

Lab 5: Backups

1. **What commands you have used to create the partition, format the filesystem, mount the partition and change the permissions of the /backup directory?**

```
sudo mkdir /backup
```

```
sudo fdisk /dev/sda
```

```
sudo mkfs.btrfs /dev/sda8
```

```
echo '/dev/sda8 /backup btrfs defaults 0 0' | sudo tee -a /etc/fstab
```

```
sudo chmod 700 /backup
```

2. **What do you need to do in order to mount this new partition automatically in read-only mode at boot-time?**

```
En el /etc/fstab
```

```
/dev/sda8 /backup btrfs defaults,ro 0 0
```

3. **How you can include automatically the date in the backup file name? for example, backup-etc-level0-200912041030.tar**

```
$(date +%Y%m%d%H%M)
```

4. **What command you have used for making the full copy of the directory /root?**

```
sudo tar -cvf /backup/backup-root-full-$(date +%Y%m%d%H%M).tar /root
```

5. **If we want to compress the backup file which option you have to add to the tar command?**

```
sudo tar -czvf backup-root-full-$(date +%Y%m%d%H%M).tar.gz /root
```

6. **Why is not generally good in terms of security to compress the backup file?**

Quan els fitxers es comprimeixen, el contingut es combina i es pot perdre l'accés granular als fitxers individuals.

Si el fitxer comprimit conté dades sensibles, és més vulnerable a ser desxifrat en cas de compromís, ja que l'arxiu complet pot ser atacat de cop.

La compressió pot revelar patrons del contingut que poden ser explotats per un atacant.

7. **What command you have added to the tar command to exclude files?**

```
sudo tar --exclude-from=excludes.txt -cvf /backup/backup-root-full-$(date +%Y%m%d%H%M).tar /root
```

En excludes.txt posem les rutes dels directoris o fitxers que volem excloure.

8. How you have used the sha512sum command to produce the sha512 signature?

```
sha512sum /backup/backup-root-full-$(date +%Y%m%d%H%M).tar.gz > /backup/backup-root-full-$(date +%Y%m%d%H%M).asc
```

9. What command you used to make the incremental copy?

```
sha512sum /backup/backup-root-full-$(date +%Y%m%d%H%M).tar.gz > /backup/backup-root-full-$(date +%Y%m%d%H%M).asc
```

10. What potential problem has the use of the full backup file for obtaining the backup date when doing the incremental copy? How you can solve this problem?

El problema principal és que el procés pot ser lent perquè el fitxer de còpia completa és gran. Això obliga el sistema a llegir les metadades del fitxer complet per obtenir la data de modificació, cosa que pot retardar el procés, especialment si el fitxer està en un dispositiu de xarxa o molt fragmentat.

Utilitzar un fitxer separat dedicat a emmagatzemar la data de la còpia completa. Aquest fitxer només contindrà una data i serà més ràpid d'accedir:

```
touch -r backup-root-full-202312061530.tar.gz last-backup-date
```

```
sudo tar --newer=last-backup-date -cvf /backup/backup-root-incremental-$(date +%Y%m%d%H%M).tar /root
```

11. What command you have used to create the second incremental copy?

```
sudo tar --newer=backup-root-incremental-202312061700.tar -cvf /backup/backup-root-incremental-$(date +%Y%m%d%H%M).tar /root
```

12. How can you verify that the content of the backup is the same as the original directory?

```
diff -r /root /tmp/backup
```

13. How can you verify, using the sha512sum command, the integrity of the backup?

```
sha512sum -c /backup/backup-root-incremental-$(date +%Y%m%d%H%M).asc
```

14. In what order you should restore the files to get the desired result?

Primer la còpia completa, després la primera incremental, i finalment la segona incremental.

15. What commands you have used?

```
sudo mv /root /root.old
```

```
sudo tar -xvf /backup/backup-root-full-202312061530.tar.gz -C /
```

```
sudo tar -xvf /backup/backup-root-incremental-202312061700.tar -C /
```

```
sudo tar -xvf /backup/backup-root-incremental-202312061900.tar -C /
```

16. What happened to the files that you had deleted before the second incremental copy?

No es restauren perquè no estan presents a la segona còpia incremental.

17. How you can detect that some files that have been deleted? When those file will be removed from the backups?

Utilitza diff per comparar /root.old amb /root.

Quan generis una nova còpia completa.

18. What commands you have used to retrieve only a portion of the backup?

`tar -xvf /backup/backup-root-incremental-202312061900.tar -C / root/subdir`

19. What are the required steps for installing and activating the ssh server?

`sudo apt update`

`sudo apt install openssh-server`

`sudo systemctl enable ssh`

`sudo systemctl start ssh`

Edita /etc/ssh/sshd_config i canvia PermitRootLogin a yes

`sudo systemctl restart ssh`

20. What is the meaning of the options “avz” passed to rsync?

-a: Mode arxiu per preservar metadades.

-v: Sortida detallada.

-z: Comprensió de dades durant la transferència.

21. What happened to the deleted file?

No s'ha eliminat del directori de destinació perquè `rsync` no sincronitza fitxers eliminats per defecte.

22. What option of rsync allows to an exact synchronization the two directories?

L'opció `--delete` sincronitza exactament eliminant fitxers que no existeixen a l'origen.

23. What's the difference between making rsync /source /destination and rsync /source /destination/?

rsync /source /destination:

Copia el directori **/source** sencer dins del directori **/destination**. Això significa que el directori **/source** es converteix en un subdirectori de **/destination**.

Exemples:

`/destination/source/file1`

`/destination/source/file2`

rsync /source /destination/:

Copia només el contingut del directori **/source** dins de **/destination**, però no el directori en si.

Exemples:

`/destination/file1`

`/destination/file2`

24. How you can make a copy of all the files in the /root directory except those that have a .txt extension?

`rsync -avz --exclude='*.txt' /root /backup/rsync-backup/`

25. What happens to the backup when the new file is modified?

Es mou el fitxer original al directori incremental i es guarda la versió modificada al directori complet.

26. And, what happens when it is deleted?

El fitxer es mou al directori incremental per mantenir-ne una còpia abans de ser eliminat.

27. How you can detect if file a and file b belong to the same inode?

Utilitza `stat` i compara els números d'inode.

28. What happens to file b if there are changes in file a content?

Els canvis es reflecteixen automàticament a `file_b`.

29. What happens to file b if there are changes in file a permissions?

Els canvis en els permisos també s'apliquen a `file_b`.

30. What happens to file b if you copy another file, overwriting the file: cp file c file a?

En aquest cas, `file_a` es vincularà a un nou inode corresponent a `file_c`. Això farà que `file_b` es mantingui vinculat al inode original, sense veure's afectat.

31. And if it is overwritten with the --remove-destination option of cp?

Quan utilitzes `--remove-destination`, `cp` elimina `file_a` abans de copiar-hi `file_c`. Això fa que l'inode de `file_a` es deslligui, i `file_b` es manté vinculat a l'inode original sense canvis.

32. And what happens to file b if file a is removed?

Si elimines `file_a`, l'inode no s'elimina immediatament perquè `file_b` encara hi està vinculat. Només quan tots els enllaços durs (com `file_b`) es deslliguin, l'inode serà eliminat definitivament.

33. How is the rsync command in the script to make the snapshot copies?

`rsync -a --delete --link-dest=$SNAPSHOT_PREVIOUS $SOURCE_DIR/
$SNAPSHOT_CURRENT/`

(no es el mateix falten flags)

34. What happens to the backup after the execution of the script?

El directori /backup.0 es converteix en la còpia actual, que reflecteix l'estat actual del directori d'origen.

Els fitxers no modificats continuen com enllaços durs a les versions anteriors. Això significa que no ocupen espai addicional.

Els fitxers nous o modificats es copien completament i ocupen espai addicional.

Els fitxers eliminats de l'origen es marquen com a eliminats i ja no es troben a /backup.0.

35. What is the size of the directory /backup.0 and other directories backup directories?

/backup.0: Conté la còpia completa actual. Però, gràcies als enllaços durs, només ocupa espai addicional per als fitxers nous o modificats.

/backup.1, /backup.2, /backup.3: També semblen còpies completes, però ocupen només espai per als canvis introduïts respecte a la còpia anterior.

L'espai total ocupat per totes les còpies és igual a:

La mida del directori d'origen actual + els canvis acumulats entre les versions.

```
du -sh /backup.0 /backup.1 /backup.2 /backup.3
```

36. Where is the most recent data?

Les dades més recents es troben al directori /backup.0, ja que aquest és la còpia actual.

37. Which command can be used to recover the data?

Per restaurar les dades més recents al directori /root, utilitza:

```
sudo rsync -a /backup.0/ /root/
```

-a: Preserva permisos, propietaris, grups i altres metadades.

/backup.0/: Indica el directori de la còpia de seguretat més recent.

/root/: És el directori de destinació on es restauren les dades.