

Script de Respaldo y Compresión

Estudia el siguiente script y explica lo que realizaría

```
#!/bin/bash
fecha=$(date +"%d-%m-%Y")
tar -czf backup_.$fecha.tar.gz /ruta/a/copiar1 /ruta/a/copiar2
```

Ahora, programa este script para que respalde /home todos los domingos del año a las 14:55. Las copias en /copias

```
tar -czf /copias/home_.$fecha.tar.gz /home/
```

```
Xubuntu_ASO [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
copia.sh (/usr/local/bin) - ge... Terminal - miguel@miguel-a... Terminal - miguel@miguel-a... Terminal - miguel@miguel-a...
Abrir + copia.sh /usr/local/bin Guardar
1#!/bin/bash
2fecha=$( date +"%Y-%m-%d" )
3echo "$fecha"
4tar -czf /copias/home_.$fecha.tar.gz /home/
Miguelillo
Terminal - miguel@miguel-a...
Archivo Editar Ver Terminal Pestañas Ayuda
miguel@miguel-aso:/tmp$ sudo crontab -e
no crontab for root - using an empty one
Select an editor. To change later, run 'select-editor'.
 1. /bin/nano          <---- easiest
 2. /usr/bin/vim.tiny
 3. /bin/ed
Choose 1-3 [1]: 1
crontab: installing new crontab
Miguelillo
GNU nano 6.2 /tmp/crontab.s4uBxc/crontab *
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# 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 -czf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow command
15 13 * * 1 /usr/local/bin/copia_programada
```

Monitoreo del Uso del Espacio en Disco

Queremos realizar un script que, cuando se ejecute, monitoree el espacio que ocupa un determinado directorio (será la partición en la que se creó) y si supera un determinado umbral, emitir una alerta.

Ayuda: **df directorio**

```
antonio@aso-srv-01:/usr/local/bin$ df /var
S.ficheros      bloques de 1K  Usados Disponibles  Uso% Montado en
/dev/sda5        28659620 2953236    24225220   11% /var
```

```
#!/bin/bash
umbral=60
read -p "Introduce un disco o directorio: " DIRE
uso=$(df $DIRE | tr -s " " | cut -d" " -f5 | cut -d"%" -f1)
#se usa para coger solamente el campo de abajo pero es más
optimo usar un tr en el comando anterior
uso1=$(echo $uso | tr -s " " | cut -d" " -f2)

echo "El porciento del disco usado es $uso1 %"
if [ $uso1 -gt $umbral ]; then
    echo "cuidado"
else
    echo "esta bien"
fi

#otra manera
#read -r particion USOV3 <<< $(df -h /etc | tail -n 1 | tr -s "
" | cut -d " " -f1,5)
#USOV3=${USOV%\%}
#
#
#otra manera
#df -h /etc | tail -n 1 | tr -s " " | cut -d " " -f5 | tr "%" "
"
#
```

Presentamos a continuación una serie de scripts relacionados con la automatización de tareas relacionadas con usuarios, grupos, directorios de trabajo, etc.

1.- Mejorar el script **addgrupos.sh** de tal forma que, cuando vaya procesando cada línea:

- a.- Compruebe, antes de crear el grupo si este existe ya.
- b.- No añada grupos cuyo nombre sea mayor a 15 caracteres.
- c.- Indique si no se especifica el grupo (línea en blanco)

Fichero de prueba: grupos.txt (en rojo las filas que se consideran inválidas)

alumnado

profesorado

direccion

pas

alumnado

secretaria

abcdefghijklmnpqrst

jefatura

Salida

Procesando la fila: alumnado

Grupo alumnado añadido al sistema

....

Procesando la fila:

Fila anulada. No se ha especificado el grupo

....

Procesando la fila: alumnado

El grupo ya existe. No lo creamos

....

Procesando la fila: abcdefghijklmnpqrst

El grupo abcdefghijklmnpqrst supera los 15 caracteres y por tanto no se crea.

```
#!/bin/bash
```

```
ROOT_UID=0
```

```
if [ $UID -ne $ROOT_UID ] ; then
```

```
    echo "Error, debes ser root"
```

```
    exit 3
```

```
fi
```

```
if [ $# -ne 1 ] ; then
    echo "Error debes introducir un fichero"
    exit 4
fi
FICHERO=$1

if [ ! -f $FICHERO ] ; then
    echo "$Fichero no es un fichero regular o no existe"
    exit 5
fi

for GRUPO in $(cat $FICHERO); do
    #>> /dev/null
    if cat /etc/group | cut -d":" -f1 | grep $GRUPO>/dev/null ;
then
        echo "Este grupo $GRUPO ya existe"

    else
        echo "creando el grupo $GRUPO ..."
        /usr/sbin/addgroup $GRUPO

    fi
done
```

- 2.- Mejorar el script **addusuarios.sh**, de tal forma que compruebe si:
- a.- Existe el grupo.
 - b.- Existe el usuario.
 - c.- La longitud de la cuenta de usuario es superior a 15 caracteres en cuyo caso no se añadirá.
 - d.- Falta algún campo, en cuyo caso no se hará nada.
- Al igual que en el ejercicio anterior, ir mostrando en pantalla la fila procesada.

Fichero de prueba: usuarios.csv (en rojo las filas que se consideran inválidas)

```
angela, alumnado,123
david, alumnado, 456
juan,blandengues,123
mario, profesorado, 789
david,profesorado,123
paraquevaleestatoneria,alumnado,123
pedro,,123
,algo,
```

```
#!/bin/bash
ROOT_UID=0
if [ $UID -ne $ROOT_UID ] ; then
    echo "Error, debes ser root"
    exit 3
fi

if [ $# -ne 1 ] ; then
    echo "Error debes introducir un fichero"
    exit 4
fi

FICHERO=$1

if [ ! -f $FICHERO ] ; then
    echo "$Fichero no es un fichero regular o no existe"
    exit 5
fi

for LINEA in $(cat $FICHERO); do
```

```

USUARIO=$(echo $LINEA | cut -d"," -f1)
GRUPO=$(echo $LINEA | cut -d"," -f2)
CLAVE=$(echo $LINEA | cut -d"," -f3)
cat /etc/group | cut -d":" -f1 | grep $GRUPO >> /dev/null
if [ $? -eq 0 ] ; then

    cat /etc/passwd | cut -d":" -f1 | grep $USUARIO >>
/dev/null
    if [ $? -eq 1 ] ; then
        LON=$(echo $USUARIO | wc -m)

        if [[ $LON -lt 16 ]] ; then
            LON=$(echo $USUARIO $GRUPO $CLAVE | wc -w)

            if [ $LON = 3 ] ; then
                echo "creando el usuario $USUARIO..."
                /usr/sbin/useradd -s /bin/bash -m -g
$GRUPO $USUARIO

                echo "$GRUPO"
                echo "$USUARIO:$CLAVE" | chpasswd
            else
                echo "El fichero tiene que tener los
tres campos actualmente contiene $USUARIO $GRUPO $CLAVE"
            fi

        else
            echo "El usuario como máximo tiene que
ocupar menos de 15 caracteres"

        fi

    else
        echo "El usuario $USUARIO no ha sido creado ya
que está creado"
    fi

```

```
fi

else
    echo "El usuario $USUARIO no ha sido creado ya que se
    ha indicado un grupo que no está creado"

fi
done
```