

Práctica 5: Administración de Sistemas de Ficheros

Objetivos

Para mantener e instalar servidores es fundamental disponer de unos conocimientos sólidos sobre la administración del sistemas de ficheros y almacenamiento. Normalmente las actividades típicas en este área incluyen crear particiones, volúmenes lógicos, creación y mantenimiento de sistemas de ficheros de diversos tipos o la automatización del acceso (*montado*) de dispositivos, entre otras. En esta práctica aprenderemos las rutinas básicas de gestión de los sistemas de ficheros y almacenamiento.

Contenidos

Preparando el entorno...

Administración de discos y particiones

Consultando el estado del sistema...

Particiones

Para saber más...

Sistemas de Ficheros

Sistema de ficheros tipo swap

Para saber más...

Gestión de volúmenes lógicos (LVM)

Physical volume (PV)

Volume group (VG)

Logical Volume (LV)

Para saber más...

Sistemas de Ficheros Encriptados

Gestión de sistemas de ficheros

Para saber más...

Test Práctica 1 - 2 - 4 - 5

Preparando el entorno...

En esta práctica necesitaremos una máquina virtual con el sistema CentOS 7 instalado, con las siguientes características:

- Hacer un clon enlazado de la instalación base disponible en el laboratorio, ver práctica 3.
- Una vez clonada la máquina CentOS 7, en las propiedades del servidor (sección almacenamiento) añadir 2 discos de 500MB cada uno en el bus SATA para realizar diferentes pruebas. En cada caso se creará un disco nuevo, de tipo VDI y asignación dinámica de espacio. La Figura 1 muestra la disposición final de discos que debe usarse en la práctica.

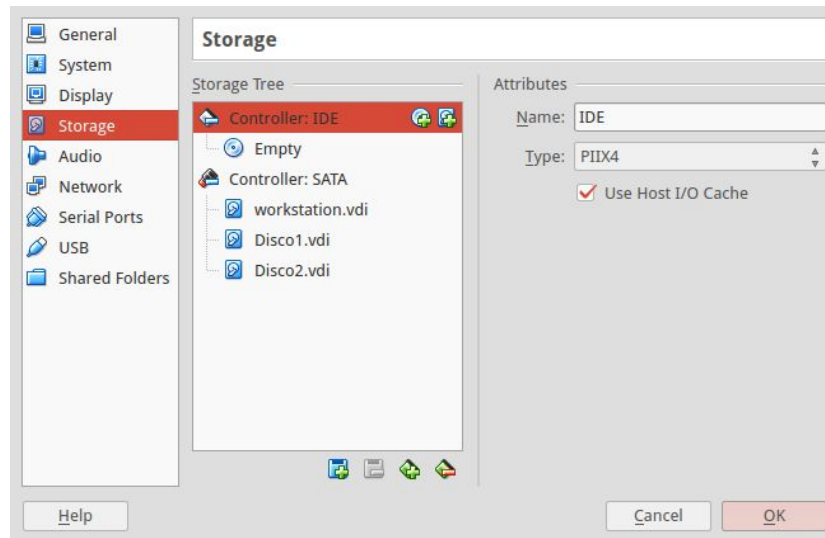


Figura 1. Disposición de los elementos de almacenamiento para la Práctica 5

Administración de discos y particiones

Probablemente la forma más fácil de crear particiones, volúmenes lógicos o configuraciones RAID es durante la instalación del sistema, aunque esto no es siempre posible. En esta sección estudiaremos cómo particionar y preparar discos para usarlos en el sistema.

Consultando el estado del sistema...

En primer lugar vamos a estudiar las utilidades que nos permiten comprobar el estado de las particiones, discos y sistemas de ficheros que están disponibles en el servidor; estas herramientas son principalmente `lsblk`, `df`, `mount`.

Ejercicio 1. *Lista de los dispositivos en modo bloque.* Consultar la salida del comando `lsblk`, e identificar los dispositivos, su tipo, el punto de acceso (punto de *montaje*) así como el tamaño usado y disponible de cada sistema de ficheros.

Ejercicio 2. *Lista de los sistemas de ficheros.* Para consultar los sistemas de fichero en el sistema disponemos de dos utilidades principalmente:

- Comando `df`, que muestra el espacio disponible. Ejecutar el comando e identificar los dispositivos, su punto de acceso (punto de *montaje*) así como el tamaño usado y disponible de cada sistema de ficheros. Comprobar varias opciones del comando, `-h` puede resultar especialmente útil.
- El comando `mount` sirve para *montar* un sistema de ficheros en un punto del árbol de directorios del sistema. Cuando se ejecuta sin argumentos muestra los sistemas de ficheros montados (tipo, dispositivo, opciones y punto de montaje). Estudiar los sistemas de ficheros disponibles en el sistema.

Nota: El comando `df` con la opción `-a` muestra todos los sistemas de ficheros, incluidos los

pseudo-sistemas de ficheros, p. ej. proc or sysfs.

Particiones

Tal como vimos en la práctica 2, los primeros bloques del disco guardan información específica del disco, como su identificador, código para el arranque del sistema donde se instalan las primeras partes del cargador del sistema operativo; y la tabla de particiones. La tabla de particiones define secciones lógicas del disco (particiones) que contendrán sistemas de ficheros.

Hay dos aplicaciones que habitualmente se utilizan para particionar un disco: `fdisk` y `parted`. Ambas permiten la edición de particiones por consola, aunque `parted` ofrece alguna operación adicional (p.ej. dar formato o cambiar tamaño) y dispone también de interfaz gráfico.

Ejercicio 1. Los discos nuevos en el sistema se encuentran en los dispositivos `/dev/sdb` y `/dev/sdc` (verificar con `lsblk`). Abrir uno de ellos con `fdisk` (p. ej. `fdisk /dev/sdb`):

- Observar el mensaje inicial sobre la etiqueta del disco
- Observar las opciones disponibles (comando `m`)
- Guardar los cambios (`w`) para escribir la tabla de particiones.

Ejercicio 2. Crear particiones. El comando que usaremos es (`n`), hay dos tipos de particiones primarias (4 máximo) y extendidas. Crear tres particiones primarias: dos de 200MB y una de 100MB, en ese orden de forma que las particiones de 200MB sean la 1 y 2; y la de 100MB la 3. Comprobar el estado de la tabla de particiones con el comando (`p`). Finalmente escribir los cambios con (`w`).

Ejercicio 3. Repetir el esquema de particionado anterior en el otro disco. Durante el proceso probar el borrado de una partición con el comando (`d`).

Ejercicio 4. Cada partición tiene un identificador específico según el uso al que se destinará la partición. El tipo de particiones disponibles puede consultarse con (`L`), y el tipo cambiarse con (`t`). A modo de ejemplo asignar el tipo swap a la partición de 100MB (`/dev/sdb3`).

Para saber más...

- Estudiar la utilidad `parted`. Repetir alguno de los ejercicios anteriores con `parted`
- Estudiar el formato y la información de la MBR.
- El formato MBR es de 1980 e impone algunas restricciones importantes (ej. 4 particiones). El formato GPT (GUID Partition Table) desarrollado como parte de UEFI; estudiar el formato y sus herramientas asociadas `gptfdisk`.

Sistemas de Ficheros

En el sistema hay diversos tipos de sistemas de ficheros, que pueden clasificarse en: sistemas de ficheros basados en disco (ej. `ext3`, `vfat`, `brtfs`), basados en memoria (ej. `proc`, `shm`, `ramdisk`) y basados

en red (ej. ntfs). En esta sección nos centraremos en el primer tipo.

Sistema de ficheros tipo swap

Ejercicio 1. Para formatear una partición como swap se usa el comando `mkswap`. Crear un sistema de ficheros swap en la partición que creamos a tal efecto en el ejercicio anterior (`/dev/sdb3`).

Ejercicio 2. El número de particiones swap activas y su uso se puede comprobar con la opción `-s` del comando `swapon`. Determinar el número de particiones swap activas. Activar, comando `swapon`, la región de intercambio y comprobarlo. El efecto en la cantidad de memoria virtual del sistema se puede verificar con el comando `free`.

Ejercicio 3. Para desactivar el área de intercambio usar `swapoff`. Desactivar la región anterior y comprobar la disminución de memoria virtual disponible.

Sistema de ficheros tipo ext

El tipo de sistemas de ficheros por defecto en Linux es extended (ext) en sus tres versiones 2, 3 y 4. De forma muy resumida: ext3 añade journaling al sistema ext2; ext4 introduce importantes mejoras de rendimiento y fiabilidad respecto a ext3. Normalmente la compatibilidad entre todos es muy alta, y resulta fácil pasar de un formato a otro.

Ejercicio 1. Para crear un sistema de ficheros usaremos `mkfs`, especificando el tipo de sistema de ficheros con la opción `-t`. Crear un sistema de ficheros tipo ext4 en una de las particiones (p.ej. `/dev/sdb1`) de 200MB disponibles. **Nota:** Alternativamente se puede usar directamente `mkfs.ext4`.

Ejercicio 2. Una vez creado el sistema de ficheros se puede ajustar algunas de sus características (ej. frecuencia de comprobación, etiqueta del sistema...):

- Usando el comando `tune2fs` consultar las características del sistema de ficheros anterior (opción `-l`). Determinar la etiqueta, el tamaño de bloque y la cantidad reservada para el super usuario.
- Usando el comando `tune2fs` añadir la etiqueta “Disco de Datos” a uno de los sistemas creados. Consultar la página de manual para determinar la opción adecuada.
- Crear un sistema de ficheros tipo ext2 en otra de las particiones de 200MB (`/dev/sdb2`). Añadir journaling con el comando `tune2fs` y la opción adecuada.

Nota: Casi todas las utilidades que referencian discos o particiones permiten usar el nombre del volumen (`/dev/sda1`), el identificador único (UUID=`ce69ee7a-222a-4f62-bad1-116fb5f74d7d`) o la etiqueta (LABEL = “Disco de Datos”). **Comprobar estos parámetros con la utilidad `blkid`.**

Ejercicio 3. La estructura de bloques, copias del superbloque y el resto de información adicional se puede obtener con el comando `dumpe2fs`. Comprobar el funcionamiento de este comando con los sistemas de ficheros anteriores.

Ejercicio 4. La integridad del sistema de ficheros se chequea con el comando fsck:

- Comprobar algunos de los sistemas anteriores con la utilidad fsck.
- Destruir el superbloque escribiendo ceros directamente en la primera copia y repararlo con fsck. Para borrar el superbloque podemos escribir zeros directamente con el comando dd.

Listado 1. Comando para escribir 0 en el primer bloque de 1024 bytes

```
dd if=/dev/zero of=<partición> seek=1 bs=1024 count=1
```

Nota: en este comando el tamaño de bloque (bs) es el tamaño de bloque del sistema de ficheros, ¿cómo se puede determinar el valor adecuado?. Además se empieza a escribir en el bloque 1, ¿por qué?

Opcional. Destruir el superbloque y restaurar una copia manualmente con el dd.

Ejercicio 5. Para acceder a los sistemas de ficheros que hemos creado, deben *montarse* primero en un punto del árbol de directorios con el comando mount. En su forma más sencilla especificar únicamente el tipo de sistema de ficheros, el dispositivo y el directorio:

Listado 2. Comando para montar un dispositivo en bloques (/dev/sde3) en un directorio (/root/mi_disco)

```
mount -t ext4 /dev/sde3 /root/mi_disco
```

Crear puntos de montaje para los discos formateados y montarlos. Comprobar que efectivamente el sistema de ficheros es accesible (cd), que está montado (mount) y el espacio disponible (df).

Para saber más...

- Estudiar el sistema de ficheros btrfs, y sus comandos asociados btrfs y mkfs.btrfs. Btrfs es el *supuestamente* futuro sistema de ficheros en Linux, ofrece entre otros snapshots, gestión de volúmenes... similar a ZFS
- RedHat7 por defecto selecciona el sistema de ficheros XFS, muy similar a ext4, pero con soporte a ficheros y sistema de ficheros mayores. Estudiar sus propiedades y las herramientas equivalentes a las vistas para ext4 (mkfs.ext4 - mkfs.xfs; e2fsck - xfs_repair; resize2fs - xfs_growfs; tune2fs - xfs_admin...).
- [Revisar la documentación oficial de RedHat](#) donde se ofrece una comparativa de los diferentes sistemas de ficheros soportados.

Gestión de volúmenes lógicos (LVM)

LVM añade una capa de abstracción por encima de la estructura de discos y particiones explicada en las secciones anteriores. LVM permite definir volúmenes lógicos, *logical volumes* (LV), que constituyen una

partición lógica que puede agregar varias particiones físicas que pueden incluso encontrarse en diferentes discos. Esta separación permite, entre otras cosas, cambiar dinámicamente el tamaño de un LV. Toda la gestión se estructura en 3 conceptos principales:

- **Physical volume (PV)**, una partición con el tipo LVM (fdisk -L).
- **Volume group (VG)**, un conjunto de PVs
- **Logical volume (LV)**, una parte de un VG, que será formateado y montado para su uso

Physical volume (PV)

Ejercicio 1. Inicializar las particiones de 200MB de los dos de los discos (/dev/sdb1, /dev/sdb2 y /dev/sdc1 y /dev/sdc2) de tipo LVM (8e) con el comando fdisk. Aunque informativo, resulta conveniente fijar el tipo al uso que finalmente se destinará el disco.

Ejercicio 2. Crea un PV en cada una de las particiones anteriores con el comando pvcreate.

Ejercicio 3. Listar los PV en el sistema con el comando pvs, y consultar las propiedades de alguno de ellos con el comando pvdisplay.

Volume group (VG)

Ejercicio 1. Creación de un volume group. Con el comando vgcreate crear un VG de nombre practica5, que incluya las particiones /dev/sdb1 y /dev/sdc1.

Ejercicio 2. Consulta de los grupos de volúmenes. Comprobar las características del volumen creado con vgdisplay y listar los VGs del sistema con el comando vgs.

Ejercicio 3. Activación manual. Normalmente LVM tiene asociado un servicio para actualizar los cambios en el sistema PV/VG/LV. Manualmente se puede explorar las unidades del sistema para descubrir los VGs disponibles con el comando vgscan. Probar su uso.

Ejercicio 4. Extender un VG existente. Mediante el comando vgextend añadir el PV /dev/sdb2 creado en el apartado anterior al VG. Comprobar el resultado con los comandos vgs, vgdisplay, pvdisplay. El VG debe tener asignados ahora 3 (/dev/sdb1, /dev/sdb2 y /dev/sdc1) de los 4 PV.

Logical Volume (LV)

Ejercicio 1. Creación de un volumen lógico. Crear un LV lvcreate con nombre vol1 (opción -n) en el VG practica5 y el tamaño (opción -L) debe ser 400M.

Ejercicio 2. Consulta de los volúmenes lógicos. Comprobar las características del volumen lógico con lvdisplay (usa como argumento el VG) y listar todos los volúmenes disponibles con lvs.

Ejercicio 3. Uso de un volumen lógico. Crear un sistema de ficheros (mkfs) en el LV vol1. El dispositivo (/dev) que utiliza se puede obtener de la salida del comando lvdisplay (LV Path). Fijarse en la estructura creada para los dispositivos correspondientes a un LV. Montar el nuevo sistema de ficheros y

comprobar el tamaño disponible.

Ejercicio 4. *Añadir más capacidad a un volumen lógico existente.* Gracias a la abstracción que ofrece LVM es posible añadir más espacio de disco a un sistema de ficheros existente, haciéndolo crecer de forma efectiva. Para simular este procedimiento en primer lugar crearemos un sistema de ficheros que posteriormente extenderemos:

- Crear un volumen vol2 con el resto del VG practica5, usar la opción `-l 100%FREE`
- Hacer un sistema de ficheros ext4 en practica5/vol2. Montarlo y comprobar que se ha creado correctamente.

Vamos a extender el sistema de ficheros que acabamos de crear con los 200M de la partición `/dev/sdc2` que tenemos disponible:

- En primer lugar desmontar vol2 con el comando `umount`.
- Añadir el PV `/dev/sdc2` al VG `practica5` con el comando `vgextend`. Comprobar que el resultado es exitoso.
- Añadir el espacio adicional en practica5 al LV vol2, con el comando `lvextend` y la opción `-l +100%FREE`.
- Finalmente hay que extender el sistema de ficheros, primero verificar el sistema con `e2fsck` y extenderlo con `resize2fs`. Montar el nuevo sistema de ficheros y comprobar el resultado.

Para saber más...

- Probar las herramientas gráficas para la gestión de LV. Instalar con `yum system-config-lvm`
- Estudiar el comando `lvsnapshot` para hacer copias de un LV.
- Estudiar las configuraciones en stripe, opciones `-i` de `lvcreate` para optimizar lecturas/escrituras.
- Estudiar las extensiones distribuidas de LVM2, cLVM.

Sistemas de Ficheros Encriptados

Por motivos de seguridad (especialmente para dispositivos extraíbles o para almacenar datos importantes) se puede encriptar el sistema de ficheros. En linux se usa LUKS (Linux Unified Key Setup) mediante la utilidad `cryptsetup`.

Ejercicio 1. El primer paso es preparar la partición que queremos guardar encriptada. Usar una de las particiones libres (`/dev/sdc3`) que tenemos disponibles. El comando es: `cryptsetup luksFormat <partición>`.

Ejercicio 2. Podemos comprobar el estado de la partición (algoritmo de encriptación, identificador del volumen...) con `cryptsetup luksDump`.

Ejercicio 3. Para poder acceder al sistema de ficheros debe habilitarse introduciendo la clave. El comando que usaremos es `cryptsetup luksOpen`. Este comando necesita el dispositivo encriptado y

un nombre que se asignará al dispositivo que usaremos para montar el sistema de ficheros. Una vez disponible el dispositivo estará en `/dev/mapper/<nombre>`.

Ejercicio 5. Dar formato y montar el sistema de ficheros encriptado como si se tratase de un dispositivo cualquiera.

Ejercicio 6. Una vez que terminamos, desmontar el sistema de ficheros y cerrarlo con `cryptsetup luksClose`.

Gestión de sistemas de ficheros

Ejercicio 1. Los sistemas de ficheros que se montan en el arranque están definidos en el fichero `/etc/fstab`. Consultar e identificar las secciones principales de este fichero (dispositivo, punto de montaje, tipo de sistema de ficheros, opciones, dump - backup, fsck - 0, 1, 2).

Ejercicio 2. Añadir los LVs que hemos creado en `fstab`. Se puede hacer por nombre (`/dev/mapper/nombre_vg-nombre_lv`) o por identificador de sistema de ficheros UUID (obtenerlo con `dumpe2fs`, o mediante el comando `blkid`).

Ejercicio 3. Reiniciar el sistema y comprobar que se montan los nuevos sistemas de forma automática.

Ejercicio 4. Mount tiene muchas opciones, comprobar las opciones en `mount(8)` las generales (`rw`, `ro`, `auto`, `noatime`, `noexec`, `nosuid`, `remount`) del sistema ext. Estas opciones se pueden especificar en `fstab` o en la línea de comandos con la opción `-o` (ej. `mount -o remount`). Consultar estas opciones.

Para saber más...

- Se pueden crear sistemas de ficheros en un archivo de forma que se tratan como un dispositivo en bloques (p.ej. crear un fichero como área de intercambio). Se puede asignar un fichero a un dispositivo en bloques con el comando `losetup`. *Opcional:* Probar a crear un sistema de ficheros encriptado en un fichero.
- Se puede programar el montado automático de sistemas de ficheros cuando se accede al punto de montaje. Estudiar el funcionamiento del Automonter.
- Los discos RAID ofrecen algunas ventajas respecto a fiabilidad y/o rendimiento. Linux ofrece una implementación software mediante el comando `md`, estudiar su uso.
- Se pueden copiar particiones completas (backup) usando el comando `dd` y una herramienta de compresión (`bzip`, `gzip`). Estudiar esta estrategia.
- Hay herramientas específicas para esta tarea ej. Clonezilla, Partimage, Gparted. Estudiar alguna de ellas.

Test Prácticas 1 - 2 - 4 - 5

Supongamos que hemos perdido la contraseña de superusuario y necesitamos volver a fijarla. El procedimiento general podría ser:

- Modificar la sentencia de arranque del sistema para que inicie en modo *single-user*, pero sin que se solicite la contraseña.

- El sistema / puede estar montado como sólo lectura (ro). *Re-montarlo* en modo lectura-escritura (rw) con las opciones adecuadas del comando mount.
- Eliminar la contraseña de root modificando el archivo adecuado
- Reiniciar el sistema en modo consola, entrar como root (no tendrá contraseña) y asignar una contraseña nueva a la cuenta de superusuario.