

ALMACENAMIENTO

JOSÉ DOMINGO MUÑOZ

IES GONZALO NAZARENO

ENERO 2023



SISTEMAS DE FICHEROS



- Es un conjunto de datos almacenados en un dispositivo de almacenamiento.
- Un fichero posee un nombre y metadatos.
- El tipo de datos contenidos está definido por su formato y en ocasiones por su extensión.
- Normalmente el formato está definido en la cabecera (**magic number**).
- Un fichero solo con datos ASCII o UTF8 se conoce como texto plano (plain text).
- Los ficheros con datos reciben el nombre de ficheros regulares.

file fichero.pdf

<https://asecuritysite.com//forensics/magic>



Aparte de los ficheros regulares, en UNIX **“todo es un fichero”**:

- Enlaces
- sockets
- Dispositivos (En /dev)
- Ficheros virtuales (En /proc o /sys)
- Ficheros virtuales en memoria (Tipo tmpfs o ramfs)



- Los directorios son contenedores de ficheros y se pueden anidar (subdirectorios).
- El directorio principal es el directorio raíz o /.
- Los directorios se organizan formando un árbol a partir del directorio raíz.
- Los ficheros se definen de forma única por el nombre que incluye su ruta completa desde el directorio raíz, p.ej. /usr/share/doc/apt/copyright.
- La estructura de los directorios es estricta y definida en el **Linux Filesystem Hierarchy Standard (FHS)** de la Linux Foundation (<https://refspecs.linuxfoundation.org/fhs.shtml>)
- Las diferentes distros de GNU/Linux deben seguir la FHS



- Los dispositivos son ficheros especiales ubicados en **/dev**.
- **Dispositivos de caracteres** (character devices)
- **Dispositivos de bloques** (block devices)
 - ▶ E/S en bloques de datos
 - ▶ Uso de buffer
 - ▶ Usados para los dispositivos de almacenamiento
- Discos duros, disquetes, CDs, DVDs, particiones, volúmenes lógicos, RAID, etc.



- Capa del kernel que proporciona una interfaz única a usuarios y programas.
- Implementa un único árbol aunque esté formado por varios dispositivos de bloques y/o tipos de sistemas de fichero.
- syscalls: `open(2)`, `stat(2)`, `read(2)`, `write(2)` y `chmod(2)`
- Montar y desmontar



- Controla cómo se almacenan y obtienen los datos.
- Capa lógica.
 - ▶ Responsable de la interacción con los usuarios y programas.
 - ▶ API.
 - ▶ Modelo de seguridad.
- Implementación: Enlaza la capa lógica con los dispositivos de almacenamiento
- Los dispositivos de bloques se formatean con un sistema de ficheros.



- **DAS** (Direct Attached Storage): Dispositivo de almacenamiento conectados directamente al ordenador o servidor.
- **NAS** (Network Attached Storage): Se comparte por red almacenamiento, normalmente sistema de ficheros.
- **SAN** (Storage Area Network): En una red de almacenamiento se comparte dispositivos de bloques.
- **Cloud** (Object Storage): Almacenamiento en la nube con características de cloud computing.



COMPARATIVA DAS, NAS Y SAN

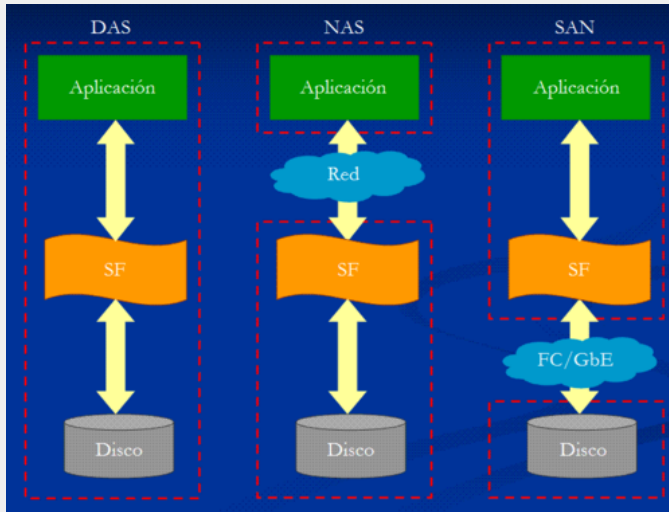


Figura 1: Comparativa DAS, NAS y SAN

- ext2/3/4
- xfs
- jfs
- reiserfs
- vfat
- zfs
- btrfs



- Sistema de ficheros por defecto en linux
- Características principales:
 - ▶ Compatible con ext2/3
 - ▶ Utiliza journaling
 - ▶ Hasta 1 EiB con ficheros de 16 TiB
 - ▶ Mejoras en el rendimiento frente a ext2/3
 - ▶ Herramientas del espacio de usuario: `e2fsprogs: mkfs.ext4 fsck.ext4 tune2fs resize2fs dumpe2fs debugfs ...`
- No supone una mejora radical frente a ext3



- Desarrollado originalmente por SGI
- Utilizado por defecto en RHEL 7
- Hasta 8 EiB
- Utiliza journaling
- Permite redimensionado en “caliente”
- Presume de tener mejor rendimiento que ext4
- Herramientas del espacio de usuario: xfsprogs: mkfs.xfs fsck.xfs xfs_db xfs_growfs xfs_info ...



- Sistema de ficheros de MS-DOS/Windows
- No guarda información de propietarios: sistema monousuario
- Enormes limitaciones
- Utilizado masivamente en dispositivos extraíbles
- VFS permite su utilización “transparente” en linux
- Herramientas del espacio de usuario: dosfstools: mkfs.vfat fsck.vfat dosfslabel



Copy on Write (CoW)

- Se descomponen los datos a almacenar en diferentes bloques.
- Al crear la copia solo se crea un nuevo puntero que apunta al conjunto de datos originales.
- Cuando los datos copiados se modifican, se van creando nuevos bloques con las modificaciones.

Deduplicación

- Utiliza las propiedades CoW de un sistema de ficheros.
- Identifica bloques idénticos y los reorganiza con CoW.
- Permite aprovechar la característica CoW sin intervención del usuario.



FUNCIONALIDADES AVANZADAS (II)

Cifrado

- Permite el cifrado de ficheros al vuelo sin utilizar software adicional.

Compresión

- Almacena los ficheros comprimidos para optimizar el uso del espacio.

Gestión de volúmenes

- Equivalente a LVM
- Permite gestionar volúmenes y sistemas de ficheros de forma independiente de los dispositivos de bloques físicos.
- Un dispositivo de bloques físico puede contener varios volúmenes o un volumen puede estar distribuido en varios dispositivos de bloques físicos.



Instantáneas (snapshots)

- Utiliza instantáneas de forma nativa.

Sumas de comprobación (checksums)

- Utilizadas para verificar la integridad de los ficheros.

Redundancia (RAID)

- RAID software nativo en el sistema de ficheros.
- Sin necesidad de usar mdadm.



COMPARATIVA

	CoW	Dedup	Cifrado	Compr	Vols	Snapshot	checksum	RAID
ext4	✗	✗	✓	✗	✗	✗	✗	✗
jfs	✗	✗	✓	✗	✗	✗	✗	✗
xfs	✓	✓	✗	✗	✗	✗	✗	✗
zfs	✓	✓	✓	✓	✓	✓	✓	✓
btrfs	✓	✓	✗	✓	✓	✓	✓	✓

https://en.wikipedia.org/wiki/Comparison_of_file_systems

Figura 2: Comparativa de Sistemas de Ficheros



- Sistema de ficheros avanzado desarrollado por Sun Microsystems. Se incluye en OpenSolaris.
- 2010: Oracle compra Sun Microsystems y abandona OpenSolaris. ZFS vuelve a ser software cerrado.
- Se crea Illumos (2010)(fork libre de OpenSolaris) y OpenZFS (2013) (fork libre de ZFS).
- 2015: Aparece ZFS on Linux, versión de OpenZFS para linux.



- Es un sistema completo de almacenamiento que no requiere otras herramientas.
- Gestiona los dispositivos de bloques directamente.
- Incluye su propia implementación de RAID, CoW, deduplicación, instantáneas, compresión, cifrado, soporte nativo de nfs, cifs o iscsi, ...
- Siempre consistente sin necesidad de chequeos.
- Se autorepara de forma continua.
- Muy escalable.
- Exigente en recursos.



- Es un sistema completo de almacenamiento que no requiere otras herramientas.
- Licencia GPL y completamente integrado en el kernel linux.
- Hace años que se considera estable, aunque aún le falten algunas características de ZFS.
- Gestiona los dispositivos de bloques directamente.
- Incluye su propia implementación de RAID, CoW, deduplicación, instantáneas, compresión, cifrado, ...
- Autoreparación
- No está extendido su uso como cabría esperar.



LVM



LVM es una implementación de un gestor de volúmenes lógicos para el núcleo Linux:

- Redimensionado de grupos lógicos.
- Redimensionado de volúmenes lógicos.
- Instantáneas de sólo lectura (LVM2 ofrece lectura y escritura).
- RAIDo de volúmenes lógicos.



Un LVM se descompone en tres partes:

- **Volúmenes físicos (PV):** Son los dispositivos de bloque que vamos a usar.
- **Grupos de volúmenes (VG):** Nos permiten agrupar los volúmenes físicos para crear nuestros volúmenes lógicos.
- **Volúmenes lógicos (LV):** Es un dispositivos de bloque que podemos formatear



TALLER 1: GESTIÓN DE POOL DE ALMACENAMIENTO LÓGICO EN KVM/LIBVIRT

- Cuando estudiamos virtualización Linux utilizamos pool de almacenamientos de tipo **dir** y los volúmenes eran ficheros **qcow2**.
- En este taller vamos a crear un pool de almacenamiento de tipo lógico, donde los volúmenes serán volúmenes lógicos creados en un grupo de volúmenes.
- Los ficheros del disco de la máquina virtual se guardará en el volumen lógico.
- **¿Qué ventajas puede tener este tipo de almacenamiento en una máquina virtual?**



RAID



- RAID es un sistema que aumenta la fiabilidad de los datos almacenados en discos utilizando mecanismos de redundancia.
- RAID hace dos cosas principalmente:
 - ▶ **Duplicar (mirroring)** los datos en varios discos, reduciendo el riesgo asociado al fallo de un disco.
 - ▶ **Mejorar el rendimiento dividiendo (stripping)** los datos en varios discos, que trabajan simultáneamente con un flujo único de datos.



- **Hardware:** Está implementado completamente dentro de la controladora de disco (controladora RAID), mediante hardware y firmware especializado. Una controladora RAID hardware debe presentar al sistema operativo los discos como un único dispositivo de almacenamiento.
- **Software:** Lo implementa mediante software el sistema operativo de forma independiente de la controladora de disco.
- **BIOS:** Está implementado parcialmente dentro de la controladora de disco, pero utilizan controladores de software específicos para poder comunicarse adecuadamente con el sistema operativo.



- **Paridad:** Los datos de paridad se utilizan para conseguir redundancia de los datos. Si una unidad falla, es posible recuperar los datos combinando los datos de las otras unidades y los de paridad (operaciones XOR).
- **Disco de reserva:** Disco que se pone en funcionamiento automáticamente al fallar un disco que esté en RAID. Utilizando un **hot spare** se reduce mucho el tiempo de recuperación de los datos.



NIVELES DE RAID

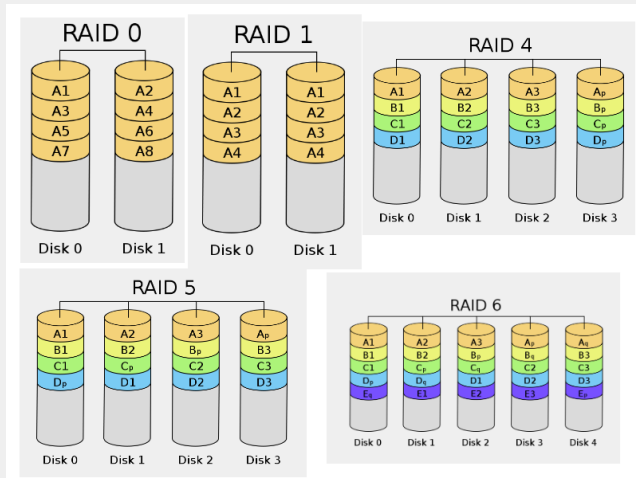


Figura 3: Niveles de RAID



NAS



El **almacenamiento conectado en red**, Network Attached Storage (**NAS**), es una tecnología de almacenamiento dedicada a compartir la capacidad de almacenamiento de un servidor con máquinas clientes a través de una red (normalmente TCP/IP).

- Protocolos usados: NFS, SMB/CIFS, ...
- Se comparte sistemas de ficheros completos.
- Normalmente para realizar copias de seguridad y compartir ficheros.



SAN



Una **red de área de almacenamiento**, en inglés Storage Area Network (**SAN**), es una red de almacenamiento integral.

- Red dedicada de almacenamiento que proporciona **dispositivos de bloques** a los servidores.
- Los elementos típicos de una SAN son:
 - ▶ Red dedicada alta velocidad (cobre o fibra óptica)
 - ▶ Equipos o servidores que proporcionan el almacenamiento
 - ▶ Servidores que utilizan los dispositivos de bloques
- Los protocolos más utilizados son iSCSI y Fibre Channel Protocol (FCP).



ESQUEMA DE SAN

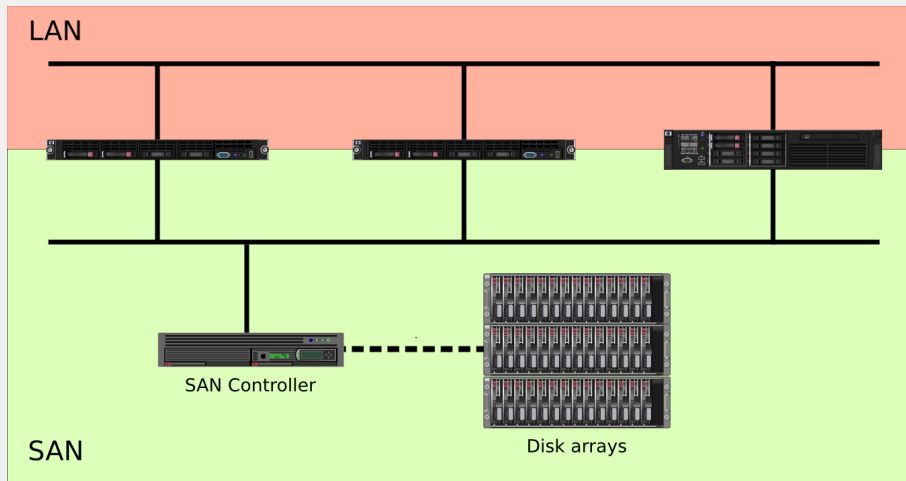


Figura 4: Esquema de SAN



-



- **Unidad lógica (LUN):** Dispositivo de bloques a compartir por el servidor iSCSI.
- **Target:** Recurso a compartir desde el servidor. Un target incluye uno o varios LUN. **Initiator:** Cliente iSCSI.
- Multipath
- **IQN** es el formato más extendido para la descripción de los recursos. Ejemplo: **iqn.2020-01.org.gonzalonazareno:sdb4**
- **iSNS:** Protocolo que permite gestionar recursos iSCSI como si fueran Fibre Channel.



- iSCSI tiene soporte en la mayoría de sistemas operativos.
- En Linux usamos **open-iscsi** como initiator.
- Existen varias opciones en Linux para el servidor iSCSI:
 - ▶ Linux-IO (LIO)
 - ▶ **tgt**
 - ▶ scst
 - ▶ istgt



SISTEMAS DE FICHEROS EN CLÚSTER



ALMACENAMIENTO DE OBJETOS EN LA NUBE

