## Act: Sistema de fitxers

**5.1 – Muntatge d’un disc extern per fer un back up**

En aquest laboratori simularem que volem muntar un disc dur extern en el nostre servidor per a realitzar còpies de seguretat de les nostres dades. Crearem una partició en el disc dur i la formatejarem amb el sistema de fitxers ext4.

Primer clonarem la nostra màquina i li crearem un disc dur de 20gb.

Una captura de pantalla de una red social

Descripción generada automáticamente

Ara executem la màquina i amb el root utilitzem la comanda lsblk per a identificar el disc dur extern.

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

Utilitzem la comanda fdisk per a crear una nova partició en el disc dur extern i crearem una nova taula de particions en el disc dur extern. Aquesta taula serà de tipus msdos  
  
Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamenteI ara crearem una nova partició primària en el disc dur externtal que així:

I ara comprovem que la partició s’ha creat correctament amb la comanda lsblk

Texto

Descripción generada automáticamente con confianza baja  
  
Formatejem la partició amb el sistema de fitxers ext4:

Texto

Descripción generada automáticamente

Montem la partició en un directori del sistema de fitxers i comprovem que la partició s'ha muntat correctament amb lsblk o df.

Imagen que contiene texto, marcador

Descripción generada automáticamente  
Còpiem totes les dades del directori /home a la nova partició i comparem les dades del directori /home amb les dades de la partició:

Texto

Descripción generada automáticamente

I per últim desmuntem la partició per a poder treure el disc dur extern.

Texto

Descripción generada automáticamente

**5.2 – Migració de directoris a particions diferents**

En aquest laboratori assumirem que volem reorganitzar els directoris del nostre sistema de fitxers per a millorar el rendiment i la seguretat del sistema. Crearem 3 noves particions en el disc dur i migrarem els directoris /var, /tmp i /opt en aquestes noves particions.

Utilitzarem el mateix disc dur que en l'escenari anterior. Com que ja tenim dades el que farem serà destruir totes les dades per a començar de nou.

Texto

Descripción generada automáticamente

Crearem les noves particions en el disc dur:  
Texto

Descripción generada automáticamente

Formatejem les particions amb els sistemes de fitxers corresponents i assignem les etiquetes, pero primer instalarem el sistema de fitxers xfs (apt install xfsprogs):  
  
Texto

Descripción generada automáticamente con confianza media

Montem les particions en directoris del sistema de fitxers. Montarem les particions a /mnt per a poder migrar els directoris.

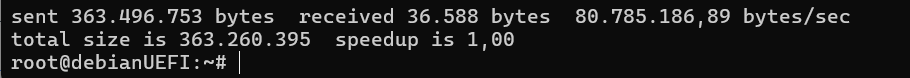
Texto

Descripción generada automáticamente

Imagen que contiene texto, marcador

Descripción generada automáticamente

Migrarem els directoris /var, /tmp i /opt a les noves particions. Per fer-ho podem utilitzar la comanda cp o rsync (primer haurem de fer apt install rsync)



I ara comprovarem que els directoris han migrat correctament:

Texto

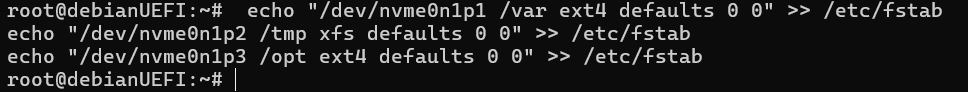
Descripción generada automáticamente

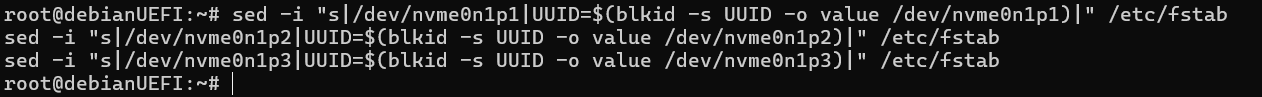
Montarem les particions en els directoris corresponents del sistema de fitxers.

Texto

Descripción generada automáticamente

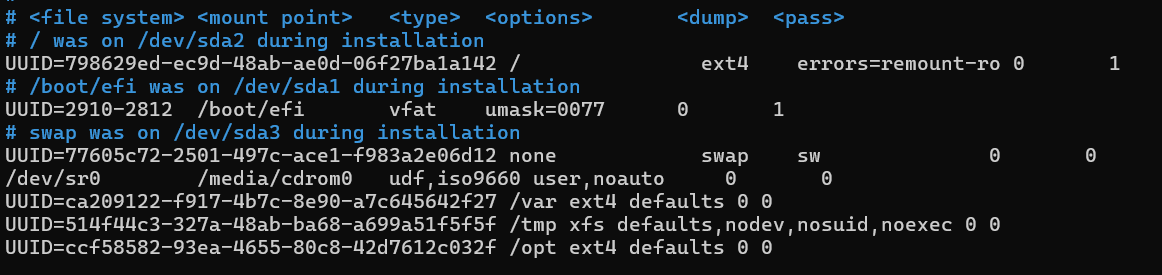
Llavors reiniciarem el sistema (reboot)  
Un cop reiniciada la màquina, modifiquem el fitxer /etc/fstab per a que les particions es muntin automàticament en l'arrencada del sistema.



Fem reboot un altre cop i comprovem que les particions es muntin automàticament a l’arrancada.  
  
En aquest punt podriam optimitzar la configuració particions per a millorar el rendiment del sistema. Utiltizarem els UUIDs en lloc dels dispositius per a muntar les particions:  


Utilitzarem opcions més específiques per protegir la partició /tmp per a evitar que s’executin programes des de la partició. Utilitzant les opcions **nodev**, **nosuid**, i **noexec**:

Editem el fitxer /etc/fstab i afegeix les opcions nodev, nosuid, i noexec a la partició /tmp.



Apliquem els canvis sense reiniciar el sistema executant

Texto

Descripción generada automáticamente

Comprovem les opcions després d'editar el fitxer /etc/fstab.



Ara farem reboot i testejarem les opcions de muntatge de la partició /temp  
Provem d'executar un programa des de la partició /tmp com a usuari no privilegiat:  
Texto

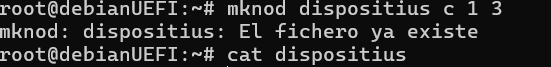
Descripción generada automáticamente

Provem de fer el mateix com a usuari privilegiat:  
Texto

Descripción generada automáticamente

Provem d'accedir a un dispositiu creat a /tmp:

Inicialitzem un dispositiu a /tmp utilitzant la comanda mknod i despres accedim al dispositiu:



Per veure, les implicacions de la opció nosuid, podem realitzar el següent experiment:

Texto

Descripción generada automáticamente

Compilem el programa amb la comanda gcc (primer haurem d’instalar el paquet gcc)

Texto

Descripción generada automáticamente

Desactivem la opció nosuid i també la opció noexec per a la partició /tmp. Per fer-ho, utilitza l'usuari root per a editar el fitxer /etc/fstab.

Remuntem la partició /tmp. I provem d'executar el programa com a usuari no privilegiat.

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

Anem a fer una simulació de com respon el sistema en cas de no tenir els directoris /var, /tmp i /opt en particions separades.

Desmuntem la partició /opt per a poder continuar amb l'exercici.



Creem un fitxer de 20GB al directori /opt i comprovem el sistema:

Texto

Descripción generada automáticamente

Intentem instal·lar un paquet amb apt i comprovem que el sistema falla.

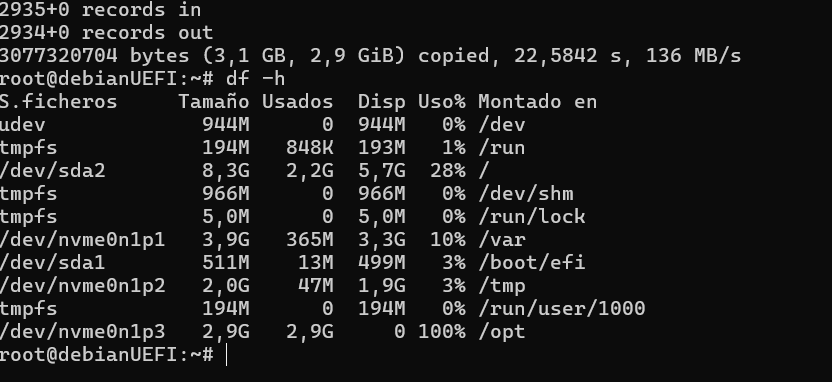
Texto

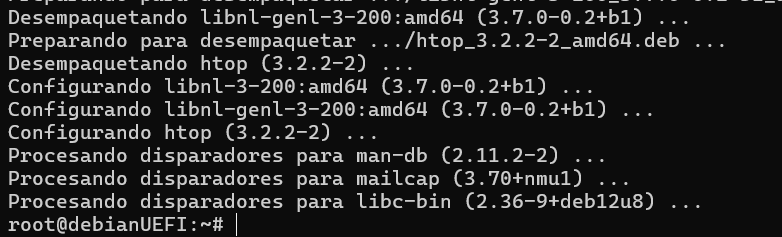
Descripción generada automáticamente

Eliminem el fitxer que hem creat per a poder continuar amb la resta de l'exercici i muntem la partició /opt i torneu a crear el fitxer per a omplir el directori.

Texto

Descripción generada automáticamente



Ara únicament tenim la partició /opt plena; però el sistema pot continuar fent tasques:

**5.3 – Simulant una corrupció a /home i recuperant-la**

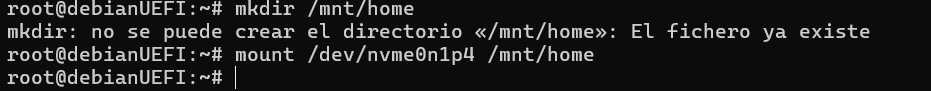
El primer pas que farem serà migrar el directori /home a una partició diferent per a poder simular la corrupció. A continuació, simularem la corrupció i finalment recuperarem les dades.

Crearem una nova partició en el disc dur extern:

Texto

Descripción generada automáticamente

Crearem un directori on muntarem la nova partició i muntarem la nova partició en el directori /mnt/home:

  
Copiarem totes les dades del directori /home a la nova partició i desmontarem la partició:

Texto

Descripción generada automáticamente

Modificarem el fitxer /etc/fstab per a que la partició es munti automàticament en l'arrencada del sistema:



I ara farem reboot per comprovar que la partició es munta automàticament.

Texto

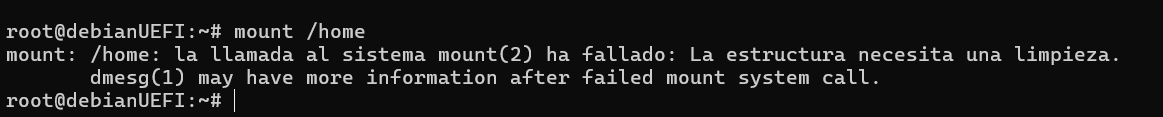
Descripción generada automáticamente

Ara podem crear fitxers i directoris al directori /home del nostre usuari normal.

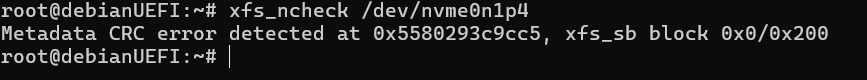
Llavors desmuntem la partició /home i obrirem la partició /home amb un editor hexadecimal (haurem d’instalar l’editor hexadecimal):  


Un cop obert modifiquem bits a l'atzar i desem els canvis.

Ara intentarem muntar la partició /home per a comprovar que ha estat corrompuda:



Els diferents sistemes de fitxers tenen eines per a comprovar la integritat dels fitxers. Per exemple, el sistema de fitxers xfs té l'eina xfs\_ncheck que permet comprovar la integritat dels fitxers.



Per a reparar la partició /home utilitzarem l'eina xfs\_repair.

Texto

Descripción generada automáticamente

Un cop reparada la partició /home, la muntarem de nou i comprovarem que la partició /home s'ha muntat correctament i conté els fitxers que havíem creat.



**5.4 -** [**Sistemes de fitxers temporals**](https://amsa-2425-gei-udl.github.io/laboratoris/Filesystems/temporal.html#sistemes-de-fitxers-temporals)

En aquest laboratori, compararem el rendiment dels sistemes de fitxers ext4, xfs i tmpfs en un entorn de còmput científic.

Per crear el sistema de fitxers temporal tmpfs, podem utilitzar la comanda següent:

PREPARACIÓ DEL ENTORN:  
1 - Creem un nova màquina virtual amb Debian 12. Podem utiltizar una configuració per defecte amb 2GB de RAM i 20GB d'espai en el disc principal.

2 - Creem un disc secundari de 20GB per crear els sistemes de fitxers ext4, xfs i tmpfs.

3 - Particionem el disc secundari amb 3 particions: ext4, xfs i tmpfs cada una amb 5GB d'espai.

4 - Muntem les particions a les rutes /mnt/ext4, /mnt/xfs i /mnt/tmpfs respectivament.

Texto

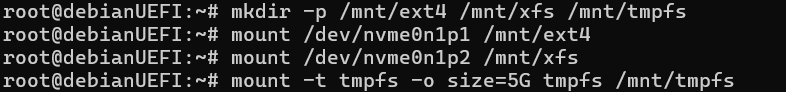
Descripción generada automáticamente

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente



Pantalla de computadora con letras

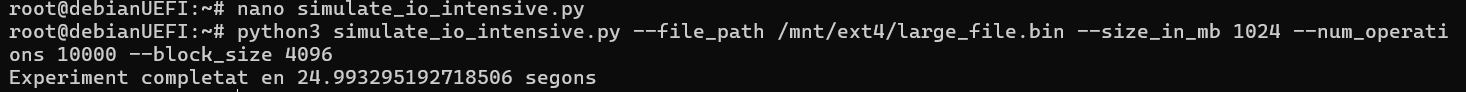
Descripción generada automáticamente con confianza baja

Per simular el nostre experiment, crearem un script de Python que generi un fitxer gran i realitzi moltes operacions d'escriptura aleatòria. Aquest script pot ser executat en un servidor de còmput per provar el rendiment del sistema de fitxers. Com a parametres d'entrada podem especificar la ruta del fitxer, la mida del fitxer, el nombre d'operacions d'escriptura aleatòria i la mida del bloc d'escriptura.

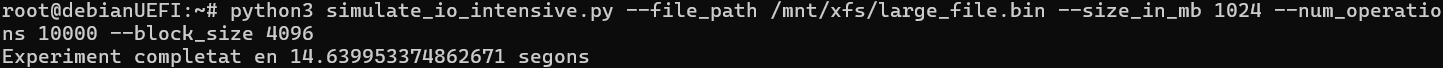
Texto

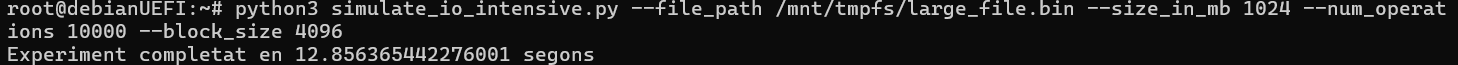
Descripción generada automáticamente

Un cop analitzat el codi, podem executar l'experiment per comparar el rendiment dels sistemes de fitxers ext4, xfs i tmpfs.

Executem l'experiment amb el sistema de fitxers ext4:

Executem l'experiment amb el sistema de fitxers xfs:



Executem l'experiment amb el sistema de fitxers tmpfs:

Ara realitzarem l'experiment 10 vegades per obtenir una mitjana del temps d'execució per a cada sistema de fitxers.

Texto

Descripción generada automáticamente**Sistema de fitxers**: ext4

MITJA DE: 19,624 segons

Texto

Descripción generada automáticamente**Sistema de fitxers**: xfs

MITJA DE: 12,317 segons

**Sistema de fitxers**: tmpfs

Texto

Descripción generada automáticamente

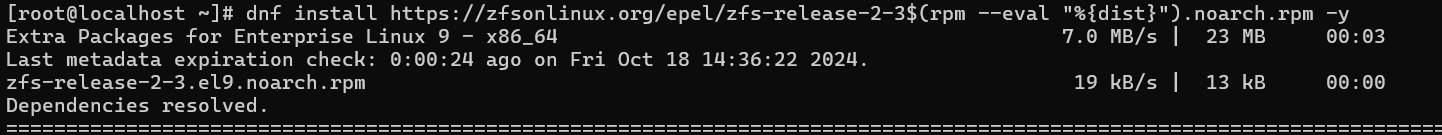
MITJA DE: 10,942 segons

**5.5 -** [**Explorant d'un sistema de fitxers avançat: zfs**](https://amsa-2425-gei-udl.github.io/laboratoris/Filesystems/zfs.html#explorant-dun-sistema-de-fitxers-avan%C3%A7at-zfs)

Obrim la màquina virtual d’AlmaLinux i instal·lem el repositori EPEL:

Texto

Descripción generada automáticamente

Afegim el repositori ZFS:  


I instal·lem també els paquets necessaris:  
Tabla

Descripción generada automáticamente con confianza baja

Quan acabi actualitzarem el sistema amb: dnf Update.  
Llavors per poder instal·lar els mòduls kABI-tracking, cal canviar el repositori predeterminat de zfs a zfs-kmod.

Texto

Descripción generada automáticamente

Farem reboot quan acabi de completar-se i carregarem el mòdul zfs a kernel de Linux:

Código QR

Descripción generada automáticamente con confianza baja

Ara començarem amb la creació d’una pool ZFS primer afegint el disc /dev/vdb que el zfs buscarà.



Comprovarem la pool amb: zpool status i crearem un sistema de fitxers que anomenarem dades amb la següent comanda:  


Ara canviarem el punt de muntatge de zfspool/dades a /mnt/dades

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

I crearem uns quants fitxers i directoris a /mnt/data:

Texto, Carta

Descripción generada automáticamente

Crearem una instantània del nostre sistema de fitxers zfspool/dades:



I eliminarem el directori **/mnt/dades/adria/config** amb la comanda rm, també podem utilitzar la comanda **zfs rollback** per restaurar el nostre sistema de fitxers a l'estat de l'instantània i comprovar que el directori s’hagi restaurat amb ls -la /mnt/...

També podrem utilitzar la comanda zfs clone per clonar el nostre sistema de fitxers. Per eliminar aquest clon utilitzarem zfs destroy i per eliminar la pool farem zpool destroy.

**En aquest últim apartat de la creació de la pool no he pogut adjuntar les imatges de la meva màquina virtual perquè no s’em van guardar correctament i vaig eliminar els hard disks de la màquina perdent-ho tot ☹**