant Array of Independent Disks)	6
1.4.1.	Ventajas	6
1.4.2.	Niveles de RAID	6
1.4.3.	Tipos de RAID	6
1.5.	Ejercicio Opcional: Configuración de RAID1 para /var	7
1.5.1.	Objetivo	7
1.5.2.	Pasos a seguir	7
1.6.	LVM (Logical Volume Manager)	7
1.6.1.	Conceptos Clave	7
1.6.2.	Visualización de LVM	8
1.6.3.	Consideraciones sobre Volumen Groups	8
1.6.4.	Creación del LVM sobre el RAID	8
1.6.5.	Movimiento de /var al RAID	8
1.7.	Resolución de Problemas	8
1.7.1.	Problema 1: Configurar el sistema de archivos en rvar	8
1.7.2.	Problema 2: Montar y trasladar /var al nuevo volumen	9
1.8.	Interpretación de lsblk	10
II	Prácticas	11

Parte I

Apuntes de Clase

Capítulo 1

Bloque 1

1.1. Ping y SSH

Sobre esta parte no decidí tomar apuntes en clase debido a que son conceptos vistos en la Asignatura de Fundamentos de Redes, por lo que consideré que no era necesario tomar apuntes de esta parte. De todas formas en la parte de Prácticas (Resolución) se comenta paso a paso lo que se hizo en esta parte.

1.2. LVM y RAID

1.3. LVM (Logical Volume Manager)

- **Discos y particiones:**
 - **sda**: Primer disco del sistema.
 - Luego pueden existir **sdb**, **sdc**, etc.
- **Consideraciones importantes:**
 - Si el directorio **/boot** se llena, el sistema podría fallar al arrancar debido a la falta de espacio disponible.
 - Se recomienda evitar el uso de **swap** en sistemas virtualizados, ya que reduce el rendimiento.

1.3.1. Directorios base en Linux

- **/boot**: Contiene los archivos de arranque del sistema.
- **/etc**: Almacena los archivos de configuración del sistema operativo.
- **/dev**: Contiene archivos especiales que representan dispositivos del sistema.
- **/mnt**: Punto de montaje temporal para sistemas de archivos.
- **/var**: Contiene datos variables del sistema, como logs y archivos temporales. Puede crecer mucho, por lo que se recomienda monitorearlo.

1.4. RAID (Redundant Array of Independent Disks)

RAID es una tecnología que permite combinar múltiples discos para mejorar la redundancia y/o el rendimiento del sistema de almacenamiento.

1.4.1. Ventajas

- Permite unir volúmenes de almacenamiento.
- Mejora el rendimiento si los discos están en buses distintos, ya que permite acceso en paralelo.

1.4.2. Niveles de RAID

- **RAID 0 (Striping):**
 - Divide los datos en bloques y los distribuye entre varios discos.
 - **Problema:** Si un disco falla, se pierde toda la información.
 - Se usa poco en la práctica debido a su baja robustez.
- **RAID 1 (Mirroring):**
 - Duplica los datos en dos o más discos.
 - Si un disco falla, el otro sigue funcionando con los mismos datos.
 - Aporta robustez al sistema.
 - **Problema:** Se paga el doble en almacenamiento.
 - Se usa frecuentemente en `/boot` para garantizar que el sistema pueda arrancar en caso de fallo de un disco.
- **RAID 5 (Paridad distribuida):**
 - Se distribuyen los datos y la paridad entre todos los discos.
 - Si un disco falla, se pueden recuperar los datos utilizando la paridad.
 - **Problema:** Puede reducir el rendimiento debido al tiempo necesario para calcular la paridad.
 - **Ventaja:** Equilibra costos, robustez y capacidad de recuperación.
 - Se puede usar un disco de repuesto que entra en acción si uno falla.

1.4.3. Tipos de RAID

- **RAID por Hardware:**
 - Utiliza un controlador RAID físico.
 - Es más eficiente y transparente para el sistema operativo.
- **RAID por Software:**
 - Administrado por el sistema operativo.
 - Puede ser modificado por el administrador, lo que representa un riesgo.
 - Requiere más recursos del sistema.

1.5. Ejercicio Opcional: Configuración de RAID1 para /var

1.5.1. Objetivo

El objetivo es proporcionar a /var un respaldo frente a fallos mediante la creación de un RAID1. Para ello, debemos montar un RAID1 y mover /var dentro de este.

1.5.2. Pasos a seguir

1. Creación de discos virtuales en la máquina virtual (MV)

- Se crean dos discos: `raid1` y `raid2`.
- Usamos `lsblk` para verificar la existencia de `sdb` y `sdc`.

2. Instalación de mdadm

- `sudo dnf provides mdadm` (para verificar qué paquete lo proporciona).
- `sudo dnf install mdadm` (para instalarlo).

3. Creación del RAID1

- `sudo mdadm -create /dev/md0 -level=1 -raid-devices=2 /dev/sdb /dev/sdc`
- Aparecerá un *warning* sobre la creación de metadatos, confirmamos con "sí".
- Para monitorear la sincronización del RAID: `watch -n 1 more /proc/mdstat`
- Luego, verificamos con `lsblk`.

4. Prueba de fallo de disco

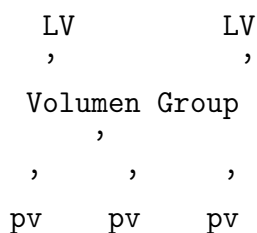
- Podemos simular la falla de un disco con: `echo 1 >/dev/sdb`
- Se observa que `md0` sigue funcionando al ser un *mirror*.

1.6. LVM (Logical Volume Manager)

1.6.1. Conceptos Clave

- **LV (Logical Volume)**
- **VG (Volume Group)**
- **PV (Physical Volume)**

La estructura básica de LVM es la siguiente:



Un **Logical Volume (LV)** puede expandirse tomando espacio libre de otros discos o estructuras.

1.6.2. Visualización de LVM

- Para ver los discos que son de tipo LVM, usamos `lsblk` en la columna de *type*.
- Para ver opciones de `pv`, `vg` y `lv`, usamos `tab-completion`.

1.6.3. Consideraciones sobre Volumen Groups

Si un **Volume Group** contiene un disco magnético, un SSD y un RAID, el acceso a los datos puede no ser homogéneo. Por ello, el volumen lógico debe montarse sobre el RAID.

1.6.4. Creación del LVM sobre el RAID

1. Convertir el RAID en un Physical Volume: `pvcreate /dev/md0`
2. Crear un Volume Group llamado `raid1`: `vgcreate raid1 /dev/md0`
3. Crear un Logical Volume de 10GB dentro del Volume Group: `lvcreate -L 10G -n rvar raid1`

1.6.5. Movimiento de `/var` al RAID

Después de crear el volumen lógico, debemos mover `/var` dentro del RAID:

- El nombre del volumen dentro del RAID es `/dev/raid1/rvar`.
- También puede aparecer como `/dev/mapper/raid1-rvar`, ya que son sinónimos creados mediante enlaces simbólicos.

1.7. Resolución de Problemas

1.7.1. Problema 1: Configurar el sistema de archivos en `rvar`

Actualmente, el volumen lógico `rvar` está vacío y sin un sistema de archivos. Para solucionarlo, seguimos estos pasos:

1. **Seleccionar un sistema de archivos:** Los sistemas de archivos recomendados son `ext4` y `xfs`, ya que son transaccionales y previenen la corrupción de datos en caso de fallos.
2. **Verificar los sistemas de archivos soportados:** Ejecutamos el siguiente comando para listar los sistemas disponibles en el kernel:

```
ls /lib/modules/$(uname -r)/kernel/fs
```

3. **Formatear el volumen lógico con `ext4`:**

```
mkfs.ext4 /dev/mapper/raid1-rvar
```


1.7.2. Problema 2: Montar y trasladar /var al nuevo volumen

El volumen lógico **rvar** debe ser montado en el sistema y debemos trasladar **/var** sin perder datos. Para ello:

1. **Montar el volumen lógico:** Primero, montamos **rvar** en un directorio temporal:

```
mount /dev/mapper/raid1-rvar /mnt/
```

Podemos verificar con:

```
mount
```

Al revisar **/mnt/**, aparecerá el directorio **lost+found**, indicando que el sistema de archivos está activo.

2. **Cambiar a modo mantenimiento:** Para evitar la pérdida de datos al copiar **/var**, debemos entrar en el **runlevel1**, también conocido como **modo mantenimiento**. Esto se hace con:

```
systemctl isolate runlevel1.target
```

Podemos confirmar el estado con:

```
systemctl status
```

3. **Copiar los datos de /var a /mnt/:** Copiamos todo el contenido de **/var** manteniendo atributos con:

```
cp -a /var/* /mnt/
```

4. **Crear un respaldo de /var:** Antes de reemplazar **/var**, hacemos una copia de seguridad por si algo falla:

```
mv /var /var_old
```

5. **Desmontar /mnt y montar el nuevo /var:**

```
umount /mnt
mkdir /var
mount /dev/mapper/raid1-rvar /var
```

Podemos verificar con:

```
df -h
```

6. **Hacer el montaje permanente en `/etc/fstab`:** Si reiniciamos ahora, la configuración se perdería. Para evitarlo, editamos el archivo `/etc/fstab` y agregamos la siguiente línea:

```
/dev/mapper/raid1-rvar    /var    ext4    defaults    0 0
```

7. **Probar la configuración antes de reiniciar:**

```
mount -a
systemctl daemon-reload
mount
```

Si todo está correcto, reiniciamos el sistema.

1.8. Interpretación de `lsblk`

El comando `lsblk` nos permite visualizar la estructura de almacenamiento del sistema. Es importante identificar:

- **Discos físicos y particiones:** Aparecen como `sda`, `sdb`, `sdc`, etc.
- **Volúmenes lógicos:** Se muestran bajo `/dev/mapper/`.
- **SR0:** Indica la unidad de CD-ROM.

Parte II

Prácticas

