# Protocolo de Backup

- master → servidor que hace las copias de seguridad
- esclavo(s) → servidor que se deja hacer las copias de seguridad

## esclavo

- (1) crear cuenta específica para backups: `$ useradd -c «Usuario de backup» -m -d /home/bck bck`
- (4) crear directorio de ficheros de backups, configurar el `bck.ini`
- (5) configurar en la cuenta de ROOT los cron a ejecutar
  - 01:00 → eliminar ficheros de backup anteriores a 1 mes: `00 01 * * * find /home/bck/backups -type f -mtime +30 -exec rm -f {} \; > /dev/null 2>&1`
  - 02:00 → volcado logs?
  - 03:00 → volcado TAR : `00 03 * * * /home/bck/bck-files.sh > /dev/null 2>&1`
  - 03:45 → volcado BBDD: `45 03 * * * /home/bck/bck-mysql.sh > /dev/null 2>&1`
  - 04:15 → encriptación si es necesaria: `15 04 * * * /home/bck/bck-gpg.sh > /dev/null 2>&1`

## master

- (0) crear certificado
- (2) copiarlo en el servidor remoto, en la cuenta creada a tal efecto: `$ copy-ssh-id -i rsa_192.168.3.199.pub bck@192.168.3.163`
- (3) copiar los scripts de backup: `$ scp -i rsa_192.168.3.199 /home/bck/bck* bck@192.168.3.163:/home/bck`
- (6) generar nuevo script de conexión y descarga de archivos del esclavo
  - ¿sincronización o mantener histórico en master?
  - en esclavo solo los últimos 30 días

## scripts

bck.ini

```
# BACKUP FILES
# directorios de los que hacer backup, separado por espacio, sin / final
LISTA="/etc /home/mate"

# RUTAS
# Local Directory for Dump Files
LOCALDIR=/home/bck/

# BDD
# Hostname
DBHOST='localhost'
# dbuser
DBUSER='root'
# dbpass
dbpasswd=''
```

```
# MISCELANEA
# Nombre del servidor
HOSTNAME='pandora'
# Estampacion de fecha
FECHA=`eval date +%y%m%d`
```

bck-files.sh

```
#!/bin/bash

#
# file system backup
#

source /home/bck/bck.ini

# crea un tar de los directorios especificados en LISTA
for directorio in $LISTA
do
    DESTINO=`echo $directorio | sed -e 's/\//-/g'`
    tar zcpf $LOCALDIR$FECHA-$HOSTNAME$DESTINO.tar.gz $directorio/*
done

exit 0
```

bck-gpg.sh

```
#!/bin/bash
FICHEROS=`ls *.tar.gz`
# FECHA=`eval date +%y%m%d`


for FICHERO in $FICHEROS
do
  #encripta
  gpg -r "Miguel Angel Torres" --encrypt $FICHERO

  # borrado fichero original
  # rm -f ~/$FICHERO

  # renombrado
  # mv $FICHERO.gpg $FECHA-$FICHERO.gpg

  # borrado fichero encriptado
  # rm -f $FECHA-$FICHERO.gpg
done
```

bck-mysql.sh

```
#!/bin/bash

#
```

```
# APP
#

source /home/bck/bck.ini

# DBS=`mysql -u$DBUSER -h$DBHOST -e"show databases"`
DBS=`mysql -u$DBUSER -p"$DBPASSWD" -h"$DBHOST" -e"show databases"`
# existe version SED para eliminar /Database/

for DATABASE in $DBS
do
        if [ $DATABASE != "Database" ]; then
                FILENAME=$FECHA-$HOSTNAME-mysql-$DATABASE.sql.gz
                mysqldump -u$DBUSER -p"$DBPASSWD" -h"$DBHOST" $DATABASE | gzip
--best > $LOCALDIR$FILENAME
        fi
done

chmod 400 $LOCALDIR*.gz

#tar --remove-files -cf $TARPREFIX-$SUFFIX.tar $SUFFIX-*.gz
tar --remove-files -cf $LOCALDIR$FECHA-$HOSTNAME-mysql.tar $LOCALDIR$FECHA-
$HOSTNAME-mysql*.gz

exit 0
```

## mejoras

- parametrizar ficheros que se encriptan
- quitar GZIP del mysqldump y hacer un TAR -cz ¿?