

µhackathon 5-6

Gestió de fitxers + temporització

Volem temporitzar un sistema de backup remot mensual on cada més es realitza una còpia de nivell 0 (completa) , cada setmana una copia de nivell 1 (incremental setmanal sobre el nivell 0) i cada dia es fa una còpia de nivell 2 (incremental diari sobre el nivell 1) que compleixin les següents condicions:

a) connexió entre màquines

- La connexió ha de ser segura (ssh).
- Per tal de poder temporitzar el backup es necessari que cada vegada que ens connectem a la màquina remota no ens demani el password

b) Còpies de seguretat

- El directori a copiar es: /var/log
- Els arxius .gz han de quedar exclosos.
- Els arxius copiats han de tenir els següents permisos al directori destí:
 - owner: rwx, group: - - - , i others: - - -.
- (Nota: mireu al man les opcions de rsync)
- La copia s'ha de realitzar en una màquina remota (clon) amb adreça IP 172.16.1.2 Aquesta màquina ha de tenir una partició exclusiva per als backups muntada a /home/aso/backups.
- Al directori de backup es guardaran solament les tres còpies més recents (rotació), identificades amb backup-<dir_bck>-nivell<x>-<any mes dia hora minut segon>. Ex.: backup-log-nivell0-202212011030.
- Les ordres de rotació es donaran des de la maquina host cap a la màquina remota

c) Temporització de les còpies

- Es realitzarà una copia nivell 0 (completa) cada primer dimarts del mes a les 00:05h
- Es realitzarà un copia incremental nivell 1 cada diumenge a les 00:15h.
- Es realitzarà un copia incremental nivell 2 cada dia a les 02:30h.

Entrega:

Especificacions:

- Tenir MV funcional
- Fer SSH sense necessitat de contrasenya
- Directori que es copia (/var/log)
- Excloure fitxers .gz
- Màquina remota clon amb IP: 172.16.1.2
- Còpies:
 - Nivell 0: Es realitza cada primer dimarts del mes a les 00:05 h.
 - Nivell 1: Es realitza cada diumenge a les 00:15 h.
 - Nivell 2: Es realitza cada dia a les 02:30 h

Llista de tasques amb responsable

- a) Connexió entre màquines → Ehsan
- b1) Còpies de seguretat - Completa → Sergi
- b2) Còpies de seguretat - Incremental 1 → Oscar
- b3) Còpies de seguretat - Incremental 2 → Ivan
- c) Temporització de les còpies → Víctor
- d) Documentar → Ausias

Algoritme

1. Connexió entre màquines
2. Còpies de seguretat - Completa
3. Còpies de seguretat - Incremental 1
4. Còpies de seguretat - Incremental 2
5. Comprovacions

Descripció de la implementació amb captures de pantalla de cada responsable

a) Connexió entre màquines

Primer observem la màquina remota té el servei SSH habilitat amb la comanda:

systemctl status ssh

```

aso@ivanP (jue dic 12) :~$systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; preset: enabled)
   Active: active (running) since Thu 2024-12-12 07:56:43 UTC; 1min 45s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
  Process: 797 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
 Main PID: 806 (sshd)
    Tasks: 1 (limit: 4631)
   Memory: 3.2M
      CPU: 43ms
   CGroup: /system.slice/ssh.service
           └─806 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Warning: some journal files were not opened due to insufficient permissions.
aso@ivanP (jue dic 12) :~$

```

Perquè la connexió SSH sigui automàtica i no ens demani la contrasenya, cal que configurem l'autenticació basada en claus SSH.

Escrivim la comanda:

ssh-keygen -t rsa -b 4096

```

root@oscarMP (Thu Dec 12):~# ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:v9UcI0ly/Qw+GCP3h18ynJh1tKURML+Jgn6HkL29THo root@aso-client
The key's randomart image is:
+---[RSA 4096]-----+
|          o.+..|
|          = = |
|          o * 0 |
|          + * # 0 |
|          S o B # =|
|          . o = + 0.|
|          . = = o .|
|          . BE.  |
|          o.o    |
+---[SHA256]-----+
root@oscarMP (Thu Dec 12):~#

```

Això generarà dues claus, una privada i una pública, per defecte a ~/.ssh/id_rsa (privada) i ~/.ssh/id_rsa.pub (pública). Llavors, per afegir la nostra clau pública a la màquina remota, farem servir la comanda ssh-copy-id. Aleshores, fem la comanda:

ssh-copy-id -p 3022 aso@maquina_remota

```

root@oscarMP (Thu Dec 12):~# ssh-copy-id -p 3022 aso@172.20.10.2
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '[172.20.10.2]:3022 ([172.20.10.2]:3022)' can't be established.
ED25519 key fingerprint is SHA256:18fbSfH/yFtxoRFSYL7YK/G0da+8DfRWlkidbqQvRhU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
aso@172.20.10.2's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh -p '3022' 'aso@172.20.10.2'"
and check to make sure that only the key(s) you wanted were added.

```

Això afegirà la clau pública al fitxer ~/.ssh/authorized_keys de la màquina remota, permetent-te la connexió sense contrasenya.

Verifiquem que podem fer SSH sense contrasenyes:

ssh -p 3022 aso@maquina_remota

```

root@oscarMP (Thu Dec 12):~# ssh -p 3022 aso@172.20.10.2
Linux aso-client 6.1.0-25-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-3 (2024-08-26) x86_64
Si tens cap problema i vols contactar amb l'administrador, pots contactar amb:
ivan.pena.carreno@estudiantat.upc.edu
Last login: Thu Dec 12 08:03:03 2024 from 172.20.10.6
aso@ivanP (jue dic 12) :~$

```

Assignem la IP 172.16.1.2 a la màquina remota (clon) modificant el fitxer /etc/network/interfaces.d

```

# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto enp0s3
iface enp0s3 inet static
    address 172.16.1.2
    netmask 255.255.255.0

```

b) Còpies de seguretat

Primer de tot hem de crear el directori de backups a /home/aso a la màquina destí

mkdir /home/aso/backups

```

root@oscarMP (Thu Dec 12):~# mkdir /home/aso/backups

```

Desmuntem la partició /dev/sda4 amb **umount /dev/sda4**

Llavors, tornem a muntar

```

root@IvanP (jue dic 12) aso-client:~# mount /dev/sda4 /home/aso/backups
root@IvanP (jue dic 12) aso-client:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0   32G  0 disk
├─sda1       8:1    0   10G  0 part /
├─sda2       8:2    0    2G  0 part [SWAP]
├─sda3       8:3    0    1K  0 part
├─sda4       8:4    0    6G  0 part /home/aso/backups
├─sda5       8:5    0    4G  0 part /usr/local
├─sda6       8:6    0    9G  0 part /home
└─sr0       11:0    1 56,1M  0 rom

```

i) Còpies de seguretat completa

Script:

```

C/C++
#!/bin/bash

DIRECTORI_ORIGEN="/var/log"
EXCLUSIONS="*.gz"
DESTI="172.16.1.2:/home/aso/backups"
NIVELL="nivell0"
DATA=$(date +"%Y%m%d%H%M%S")
NOM_COPIA="backup-log-${NIVELL}-${DATA}"
DIRECTORI_DESTI="$DESTI/$NOM_COPIA"

# Comprovar connexió SSH
if ! ssh -q 172.16.1.2 exit; then
    echo "Error: No es pot establir connexió SSH amb la màquina remota."
    exit 1
fi

# Esborrar còpia de seguretat anterior de nivell 0
ssh 172.16.1.2 "cd /home/aso/backups && rm -rf backup-log-${NIVELL}-*"
DELETE_RESULT=$?

if [ $DELETE_RESULT -eq 0 ]; then
    echo "Còpia de seguretat anterior esborrada amb èxit."
else
    echo "Error: No s'ha pogut esborrar la còpia de seguretat anterior."
    exit 2
fi

# Crear nova còpia de seguretat
rsync -av --exclude="$EXCLUSIONS" --chmod=u=rwx,g=,o= "$DIRECTORI_ORIGEN/"
"$DIRECTORI_DESTI"
RSYNC_RESULT=$?

if [ $RSYNC_RESULT -eq 0 ]; then
    echo "Còpia de seguretat de nivell 0 completada amb èxit."
else
    echo "Error: No s'ha pogut realitzar la còpia de seguretat."
    exit 3
fi

```

Comprovació:

Comprovem que el script funcioni correctament donant els permisos al script complta.sh amb `chmod +x completa.sh` i executant amb `./completa.sh`

```
xterm
journal/61351dfd56e248b9a14564ee51c2d153/user-1014.journal
journal/61351dfd56e248b9a14564ee51c2d153/user-1015.journal
journal/61351dfd56e248b9a14564ee51c2d153/user-1018.journal
journal/61351dfd56e248b9a14564ee51c2d153/user-1022.journal
journal/61351dfd56e248b9a14564ee51c2d153/user-1022@000627f71c52ceef-022b7ed1492f8e31.journal~
journal/61351dfd56e248b9a14564ee51c2d153/user-1023.journal
lightdm/
lightdm/lightdm.log
lightdm/lightdm.log.old
lightdm/seat0-greeter.log
lightdm/seat0-greeter.log.old
lightdm/x-0.log
lightdm/x-0.log.old
lightdm/x-1.log
lightdm/x-1.log.old
private/
runit/
runit/ssh/

sent 435,959,519 bytes received 2,069 bytes 124,560,453.71 bytes/sec
total size is 435,844,289 speedup is 1.00
Càpia de seguretat de nivell 0 completada amb èxit.
root@Victor (Thu Dec 12) >:~/Downloads#
```

Comprovem que s'ha creat el primer backup

```
root@Victor (Thu Dec 12) >:~/Downloads#cd /home/homeB/aso/backups/
root@Victor (Thu Dec 12) >:/home/homeB/aso/backups#ls
backup-log-nivell0-20241212090012 lost+found
root@Victor (Thu Dec 12) >:/home/homeB/aso/backups#
```

Comprovem el que hi ha dins del backup

```
root@Victor (Thu Dec 12) >:/home/homeB/aso/backups/backup-log-nivell0-20241212090012#ls
README          auth.log         bttmp           faillog         runit
Xorg.0.log       auth.log.1       bttmp.1         fontconfig.log  syslog
Xorg.0.log.old   boot.log         cron.log        journal         syslog.1
Xorg.1.log       boot.log.1       cron.log.1      kern.log        user.log
Xorg.1.log.old   boot.log.2       daemon.log      kern.log.1      user.log.1
alternatives.log boot.log.3       debug          lastlog         wtmp
alternatives.log.1 boot.log.4       dpkg.log        lightdm
apparmor         boot.log.5       dpkg.log.1      messages
apt             bootstrap.log    exim4           private
```

ara per comprovar que funciona la sobreescritura dels backups del mateix nivell per això creem un fitxer al directori `var/log`

```
root@Victor (Thu Dec 12) >:/var/log#touch Fitxeraux.txt
root@Victor (Thu Dec 12) >:/var/log#ls
Fitxeraux.txt  auth.log.1       bttmp.1         faillog         runit
README        auth.log.2.gz    cron.log        fontconfig.log  syslog
Xorg.0.log     auth.log.3.gz    cron.log.1      journal         syslog.1
Xorg.0.log.old boot.log         cron.log.2.gz   kern.log        syslog.2.gz
Xorg.1.log     boot.log.1       cron.log.3.gz   kern.log.1      syslog.3.gz
Xorg.1.log.old boot.log.2       daemon.log      kern.log.2.gz   user.log
alternatives.log boot.log.3       debug          kern.log.3.gz   user.log.1
alternatives.log.1 boot.log.4       dpkg.log        lastlog         user.log.2.gz
apparmor       boot.log.5       dpkg.log.1      lightdm         user.log.3.gz
apt            bootstrap.log    dpkg.log.2.gz   messages        wtmp
auth.log       bttmp           exim4           private
```

executem un altre cop el script i comprovem que en el nou backup estigui l'arxiu que hem creat

```
root@Victor (Thu Dec 12) >:/home/homeB/aso/backups/backup-log-nivell10-20241212090247#ls
Fitxeraux.txt      apt          bootstrap.log  exim4         private
README            auth.log     bttmp         faillog        runit
Xorg.0.log         auth.log.1   bttmp.1       fontconfig.log syslog
Xorg.0.log.old     boot.log     cron.log      journal        syslog.1
Xorg.1.log         boot.log.1   cron.log.1    kern.log       user.log
Xorg.1.log.old     boot.log.2   daemon.log    kern.log.1     user.log.1
alternatives.log   boot.log.3   debug         lastlog        wtmp
alternatives.log.1 boot.log.4   dpkg.log      lightdm
apparmor           boot.log.5   dpkg.log.1    messages
```

Comprovem que s'ha eliminat el primer backup i ha canviat pel segon que hem creat

```
root@Victor (Thu Dec 12) >~/Downloads#cd /home/homeB/aso/backups/
root@Victor (Thu Dec 12) >:/home/homeB/aso/backups#ls
backup-log-nivell10-20241212090648  lost+found
root@Victor (Thu Dec 12) >:/home/homeB/aso/backups#
```

ii) Còpies de seguretat incremental 1

Script:

```
Unset
#!/bin/bash

# Variables
REMOTE_USER="aso"
REMOTE_HOST="172.16.1.2"
REMOTE_BACKUP_DIR="/home/aso/backups"
SOURCE_DIR="/var/log"
EXCLUDE="*.gz"
TIMESTAMP=$(date +%Y%m%d%H%M")
BACKUP_NAME="backup-log-nivell11-${TIMESTAMP}"
LOCAL_BACKUP_DIR="/tmp/${BACKUP_NAME}"

# Rsync incremental
rsync -av --exclude="${EXCLUDE}" \
    --chmod=ug=rwX,o= \
    "${SOURCE_DIR}/" "${LOCAL_BACKUP_DIR}"

# Copiar al servidor remot
rsync -av "${LOCAL_BACKUP_DIR}" \
    "${REMOTE_USER}@${REMOTE_HOST}:${REMOTE_BACKUP_DIR}/"

# Rotación: mantener las 3 últimas copias de nivel 1
ssh ${REMOTE_USER}@${REMOTE_HOST} "cd ${REMOTE_BACKUP_DIR} && ls -t | grep
'backup-log-nivell11' | tail -n +2 | xargs -I {} rm -rf {}"
```

Comprovació:

iii) Còpies de seguretat incremental 2

Script:

```
Unset
#!/bin/bash

# Variables
REMOTE_USER="aso"
REMOTE_HOST="172.16.1.2"
REMOTE_BACKUP_DIR="/home/aso/backups"
SOURCE_DIR="/var/log"
EXCLUDE="*.gz"
TIMESTAMP=$(date +%Y%m%d%H%M")
BACKUP_NAME="backup-log-nivell2-${TIMESTAMP}"
LOCAL_BACKUP_DIR="/tmp/${BACKUP_NAME}"

# Rsync incremental
rsync -av --exclude="${EXCLUDE}" \
      --chmod=ug=rwx,o= \
      "${SOURCE_DIR}/" "${LOCAL_BACKUP_DIR}"

# Copiar al servidor remot
rsync -av "${LOCAL_BACKUP_DIR}"
      "${REMOTE_USER}@${REMOTE_HOST}:${REMOTE_BACKUP_DIR}/"

# Rotació: mantenir la última copia de nivell 2
ssh ${REMOTE_USER}@${REMOTE_HOST} "cd ${REMOTE_BACKUP_DIR} && ls -t | grep
'backup-log-nivell2' | tail -n +2 | xargs -I {} rm -rf {}"
```

Comprovació:

c) Temporització de les còpies

Per temporitzar les còpies de seguretat hem de executar la comanda

crontab -e

I hem d'afegir les següents línies.

Per a la copia completa

```
# m h dom mon dow  command
5 0 1-7 * * ["$(date '+\%a')" == "Tue"] && ./completa.sh
```

la primera part 5 0 1-7 * *, marca que el script s'ha executar cada dia de la primera setmana del mes a les 00:05. Per fer que només ho executi dimarts, filtrem la data per a que el dia sigui igual a "Tue", que significa Tuesday que es Dimarts en anglès.

Per a la copia incremental de nivell 1, ens demanen que es faci cada diumenge a les 00:15h.

Per marcar l'hora posem 15 0 en els dos primers valors.

El dia del mes i el més ens dona igual per tant * *.

I el dia de la setmana volem que sigui Diumenge, per tant posem 7.

La comanda quedaria així:

```
15 0 * * 7 ./incremental1.sh
```

I per la incremental de nivell 2, ens demanen que sigui cada dia a les 2:30.

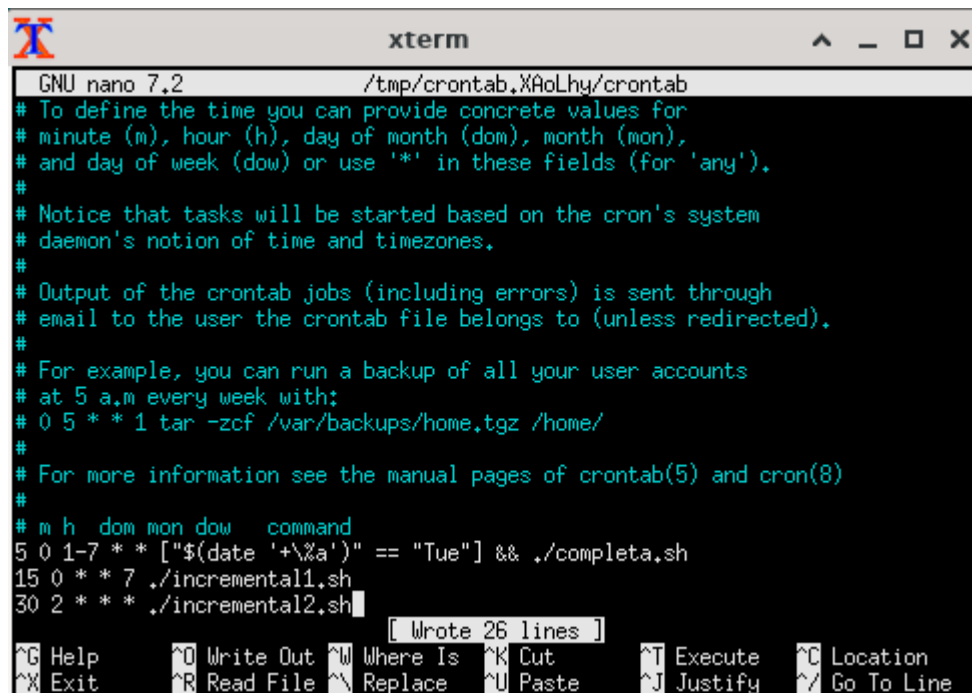
Per marcar l'hora posem 30 2 en els dos primers valors.

Com volem que sigui cada dia, els valors restants han de ser *.

La comanda quedaria així:

```
30 2 * * * ./incremental2.sh
```

El document s'hauria de veure així:



```
GNU nano 7.2 /tmp/crontab.XAoLhy/crontab
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
5 0 1-7 * * [ "$(date '+%a')" == "Tue" ] && ./completa.sh
15 0 * * 7 ./incremental1.sh
30 2 * * * ./incremental2.sh
```

[Wrote 26 lines]

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line