

Capítol 9

Còpies de seguretat de les dades
Nivells de redundància: còpies locals i en xarxa
Automatització del procés

9.1 Seguretat de les dades del servidor: còpies de seguretat

Segurament la seguretat de les dades és un dels problemes més importants si s'utilitza un sistema informàtic amb normalitat. Una empresa o institució que desplegui tot el potencial d'un sistema informàtic en xarxa necessita una fiabilitat molt important respecte a que les dades (informacions sensibles, documents, configuracions, transaccions, comptabilitats i altres dades “crítiques”) estiguin convenientment assegurades. No cal oblidar que, en definitiva, un servidor és una màquina formada per moltes parts i que en un moment donat pot fer fallida ... en realitat “segur” que farà fallida si el seu ús és intensiu (24 hores/365 dies). Per experiència pròpia podria dir que els components de hardware que fan fallida en el temps per un servidor de gran disponibilitat són, per ordre de més a menys:

1. Les fonts d'alimentació. Això no implica cap problema important ja que es canvia la font i el sistema ha de ser capaç de ficar-se en marxa. Pot ser la forma d'evitar-ho és utilitzar fonts d'alimentació redundants i intercanviables en calent. Això encareix poc el cost del servidor i permet seguir amb les operacions “normals” del sistema sense cap interrupció observable per l'usuari.
2. Els discs durs. Aquest és el mal son dels administradors de sistemes. Una configuració en RAID 1 (espill) evita gran part dels problemes ja que permet el funcionament del sistema amb 1 dels disc durs trencat. Encara així, per substituir el disc dur trencat caldrà fer una parada tècnica del sistema. Ho podem evitar utilitzant discs durs de gran qualitat (SCSI enlloc de SATA o PATA) i que siguin intercanviables en calent, però això sí que encareix significativament el servidor. Una altra alternativa, o millor un afegit a la seguretat de les dades, és realitzar còpies de seguretat de les dades crítiques del sistema. Això és el que explicaré en aquest capítol.
3. Targetes de xarxa, targetes PCI i altres components interns. Impliquen una parada tècnica obligatòria. Com que són components relativament barats és interessant utilitzar els de millor qualitat per un servidor d'alta disponibilitat.
4. Plaques base. Com al cas anterior impliquen una parada tècnica obligatòria. Són components que haurien de ser de la millor qualitat possible.
5. Processadors. De moment, durant el temps que he estat treballant en aquest camp, no m'he trobat el cas d'haver de canviar un processador d'un servidor. Implica una parada tècnica obligatòria.

Les còpies de seguretat són un afegit a la seguretat de les dades del servidor però també poden ser un servei addicional de les prestacions de la nostra xarxa. Per entendre aquest fet ens cal diferenciar entre els tipus de còpies de seguretat que podem fer i la redundància de dades que impliquen. Per una administrador de sistemes, possiblement, una còpia de seguretat “instantània” seria prou. Això vol dir que en fer una còpia de les dades una vegada el sistema funciona de manera raonable i estable tenim prou per poder tornar a recuperar l'estat inicial en cas de pèrdua d'algun component. Però clar, el pensament d'un usuari estàndard de vegades és diferent. Imaginem que el departament de comptabilitat d'una empresa fa còpies de seguretat: segur que farien còpies de seguretat instantànies però de diferents dies. Això vol dir que, per exemple, durant tota una setmana, es faran còpies de seguretat dia per dia i es conservaran. Aquesta forma d'actuar implica que el suport sobre el que es conserven les còpies s'ha de multiplicar pel número de dies “diferents” dels que es vol conservar còpia de seguretat. El motiu es senzill d'entendre. en un moment donat ens pot interessar tornar a l'estat d'un dia anterior al dia en que s'ha realitza la darrera còpia de seguretat. Aquesta necessitat no és estranya. Hi ha quantitat d'empreses i organismes que necessiten realitzar les còpies de seguretat d'aquesta forma. El motiu deixa de ser la possible pèrdua de dades per mal funcionament del servidor i passa a ser un motiu de funcionament normal de l'empresa u organisme. Aquesta idea ens porta a la necessitat de configurar un sistema de còpies de seguretat que s'adapti a les necessitats dels usuaris de la nostra xarxa. Per fer-ho tenim alguns mecanismes interessants dins

dels sistemes UNIX/Linux. La programació de shell-script i la utilització de comandes com ara *tar*, *gz*, *rsync* i *cron* ens permetran automatitzar en gran part tot el procés de còpies de seguretat del servidor.

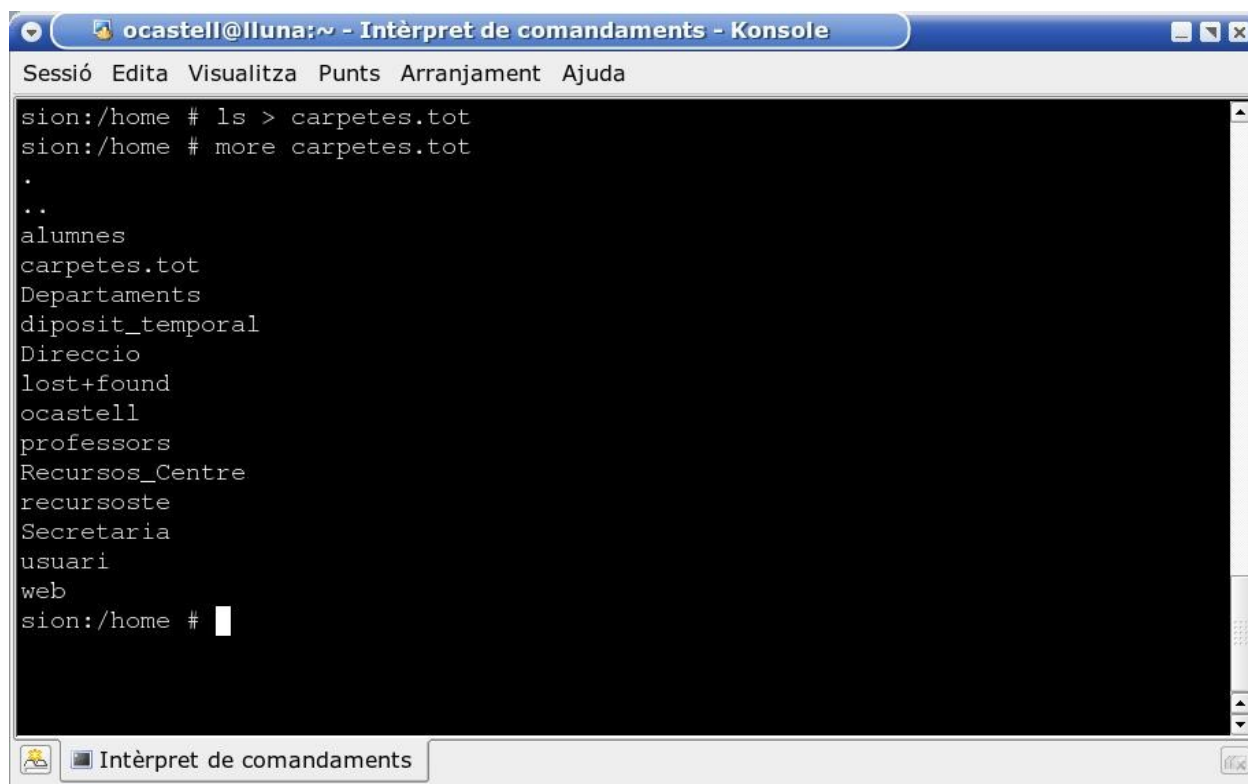
9.2 Còpies de seguretat locals

La primera aproximació a les còpies de seguretat és realitzar una còpia en fitxers comprimits dels directoris i documents “sensibles” en un disc dur del mateix servidor. Aquest disc dur convé que sigui diferent al disc dur utilitzat pel funcionament del sistema i pot ser intern o extern (els discs USB o Firewire donen molts bons resultats).

Imaginem que tenim un disc dur addicional al nostre sistema i que s'ha creat una partició en aquest disc que es munta sobre la carpeta */backup*. Aquí és on farem les còpies de seguretat. Ens interessa crear còpies diàries de les carpetes personals dels usuaris i d'uns carpetes que corresponen a recursos compartits del nostre servidor SAMBA.

Totes les dades es troben sota */home* però no volem fer còpies de seguretat de totes les carpetes que hi ha sota */home* ja que tenim una zona de dipòsit temporal on els usuaris poden abocar tot d'informació que es manté durant un cert temps. Ens caldrà un fitxer on especifiquem les carpetes de les que volem fer les còpies de seguretat i altre fitxer on introduïrem els usuaris dels quals volem fer còpia de seguretat.

Mirem què fem per crear els fitxers amb les carpetes que volem fer còpies de seguretat:



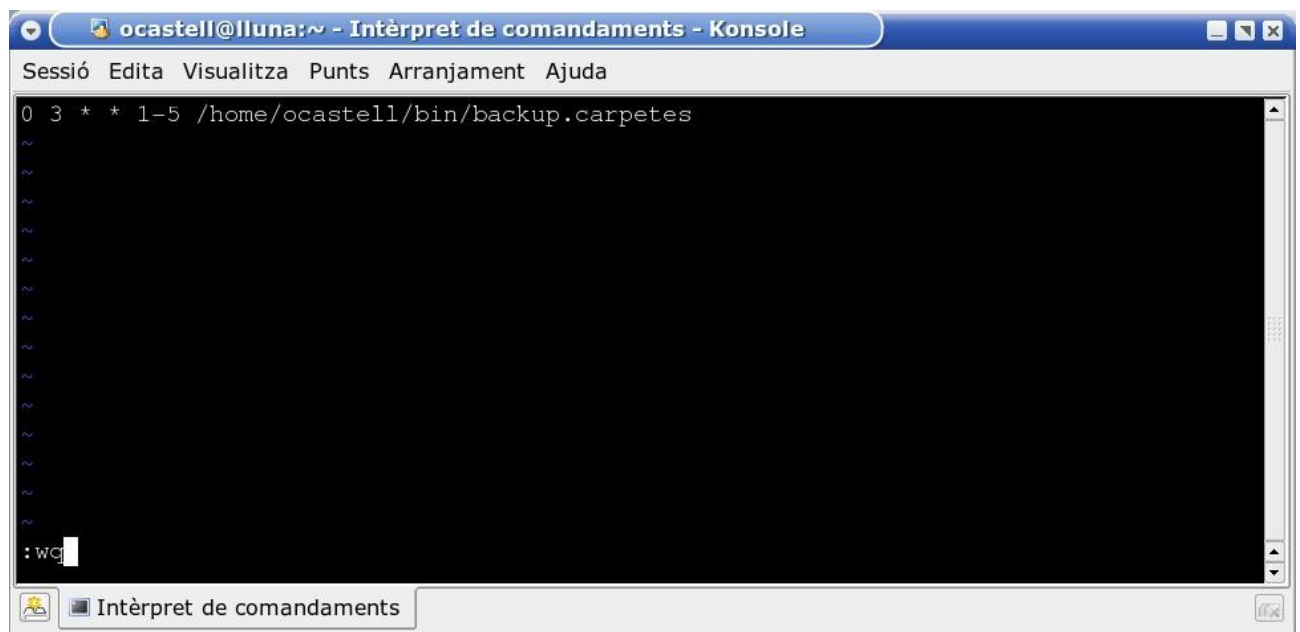
```
o castell@lluna:~ - Intèrpret de comandaments - Konsole
Sessió Edita Visualitza Punts Arranjament Ajuda
sion:/home # ls > carpetes.tot
sion:/home # more carpetes.tot
.
..
alumnes
carpetes.tot
Departaments
diposit_temporal
Direccio
lost+found
o castell
professors
Recursos_Centre
recursoste
Secretaria
usuari
web
sion:/home #
```

D'aquest fitxer volem esborrar les dues primeres línies (. i ..), la quarta línia que és el mateix fitxer, la sisena línia ja que no volem fer còpia de la zona d'scratch, la vuitena (*lost+found*), la desena línia que correspon a la carpeta de professors, ja que conté les carpetes personals dels professors i volem fer còpia independent de cadascuna, i finalment, la línia que es refereix a la carpeta usuari ja que no conté cap informació important que ens interessi conservar.

Ara tenim un fitxer el fitxer amb el nom *carpetes.tot* que conté els noms de les carpetes de les que volem fer còpia de seguretat. Mirem el shell-script que utilitzarem per fer les còpies de seguretat:

```
#!/bin/bash
#
# Construïm el nom del fitxer backup_DiaMes
#
export LANG=ca_ES
nom1="backup_Carpetes"
nom2=`date +"%d%B"`
#
# Generem la copia i es compacta
#
cd /home/
directoris=`more /home/carpetes.tot`
for i in $directoris
do
nom="$i$nom2"
tar -cvf /home/$nom.tar $i 2>>/dev/null
gzip $nom.tar
mv /home/$nom.tar.gz /backup/Carpetes
done
#
# Còpia acabada amb èxit
#
exit
```

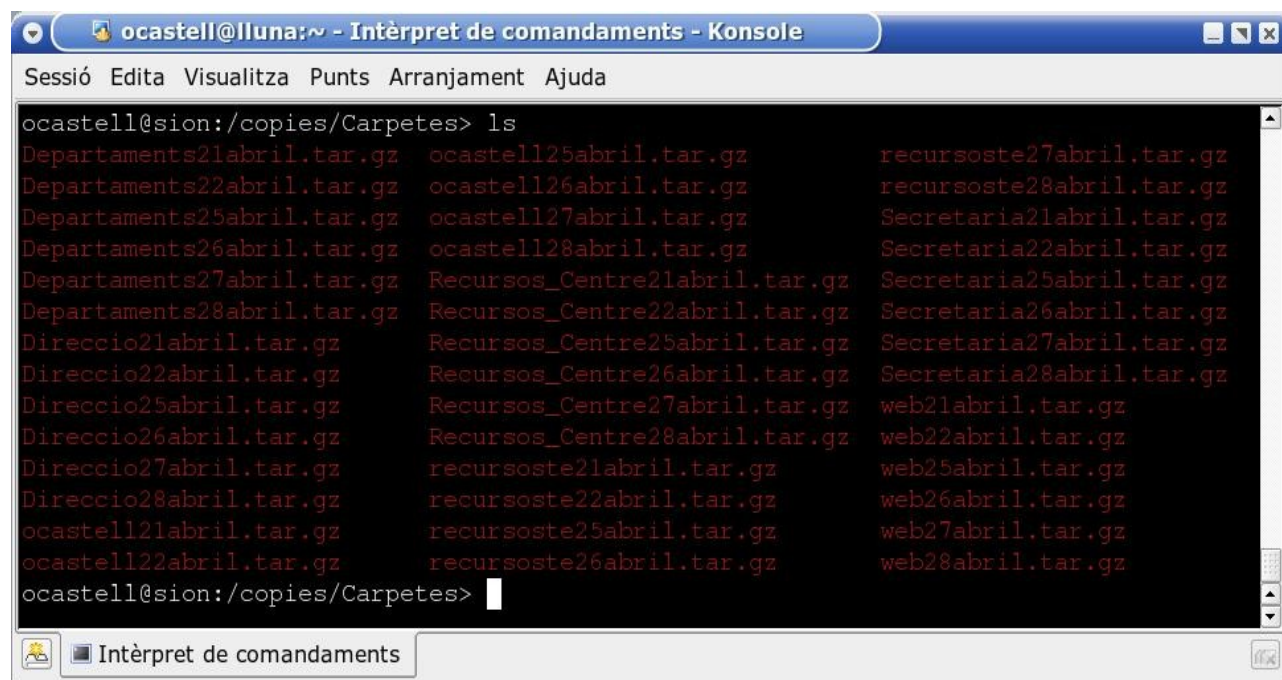
L'anterior seqüència de comandes crearà un fitxer amb el nom de la carpeta que volem fer còpia a la que s'adjunta la data del dia i que estarà comprimit. Ara cal fer que la seqüència de comandes s'executi cada cert temps de manera automàtica. En concret decidim d'executar-ho cada dia laborable a les 3 de la matinada. Per automatitzar el procés utilitzem la comanda *crontab -e* i afegim una entrada. La seqüència de comandes anterior s'ha guardat en un fitxer anomenat *backup.carpetes* sota el directori */home/ocastell/bin/*, per tant tindrem:



Ara mirem de fer el mateix amb la zona de professors. Crearem un fitxer amb tots els noms de les carpetes del professorat (sota /home/professors/) i crearem un shell-script semblant a l'anterior per generar les còpies de seguretat. Aquest seria una cosa semblant a això:

```
#!/bin/bash
#
# Construïm el nom del fitxer backup_DiaMes
#
export LANG=ca_ES
nom1="backup_Documents"
nom2=`date +"%d%B"`
#
# Generem la còpia i es compacta
#
cd /home/professors/
directoris=`more /home/professors/usuarios`
for i in $directoris
do
nom="$i$nom2"
tar -cvf $nom.tar $i 2>>/dev/null
gzip $nom.tar
mv /home/professors/$nom.tar.gz /backup/Usuaris/professors/
done
rm /home/professors/usuarios*.tar.gz
#
# Còpia acabada amb èxit
#
exit
```

Mirem què cal esperar a veure sota el directori /copies/Carpetes (o /copies/Usuaris) al cap d'uns dies de fer còpies de seguretat:



```
oacastell@lluna:~ - Intèrpret de comandaments - Konsole
Sessió Edita Visualitza Punts Arranjament Ajuda
oacastell@sion:/copies/Carpetes> ls
Departaments21abril.tar.gz  oacastell25abril.tar.gz  recursoste27abril.tar.gz
Departaments22abril.tar.gz  oacastell26abril.tar.gz  recursoste28abril.tar.gz
Departaments25abril.tar.gz  oacastell27abril.tar.gz  Secretaria21abril.tar.gz
Departaments26abril.tar.gz  oacastell28abril.tar.gz  Secretaria22abril.tar.gz
Departaments27abril.tar.gz  Recursos_Centre21abril.tar.gz  Secretaria25abril.tar.gz
Departaments28abril.tar.gz  Recursos_Centre22abril.tar.gz  Secretaria26abril.tar.gz
Direccio21abril.tar.gz      Recursos_Centre25abril.tar.gz  Secretaria27abril.tar.gz
Direccio22abril.tar.gz      Recursos_Centre26abril.tar.gz  Secretaria28abril.tar.gz
Direccio25abril.tar.gz      Recursos_Centre27abril.tar.gz  web21abril.tar.gz
Direccio26abril.tar.gz      Recursos_Centre28abril.tar.gz  web22abril.tar.gz
Direccio27abril.tar.gz      recursoste21abril.tar.gz  web25abril.tar.gz
Direccio28abril.tar.gz      recursoste22abril.tar.gz  web26abril.tar.gz
oacastell21abril.tar.gz     recursoste25abril.tar.gz  web27abril.tar.gz
oacastell22abril.tar.gz     recursoste26abril.tar.gz  web28abril.tar.gz
oacastell@sion:/copies/Carpetes>
```


9.3 Com crear un servidor de rèplica amb Linux. Utilitat per fer còpies de seguretat en xarxa

Disposem d'un servidor Linux on es centralitza la gestió d'usuaris, la web, el correu electrònic, recursos compartits amb samba, connectivitat dels usuaris des de l'exterior a la intranet i altres serveis de menor importància. Com és normal, tot servidor que estigui 24h/365 dies en funcionament necessita d'un manteniment regular, ja sigui per actualitzacions, reposició de hardware, manteniment d'usuaris, etc ... i no cal descartar altres problemes. El cas és que durant el lapsus de temps que es realitza el manteniment els serveis de xarxa estaran fora de l'abast dels usuaris amb els inconvenients que això pot comportar. La solució pot ser crear un segon servidor que substitueixi de manera ràpida i eficient al servidor principal: això vol dir que tingui tota la configuració i la informació del primer servidor en temps real i en qualsevol moment. El podem anomenar servidor de rèplica. El tema es podria identificar com la realització d'una imatge actualitzada en temps real del primer servidor cap al servidor de rèplica. El cas que us explicaré està basat en un servidor SuSE 9.2 corrent sobre un processador XEON amb dos discs durs de 200 GB en mirall (RAID de software)+un disc dur addicional per fer còpies de seguretat i tots els serveis de xarxa configurats (DHCP, SAMBA i com a PDC, APACHE, SSH, POSTFIX, ...).

Per crear el servidor de rèplica cal un altre ordinador amb un disc dur senzill amb la capacitat suficient per albergar el sistema i la zona d'usuaris del servidor principal, en el cas que us comento un PIV amb un disc de 120 GBy. No cal dir que tots dos ordinadors incorporen targetes de xarxa i estan connectats a un mateix switch sense tallafocs que els emmascaren. El servidor principal té com adreça IP 192.168.0.2 i el servidor de rèplica té com adreça IP 192.168.0.4.

Les eines del sistema que utilitzarem per fer la sincronització de tots dos servidors són: SSH, rsync i cron. Com ja he comentat SSH està configurat per defecte (encara que ens caldrà fer unes petites modificacions) i rsync s'instal·la també amb el sistema mínim. L'únic que 'ha de comprovar és que tots dos serveis estiguin instal·lats i en funcionament. Podem mirar si els tenim a */etc/init.d* (són *sshd* i *rsyncd*). Per veure si estan engegats podem usar el flag status (*# sshd status* o *# rsyncd status*)... Bé, dono per entès que ja sabeu com fer que s'engeguin en iniciar el sistema.

El servidor de rèplica pot estar verge (és a dir amb el disc dur intacte) o pot tenir la instal·lació del sistema SuSE 9.2 que s'ha realitzat al servidor principal. En el cas que us explico, per anar més ràpid i ja que el sistema principal feia temps que estava en marxa, el que he fet és una imatge del disc dur sobre un disc exterior USB amb prou capacitat. Això ho podeu fer amb GHOST per exemple, jo ho he fet bolcant (comanda *dd*) les particions una per una a fitxers ... hi ha moltes eines que permeten fer les imatges i no és el que voldria tractar aquí. Una vegada feta la imatge del servidor principal restaurem en el servidor de rèplica. Ens podem trobar amb l'inconvenient que una vegada restaurada la imatge el servidor no engegui ja que no troba un sector d'arrancada. Per resoldre-ho cal instal·lar un BOOTLOADER (Lilo o grub) utilitzant un disquet d'arrancada o el primer CD de la distribució. També s'ha de tenir en compte de canviar la IP del servidor de rèplica a 192.168.0.4 i el seu nom perquè no coincideixi amb el del servidor principal (també el nom del PDC de SAMBA no cal oblidar-lo). Ara ja podem connectar el servidor de rèplica a la xarxa. En aquests moments disposem de dos servidors idèntics encara que no del tot, ja que, durant el procés, segur que els usuaris de la xarxa han modificat algun dels seus documents. Mirem com tenir en compte aquestes modificacions. El que farem serà una sincronització de dades entre tots dos servidors. Per mantenir sincronitzats els fitxers (i carpetes) de tots dos servidors utilitzarem rsync.

Com a exemple inicial i per comprovar que tot funciona correctament farem el següent: des del servidor principal executarem la comanda (naturalment com a root)

```
# rsync -praulHogt -e "ssh -lroot -p22" /home/professors/ root@192.168.0.4:/home/professors/
```

Ens demanarà la paraula clau de root, l'introduïm i comprovem que al cap d'uns moments acaba de fer la sincronització. Això vol dir que qualsevol canvi que s'hagi realitzat en el directori */home/professors* del servidor principal ara està idèntic en el servidor de rèplica. Podem fer alguna comprovació, per exemple crear un fitxer en el directori */home/professors/*, fer la sincronització i comprovar que s'ha creat aquest fitxer en el servidor de rèplica. Ara ja sabem i tenim constància de què el sistema funciona. Cal decidir quins són els directoris i fitxers que interessa tenir sincronitzats en tots dos servidors. Cal anar amb molt de compte ja que no "tot" ha de ser idèntic en tots dos servidors. Com a exemple us escric un shell-script que faig anar per la sincronització i que caldrà que cadascú personalitzi en el seu cas:

```
# Fitxer de sincronització entre servidor principal i mirall
#
# Sincronització de la zona d'Usuaris
#
rsync -praulHogt -e "ssh -lroot -p22" /home/recursoste/ root@192.168.0.4:/home/recursoste/
rsync -praulHogt -e "ssh -lroot -p22" /home/ocastell/ root@192.168.0.4:/home/ocastell/
rsync -praulHogt -e "ssh -lroot -p22" /home/web/ root@192.168.0.4:/home/web/
rsync -praulHogt -e "ssh -lroot -p22" /home/professors/ root@192.168.0.4:/home/professors/
rsync -praulHogt -e "ssh -lroot -p22" /home/alumnes/ root@192.168.0.4:/home/alumnes/
rsync -praulHogt -e "ssh -lroot -p22" /home/Secretaria/ root@192.168.0.4:/home/Secretaria/
rsync -praulHogt -e "ssh -lroot -p22" /home/Direccio/ root@192.168.0.4:/home/Direccio/
rsync -praulHogt -e "ssh -lroot -p22" /home/Departaments/ root@192.168.0.4:/home/Departaments/
#
# Sincronització de fitxers i directoris importants sota /etc
#
rsync -praulHogt -e "ssh -lroot -p22" /etc/passwd root@192.168.0.4:/etc/passwd
rsync -praulHogt -e "ssh -lroot -p22" /etc/shadow root@192.168.0.4:/etc/shadow
rsync -praulHogt -e "ssh -lroot -p22" /etc/group root@192.168.0.4:/etc/group
rsync -praulHogt -e "ssh -lroot -p22" /etc/named.conf root@192.168.0.4:/etc/named.conf
rsync -praulHogt -e "ssh -lroot -p22" /etc/samba/ root@192.168.0.4:/etc/samba/
rsync -praulHogt -e "ssh -lroot -p22" /etc/squid/ root@192.168.0.4:/etc/squid/
rsync -praulHogt -e "ssh -lroot -p22" /etc/apache2/ root@192.168.0.4:/etc/apache2/
rsync -praulHogt -e "ssh -lroot -p22" /etc/postfix/ root@192.168.0.4:/etc/postfix/
rsync -praulHogt -e "ssh -lroot -p22" /etc/init.d/squid root@192.168.0.4:/etc/init.d/squid
#
# Sincronització de fitxers i directoris importants sota /var
#
rsync -praulHogt -e "ssh -lroot -p22" /var/lib/named/ root@192.168.0.4:/var/lib/named/
rsync -praulHogt -e "ssh -lroot -p22" /var/lib/samba/ root@192.168.0.4:/var/lib/samba/
rsync -praulHogt -e "ssh -lroot -p22" /var/spool/ root@192.168.0.4:/var/spool/
```

En executar el shell-script anterior en el servidor principal obtindrem una còpia actualitzada al servidor de rèplica. Però clar, això ho hem d'executar a mà i el que interessa és que el sistema sigui totalment automàtic, com ho fem? El primer problema que ens trobem per fer la sincronització automàtica és haver d'escriure el *password* de *root*. Per evitar això utilitzarem les eines que fica al nostre abast el paquet *ssh*. Crearem un parell de claus úniques que facin que la identificació del servidor principal al servidor de rèplica sigui automàtica. Com ho fem:

- a.- Generem la clau al servidor principal:


```
# ssh-keygen -t dsa
```
- b.- Copiem la clau generada al servidor de rèplica:


```
# ssh-copy-id -i /root/.ssh/id_dsa root@192.168.0.4
```
- c.- Comprovem que tot ha funcionat bé:


```
# ssh 192.168.0.4 (ja no ens demana la clau)
```

No cal explicar a fons el que fa cadascuna de les comandes i les possibilitats que tenen. Mireu la documentació del servidor *ssh* per veure les diferents tipus de claus que existeixen i les diferents formes de generar-les. Queda a discreció de l'usuari.

Per assegurar que el servidor de rèplica està actualitzat en tot moment de manera automàtica utilitzarem el cron per fer que l'script que hem creat s'executi cada cert temps. En el meu cas faig l'actualització cada dues hores durant el dia i més espaiat (cada 6) a la nit. Afegirem la següent entrada (*crontab -e*):

```
0 0,6,8,10,12,14,16,18,20,22 * * * /home/ocastell/bin/sincronisme
```

Evidentment l'adreça i el nom del fitxer seran els que ja heu personalitzat per el vostre cas. El temps per realitzar el sincronisme és petit, per tant, podem ficar que la sincronització es faci cada hora o cada mitja hora sense problemes. En el cas que jo he tractat amb més de 1300 alumnes, 140 professors i gran quantitat d'informació (unes 100 GBy a la zona d'usuaris) la sincronització està al voltant de 10 minuts (ràpid no?).

Mirem ara què cal fer per aturar el servidor principal:

- Que primer ens assegurem que s'ha realitzat correctament el darrer sincronisme. Si són les 15:00 hores, per exemple, en el meu cas no tindrè cap problema.
- Desconnecto el servidor rèplica de la xarxa, canvio el nom, l'IP i el nom SAMBA del PDC perquè coincideixin amb el del servidor principal.
- Faig un *shutdown* del servidor principal.
- Torno a connectar el servidor rèplica a la xarxa. Comprovo que tot ha anat com cal, m'identifico en un client, visito la web, etc ...

Tot el procés pot durar uns deu minuts. Ara ja ens podem emportar el servidor principal per fer el manteniment, canviar discs durs ... el que calgui. El sistema és com tenir una roda de recanvi al cotxe. La roda no cal que sigui massa gran, lo just per suportar temporalment la desconnexió del servidor principal. Una vegada fet el manteniment al servidor principal voldrem que tot quedi com abans. Queda clar que si el manteniment ha durat una setmana, tota la variació en la informació està reflectida únicament en el servidor de rèplica no pas en el servidor principal. Què fem?, el procés contrari en direcció, passarem dades des del servidor de rèplica al principal. Per exemple, a les 15:00 hores com abans:

- Desconnecto el servidor rèplica de la xarxa.
- Canvio el nom, l'IP i el nom SAMBA del PDC perquè siguin diferents a les del servidor principal. Recordem que la IP tornarà a ser 192.168.0.4.
- Connecto el servidor principal (i l'engego) i el servidor de rèplica a la xarxa.
- Ara cal fer el procés contrari de sincronisme des del servidor de rèplica al servidor principal.

Aquí no val la pena produir el parell de claus (i per motius de seguretat no és aconsellable). Al servidor de rèplica cal executar un shell-script com el següent:

```
# Fitxer de recuperació del sincronisme entre servidor mirall i principal
#
# Sincronització de la zona d'Usuaris
#
rsync -praulHogt -e "ssh -lroot -p22" /home/recursoste/ root@192.168.0.2:/home/recursoste/
rsync -praulHogt -e "ssh -lroot -p22" /home/ocastell/ root@192.168.0.2:/home/ocastell/
rsync -praulHogt -e "ssh -lroot -p22" /home/web/ root@192.168.0.2:/home/web/
```

```
rsync -praulHogt -e "ssh -lroot -p22" /home/professors/ root@192.168.0.2:/home/professors/
rsync -praulHogt -e "ssh -lroot -p22" /home/alumnes/ root@192.168.0.2:/home/alumnes/
rsync -praulHogt -e "ssh -lroot -p22" /home/Secretaria/ root@192.168.0.2:/home/Secretaria/
rsync -praulHogt -e "ssh -lroot -p22" /home/Direccio/ root@192.168.0.2:/home/Direccio/
rsync -praulHogt -e "ssh -lroot -p22" /home/Departaments/ root@192.168.0.2:/home/Departaments/
#
# Sincronització de fitxers i directoris importants sota /etc
#
rsync -puralHogt -e "ssh -lroot -p22" /etc/passwd root@192.168.0.2:/etc/passwd
rsync -puralHogt -e "ssh -lroot -p22" /etc/shadow root@192.168.0.2:/etc/shadow
rsync -puralHogt -e "ssh -lroot -p22" /etc/group root@192.168.0.2:/etc/group
rsync -puralHogt -e "ssh -lroot -p22" /etc/named.conf root@192.168.0.2:/etc/named.conf
rsync -puralHogt -e "ssh -lroot -p22" /etc/samba/ root@192.168.0.2:/etc/samba/
rsync -puralHogt -e "ssh -lroot -p22" /etc/squid/ root@192.168.0.2:/etc/squid/
rsync -puralHogt -e "ssh -lroot -p22" /etc/apache2/ root@192.168.0.2:/etc/apache2/
rsync -puralHogt -e "ssh -lroot -p22" /etc/postfix/ root@192.168.0.2:/etc/postfix/
rsync -puralHogt -e "ssh -lroot -p22" /etc/init.d/squid root@192.168.0.2:/etc/init.d/squid
#
# Sincronització de fitxers i directoris importants sota /var
#
rsync -puralHogt -e "ssh -lroot -p22" /var/lib/named/ root@192.168.0.2:/var/lib/named/
rsync -puralHogt -e "ssh -lroot -p22" /var/lib/samba/ root@192.168.0.2:/var/lib/samba/
rsync -puralHogt -e "ssh -lroot -p22" /var/spool/ root@192.168.0.2:/var/spool/
```

Executem l'anterior shell-script i anem ficant el password de root cada vegada que ens ho demana. En acabar el procés el servidor principal tindrà la darrera informació que s'ha escrit en el servidor de rèplica mentre ha estat com a servidor principal.

9.4 Còpies de seguretat en xarxa utilitzant rsync

El tema de les còpies de seguretat és molt complex. Hi ha solucions per tots els tipus d'entorn i de nivell de redundància que es vulgui utilitzar. Una petita reflexió:

- a) Disposo de dos servidors més a la xarxa, més petits, que donen servei a alguns departaments i a la gestió del centre. No puc crear un servidor de rèplica per cada ordinador però si voldria estar segur que no es perd cap informació important dels servidors.
- b) En el nostre cas es fa còpia de tot el sistema via RAID 1, per tant estem coberts en cas de fallida d'un dels discos ... però, i si fallen tots dos discos? i si es crema la placa base del servidor? i si ...?
- c) Sempre podem fer una còpia en un disc redundant dins del mateix servidor.
- d) O.K. ja fem la còpia compactada amb .tar.gz de les zones crítiques del sistema i de la informació dels usuaris. Si el servidor fa fallida caldrà obrir la caixa per treure el disc redundant i ficar-lo en un altre ordinador per poder recuperar la informació. És lent i incòmode.
- e) Podem utilitzar un disc USB extern per fer les còpies de seguretat i automatitzar-lo amb cron. O.K. és una bona solució, però cal deixar el disc USB a disposició del servidor en tot moment i jo el vull utilitzar per altres feines. Els discos USB externs són cars ...
- f) Per què no fem una còpia de la informació en el servidor rèplica? O.K.

El procés és semblant a l'anterior. En el servidor rèplica crearem un directori anomenat `copies_seguretat` i dins d'aquest dos directoris que es diran `servidor_professors` i `servidor_gestio`.

Fem el procés per tal que root es pugui connectar sense password al servidor de rèplica. Creem un script per automatitzar el sincronisme, per exemple al servidor de professorat:

```
#
# Còpies en xarxa
#
rsync -purahogt -e "ssh -lroot -p22" /home/Departaments/
root@192.168.0.4:/copies_seguretat/servidor_professorat/Departaments/
rsync -purahogt -e "ssh -lroot -p22" /home/Recursos_Centre/
root@192.168.0.4:/copies_seguretat/servidor_professorat/Recursos_Centre/
rsync -purahogt -e "ssh -lroot -p22" /home/professors/
root@192.168.0.4:/copies_seguretat/servidor_professorat/professors/
rsync -purahogt -e "ssh -lroot -p22" /home/ocastell/
root@192.168.0.4:/copies_seguretat/servidor_professorat/ocastell/
rsync -purahogt -e "ssh -lroot -p22" /home/web/
root@192.168.0.4:/copies_seguretat/servidor_professorat/web/
```

Amb el cron faig que es sincronitzi cada cert temps, per exemple cada 2 hores com ja s'ha explicat abans.

Referències:

- [1] Comanda tar. Manual en línia: http://www.gnu.org/software/tar/manual/html_mono/tar.html
- [2] Comanda gzip. Manual en línia: <http://www.gnu.org/software/gzip/manual/gzip.html>
- [3] Comanda cron. Manual en línia: <http://www.gnu.org/software/gzip/manual/gzip.html>
- [4] Comanda rsync. Manual en línia: <http://samba.anu.edu.au/rsync/documentation.html>
- [5] Programació shell. Curs on-line: <http://www.ciberdroide.com/misc/novato/curso/>