Capítulo 1. Copias de seguridad

Tabla de contenidos

Introducción

Las copias de seguirdad del sistema son con frecuencia el único mecanismo de recuperación que poseen los administradores para restaurar una máquina que por cualquier motivo - no siempre se ha de tratar de un pirata que borra los discos - ha perdido datos. Por tanto, una correcta política para realizar, almacenar y, en caso de ser necesario, restaurar los backups es vital en la planificación de seguridad de todo sistema.

Asociados a los backups suelen existir unos problemas de seguridad típicos en muchas organizaciones. Por ejemplo, uno de estos problemas es la no verificación de las copias realizadas: el administrador ha diseñado una política de copias de seguridad correcta, incluso exhaustiva en muchas ocasiones, pero nadie se encarga de verificar estas copias...hasta que es necesario restaurar ficheros de ellas. Evidentemente, cuando llega ese momento el responsable del sistema se encuentra ante un gran problema, problema que se podría haber evitado simplemente teniendo la precaución de verificar el correcto funcionamiento de los backups; por supuesto, restaurar una copia completa para comprobar que todo es correcto puede ser demasiado trabajo para los métodos habituales de operación, por lo que lo que se suele hacer es tratar de recuperar varios ficheros aleatorios del backup, asumiendo que si esta recuperación funciona, toda la copia es correcta.

Otro problema clásico de las copias de seguridad es la política de etiquetado a seguir. Son pocos los administradores que no etiquetan los dispositivos de backup, algo que evidentemente no es muy útil: si llega el momento de recuperar ficheros, el operador ha de ir cinta por cinta (o disco por disco, o CD-ROM por CD-ROM...) tratando de averiguar dónde se encuentran las últimas versiones de tales archivos. No obstante, muchos administradores siguen una política de etiquetado exhaustiva, proporcionando todo tipo de detalles sobre el contenido exacto de cada medio; esto, que en principio puede parecer una posición correcta, no lo es tanto: si por cualquier motivo un atacante consigue sustraer una cinta, no tiene que investigar mucho para conocer su contenido exacto, lo que le proporciona acceso a información muy concreta (y muy valiosa) de nuestros sistemas sin ni siquiera penetrar en ellos. La política correcta para etiquetar los backups ha de ser tal que un administrador pueda conocer la situación exacta de cada fichero, pero que no suceda lo mismo con un atacante que roba el medio de almacenamiento; esto se consigue, por ejemplo, con códigos impresos en cada etiqueta, códigos cuyo significado sea conocido por los operadores de copias de seguridad pero no por un potencial atacante.

La ubicación final de las copias de seguridad también suele ser errónea en muchos entornos; generalmente, los operadores tienden a almacenar los backups muy cerca de los sistemas, cuando no en la misma sala. Esto, que se realiza para una mayor comodidad de los técnicos y para recuperar ficheros fácilmente, es un grave error: no hay más que imaginar cualquier desastre del entorno, como un incendio o una inundación, para hacerse una idea de lo que les sucedería a los backups en esos casos. Evidentemente, se destruirían junto a los sistemas, por lo que nuestra organización perdería toda su información; no obstante, existen voces que reivindican como correcto el almacenaje de las copias de seguridad junto a los propios equipos, ya que así se consigue centralizar un poco la seguridad (protegiendo una única estancia se salvaguarda tanto las máquinas como las copias). Lo habitual en cualquier organización suele ser un término medio entre ambas aproximaciones: por ejemplo, podemos tener un juego de copias de seguridad completas en un lugar diferente a la sala de operaciones, pero protegido y aislado como esta, y un juego para uso diario en la propia sala, de forma que los operadores tengan fácil la tarea de recuperar ficheros; también podemos utilizar armarios ignífugos que requieran de ciertas combinaciones para su

apertura (combinaciones que sólo determinado personal ha de conocer), si decidimos almacenar todos los backups en la misma estancia que los equipos.

Por último, >qué almacenar? Obviamente debemos realizar copias de seguridad de los archivos que sean únicos a nuestro sistema; esto suele incluir directorios como /etc/, /usr/local/ o la ubicación de los directorios de usuario (dependiendo del Unix utilizado, /export/home/, /users/, /home/...). Por supuesto, realizar una copia de seguridad de directorios como /dev/ o /proc/ no tiene ninguna utilidad, de la misma forma que no la tiene realizar backups de directorios del sistema como /bin/ o /lib/: su contenido está almacenado en la distribución original del sistema operativo (por ejemplo, los CD-ROMs que utilizamos para instalarlo).

Dispositivos de almacenamiento

Existen multitud de dispositivos diferentes donde almacenar nuestras copias de seguridad, desde un simple disco flexible hasta unidades de cinta de última generación. Evidentemente, cada uno tiene sus ventajas y sus inconvenientes, pero utilicemos el medio que utilicemos, éste ha de cumplir una norma básica: ha de ser estándar. Con toda probabilidad muchos administradores pueden presumir de poseer los streamers más modernos, con unidades de cinta del tamaño de una cajetilla de tabaco que son capaces de almacenar gigas y más gigas de información; no obstante, utilizar dispositivos de última generación para guardar los backups de nuestros sistemas puede convertirse en un problema: >qué sucede si necesitamos recuperar datos y no disponemos de esa unidad lectora tan avanzada? Imaginemos simplemente que se produce un incendio y desaparece una máquina, y con ella el dispositivo que utilizamos para realizar copias de seguridad. En esta situación, o disponemos de otra unidad idéntica a la perdida, o recuperar nuestra información va a ser algo difícil. Si en lugar de un dispositivo moderno, rápido y seguramente muy fiable, pero incompatible con el resto, hubiéramos utilizado algo más habitual (una cinta de 8mm., un CD-ROM, o incluso un disco duro) no tendríamos problemas en leerlo desde cualquier sistema Unix, sin importar el hardware sobre el que trabaja.

Algunas órdenes para realizar copias de seguridad

Aunque muchos clones de Unix ofrecen sus propias herramientas para realizar copias de seguridad de todo tipo (por ejemplo, tenemos mksysb y savevg/restvg en AIX, fbackup y frecover en HP-UX, bru en IRIX, fsphoto en SCO Unix, ufsdump/ufsrestore en Solaris...), casi todas estas herramientas suelen presentar un grave problema a la hora de recuperar archivos: se trata de software propietario, por lo que si queremos restaurar total o parcialmente archivos almacenados con este tipo de programas, necesitamos el propio programa para hacerlo. En determinadas situaciones, esto no es posible o es muy difícil: imaginemos un departamento que dispone de sólo una estación Silicon Graphics corriendo IRIX y pierde todos los datos de un disco, incluida la utilidad bru; si ha utilizado esta herramienta para realizar backups, necesitará otra estación con el mismo operativo para poder restaurar estas copias, lo que obviamente puede ser problemático.

Por este motivo, muchos administradores utilizan herramientas estándar para realizar las copias de seguridad de sus máquinas; estas herramientas suelen ser tan simples como un shellscript que se planifica para que automáticamente haga backups utilizando órdenes como tar o cpio, programas habituales en cualquier clon de Unix y que no presentan problemas de interoperabilidad entre diferentes operativos. De esta forma, si en la estación Silicon Graphics del ejemplo anterior se hubiera utilizado tar para realizar las copias de seguridad, éstas se podrían restaurar sin problemas desde una máquina SPARC corriendo Solaris, y transferir los ficheros de nuevo a la Silicon.

dump/restore

La herramienta clásica para realizar backups en entornos Unix es desde hace años dump, que vuelca sistemas de ficheros completos (una partición o una partición virtual en los sistemas que las soportan, como Solaris); restore se utiliza para recuperar archivos de esas copias. Se trata de una utilidad disponible en la mayoría de clones del sistema operativo8.1, potente (no diremos ‘sencilla’) y lo más importante: las copias son completamente compatibles entre Unices, de forma que por ejemplo podemos restaurar un backup realizado en IRIX en un

sistema HP-UX. Además, como veremos luego, la mayor parte de las versiones de dump permiten realizar copias de seguridad sobre máquinas remotas directamente desde línea de órdenes (en el caso que la variante de nuestro sistema no lo permita, podemos utilizar rdump/rrestore) sin más que indicar el nombre de máquina precediendo al dispositivo donde se ha de realizar la copia.

La sintaxis general de la orden dump es

```
dump opciones argumentos fs
```

donde 'opciones' son las opciones de la copia de seguridad, 'argumentos' son los argumentos de dichas opciones, y 'fs' es el sistema de ficheros a salvaguardar. Se trata de una sintaxis algo peculiar: mientras que lo habitual en Unix es especificar cada argumento a continuación de la opción adecuada (por ejemplo, 'find . -perm 700 -type f' indica un argumento '700' para la opción 'perm' y uno 'f' para 'type'), en la orden dump primero especificamos toda la lista de opciones y a continuación todos sus argumentos; no todas las opciones necesitan un argumento, y además la lista de argumentos tiene que corresponderse exactamente, en orden y número, con las opciones que los necesitan (por ejemplo, si 'find' tuviera una sintaxis similar, la orden anterior se habría tecleado como 'find . -perm -type 700 f'). AIX y Linux son los únicos Unices donde la sintaxis de dump (recordemos que en el primero se denomina backup) es la habitual.

Las opciones de 'dump' más utilizadas son las que se muestran en la tabla 7.2; en las páginas man de cada clon de Unix se suelen incluir recomendaciones sobre parámetros específicos para modelos de cintas determinados, por lo que como siempre es más que recomendable su consulta. Fijándonos en la tabla, podemos ver que la opción 'u' actualiza el archivo /etc/dumpdates tras realizar una copia de seguridad con éxito; es conveniente que este archivo exista antes de utilizar dump por primera vez (podemos crearlo con la orden touch), ya que si no existe no se almacenará información sobre las copias de seguridad de cada sistema de ficheros (información necesaria, por ejemplo, para poder realizar backups progresivos). En este archivo dump - la propia orden lo hace, el administrador no necesita modificar el archivo a mano...y no debe hacerlo - registra información de las copias de cada sistema de archivos, su nivel, y la fecha de realización, de forma que su aspecto puede ser similar al siguiente:

```
anita:~# cat /etc/dumpdates
/dev/dsk/c0d0s6  0 Thu Jun 22 05:34:20 CEST 2000
/dev/dsk/c0d0s7  2 Wed Jun 21 02:53:03 CEST 2000
anita:~#
```

Tabla 1.1. Opciones de la orden dump

Opción	Acción	Real-Argumentoizada
0-9	Nivel de la copia de seguridad	NO
u	Actualiza /etc/dumpdates al finalizar el backup	NO
f	Indica una cinta diferente de la usada por defecto	SÍ
b	Tamaño del bloque	SÍ
c	Indica que la cinta destino es un cartucho	NO
W	Ignora todas las opciones excepto el nivel del backup	NO

El uso de dump puede ser excesivamente complejo, especialmente en sistemas antiguos donde es incluso necesario especificar la densidad de la cinta en bytes por pulgada o su longitud en pies; no obstante, hoy en día la forma más habitual de invocar a esta orden es 'dump [1-9]ucf cinta fs', es decir, una copia de seguridad del sistema de ficheros recibido como argumento, de un determinado nivel y sobre la unidad de cinta especificada. Por ejemplo para realizar una copia de seguridad completa sobre la unidad de cinta /dev/rmt de la partición lógica /dev/dsk/c0d0s7, en Solaris podemos utilizar la orden siguiente (podemos ver que nos muestra mucha información sobre el progreso de nuestra copia de seguridad en cada momento):

```
anita:~# ufsdump 0cuf /dev/rmt /dev/dsk/c0d0s7
DUMP: Date of this level 0 dump: Thu Jun 22 10:03:28 2000
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/dsk/c0d0s7 (/export/home) to /dev/rmt
DUMP: mapping (Pass I) [regular files]
DUMP: mapping (Pass II) [directories]
DUMP: estimated 24523 blocks (118796KB)
DUMP: Writing 63 Kilobyte records
DUMP: dumping (Pass III) [directories]
DUMP: dumping (Pass IV) [regular files]
DUMP: level 0 dump on Thu Jun 22 10:05:31 CEST 2000
DUMP: 24550 blocks (118927KB) on 1 volume
DUMP: DUMP IS DONE
anita:~#
```

Para realizar copias remotas, como hemos dicho antes, no tenemos más que anteponer el nombre del sistema donde deseemos realizar el volcado al nombre del dispositivo donde se va a almacenar, separado de éste por el carácter ':'; opcionalmente se puede indicar el nombre de usuario en el sistema remoto, separándolo del nombre de máquina por '@':

```
anita:~# ufsdump 0cuf toni@luisa:/dev/st0 /dev/dsk/c0d0s7
```

Si estamos utilizando rdump, hemos de tener definido un nombre de máquina denominado 'dumphost' en nuestro archivo /etc/hosts, que será el sistema donde se almacene la copia remota. De cualquier forma (usemos dump, ufsdump o rdump), el host remoto ha de considerarnos como una máquina de confianza (a través de /etc/hosts.equiv o .rhosts), con las consideraciones de seguridad que esto implica.

>Cómo restaurar los backups realizados con dump? Para esta tarea se utiliza la utilidad restore (ufsrestore en Solaris), capaz de extraer ficheros individuales, directorios o sistemas de archivos completos. La sintaxis de esta orden es

```
restore opciones argumentos archivos
```

donde 'opciones' y 'argumentos' tienen una forma similar a 'dump' (es decir, toda la lista de opciones seguida de toda la lista de argumentos de las mismas, excepto en AIX y Linux, donde la notación es la habitual), y 'archivos' evidentemente representa una lista de directorios y ficheros para restaurar. En la tabla 7.3 se muestra un resumen de las opciones más utilizadas.

Tabla 1.2. Opciones de la orden restore

Opción	Acción realizada	Argumento
r	Restaura la cinta completa	NO
f	Indica el dispositivo o archivo donde está el backup	SÍ
i	Modo interactivo	NO
x	Extrae los archivos y directorios desde el directorio actual	NO
t	Imprime los nombres de los archivos de la cinta	NO

Por ejemplo, imaginemos que deseamos restaurar varios archivos de un backup guardado en el fichero ‘backup’; en primer lugar podemos consultar el contenido de la cinta con una orden como la siguiente (en Linux):

```
luisa:~# restore -t -f backup>contenido
Level 0 dump of /home on luisa:/dev/hda3
Label: none
luisa:~# cat contenido|more
Dump  date: Fri Jun 23 06:01:26 2000
Dumped from: the epoch
       2    .
      11    ./lost+found
   30761    ./lost+found/#30761
   30762    ./lost+found/#30762
   30763    ./lost+found/#30763
   30764    ./lost+found/#30764
   30765    ./lost+found/#30765
   30766    ./lost+found/#30766
   30767    ./lost+found/#30767
    4097    ./ftp
    8193    ./ftp/bin
          8194    ./ftp/bin/compress
          8195    ./ftp/bin/cpio
    8196    ./ftp/bin/gzip
    8197    ./ftp/bin/ls
    8198    ./ftp/bin/sh
    8199    ./ftp/bin/tar
          8200    ./ftp/bin/zcat
   12289    ./ftp/etc
   12290    ./ftp/etc/group
   Broken pipe
   luisa:~#
```

Una vez que conocemos el contenido de la copia de seguridad - y por tanto el nombre del archivo o archivos a restaurar - podemos extraer el fichero que nos interese con una orden como

```
luisa:~# restore -x -f backup ./ftp/bin/tar
```

```
You have not read any tapes yet.
Unless you know which volume your file(s) are on you should start
with the last volume and work towards the first.
Specify next volume #: 1
set owner/mode for '.'? [yn] n
luisa:~# ls -l ftp/bin/tar
---x--x--x  1 root    root     110668 Mar 21  1999 ftp/bin/tar
luisa:~#
```

Como podemos ver, la extracción se ha realizado a partir del directorio de trabajo actual; si quisiéramos extraer archivos en su ubicación original deberíamos hacerlo desde el directorio adecuado, o, en algunas versiones de restore, especificar dicho directorio en la línea de órdenes.

Una opción muy interesante ofrecida por restore es la posibilidad de trabajar en modo interactivo, mediante la opción 'i'; en este modo, al usuario se le ofrece un prompt desde el cual puede, por ejemplo, listar el contenido de una cinta, cambiar de directorio de trabajo o extraer archivos. El siguiente ejemplo (también sobre Linux) ilustra esta opción:

```
luisa:~# restore -i -f backup
restore > help
Available commands are:
      ls [arg] - list directory
      cd arg - change directory
      pwd - print current directory
      add [arg] - add 'arg' to list of files to be extracted
      delete [arg] - delete 'arg' from list of files to be extracted
      extract - extract requested files
      setmodes - set modes of requested directories
      quit - immediately exit program
      what - list dump header information
      verbose - toggle verbose flag (useful with "ls")
      help or '?' - print this list
If no 'arg' is supplied, the current directory is used
restore > ls
.:
ftp/      httpd/    httpsd/    lost+found/ samba/    toni/

restore > add httpd
restore > extract
You have not read any tapes yet.
Unless you know which volume your file(s) are on you should start
with the last volume and work towards the first.
Specify next volume #: 1
set owner/mode for '.'? [yn] n
restore > quit
luisa:~#
```

Como podemos ver, hemos consultado el contenido de la copia de seguridad, añadido el directorio httpd/ a la lista de ficheros a extraer (inicialmente vacia), y extraído dicho directorio a partir del actual. Este uso de restore proporciona una gran comodidad y facilidad de uso, ya que las órdenes en modo interactivo son muy sencillas.