

Programación y Administración de Sistemas

8. Copias de seguridad y restauración

Pedro Antonio Gutiérrez

Asignatura "Programación y Administración de Sistemas"
2º Curso Grado en Ingeniería Informática
Escuela Politécnica Superior
(Universidad de Córdoba)
pagutierrez@uco.es

19 de abril de 2015



- 1 Contenidos
- 2 Planes de prevención de catástrofes
 - Escenarios de pérdida de información
 - Consejos generales
 - Factores
 - Estrategias
- 3 Soportes de seguridad
- 4 Copias de seguridad y restauración
 - tar
 - cpio
 - dump
 - restore
- 5 Restauración de un sistema completo
- 6 Referencias



Planes de prevención de catástrofes

- En cualquier momento algún archivo/s serán totalmente ilegibles por algún motivo:
 - Se exige capacidad de recuperación.
- Las copias de seguridad dependen de la situación y es necesario determinar:
 - De qué archivos hacer la copia, dónde, cómo y cuándo...
- El administrador del sistema debe:
 - Planear e implementar un sistema de copias de seguridad.
 - Periódicamente, hacer copias de seguridad de los ficheros.
 - Guardar las copias de seguridad en un lugar seguro.



Planes de prevención de catástrofes

- La estrategia de copias de seguridad tiene que ser efectiva, para conseguir **seguridad**:
 - El tiempo empleado es un esfuerzo que prevé futuras pérdidas.
 - El dinero gastado se compensa al evitar el desastre que supone una pérdida de datos (que conlleva enormes pérdidas de trabajo y por tanto dinero).
- Tener en cuenta:
 - Capacidad restaurar el sistema entero o parte del mismo, en un tiempo aceptable.
 - Tiempo que tarda en hacerse la copia de seguridad.
 - Facilidad de recuperar algún fichero de forma independiente.



Escenarios de pérdida de información

- Causas:
 - Fuerzas mayores: desastres naturales, electricidad estática, ...
 - Errores de usuario.
 - Virus y software destructivo.
 - Personas malintencionadas.
 - Fallos mecánicos.
- Si valoramos los costes, merece la pena incluir mecanismos/dispositivos específicos para esta labor.



Escenarios de pérdida de información: errores humanos

- Comandos mal escritos:

```
1 $ rm foo *
```

- Errores durante el redireccionamiento y uso de tuberías:

```
1 $ cat fstab | sed 's/ext2/ext3' > fstab
```

- Usuarios con acceso de root:

- Los errores anteriores serían catastróficos si ocurrieran sobre directorios o archivos de sistema.



Escenarios de pérdida de información: prevención de errores humanos

- Medidas de prevención sencillas:

- Utilizar alias:

```
1 Alias rm='rm -i'
```

- Utilizar SVN o CVS, sistema de control de versiones:

- Conservan el archivo original y lleva un histórico de los cambios realizados sobre éste.

- Crear copias de seguridad personales.

- Utilizar sudo para limitar el acceso de los usuarios con privilegios de root:

- Se limitará el acceso únicamente a los comandos necesarios para que el usuario pueda llevar a cabo su tarea.



Virus y software destructivo

- Virus: programa que se adhiere a un ejecutable y se propaga a otros al mismo tiempo que realiza otra acción (desde escribir un mensaje hasta mezclar las tablas de particiones).
 - **Caballos de Troya**: Programas que se hacen pasar por otros, funcionando como éstos, pero además realizando otras operaciones como obtener y enviar contraseñas. El grado de destrucción depende de quien los ejecuta.
 - **Gusanos**: Programas que se aprovechan de las debilidades de un sistema para propagarse a otros.
 - **Software destructivo**: Aplicaciones no mal intencionadas pero con errores de programación que pueden ser muy dañinos.
- Linux dispone de mecanismos de seguridad que dificultan su propagación (los usuarios no tienen control total del sistema).



Virus y software destructivo: prevención

- Medidas de prevención sencillas:
 - Software específico de búsqueda y destrucción de virus.
 - *Host* víctimas:
 - Se usan ciertos equipos para probar *software* nuevo, asumiendo que puede resultar dañado (*honeypots*).
 - Configuración del entorno: p.ej. la variable `PATH` no incluye la carpeta actual:

```
1 $ echo $PATH
2 /usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
```



Personas malintencionadas

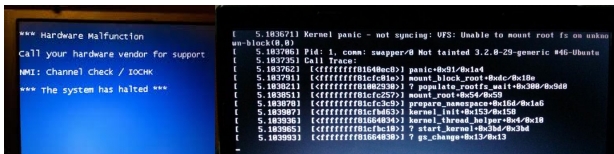


- *Crackers* (\neq *Hackers*):
 - Personas que entran en los sistemas de forma, a veces, ilegal con fines malintencionados.
- Usuarios descontentos:
 - Usuario que tiene acceso al sistema y con recelo hacia la empresa por algún motivo.
- Medidas preventivas:
 - Cortafuegos y Seguridad Física para los *crackers*.
 - Seguimiento de personas sospechosas de ser “usuarios descontentos” controlando sus accesos y sus privilegios.



Fallos de *hardware*

- Fallo en la unidad de disco duro:
 - El kernel suele avisar antes de un fallo completo.
- Fallo de la memoria:
 - Pérdida de información por la caída del sistema o información corrupta en memoria es copiada a disco.
- Prevención y recuperación:
 - Redundancia de la información: utilizar RAID.
 - Supervisión de registros del sistema (orden `dmesg`).
 - Recuperación desde copias de seguridad.
 - Intentar leer bloques para construir una imagen con '`dd`'.
 - Recuperación en entorno estéril: empresa dedicada.



```
*** Hardware Malfunction
Call your hardware vendor for support
HMI: channel check / IOCHK
*** The system has halted ***

[ 5.183671] Kernel panic - not syncing: UFS: Unable to mount root fs on unkno
un-block(0.0)
[ 5.183706] Pid: 1, comm: swapper/0 Not tainted 3.2.0-29-generic #46-Ubuntu
[ 5.183725] Call Trace:
[ 5.183762] [] panic+0x91/0x1a4
[ 5.183791] [] mount_block_root+0x8dc/0x18e
[ 5.183821] [] ? populate_rootfs_wait+0x308/0x9d8
[ 5.183851] [] mount_root+0xc4/0xc9
[ 5.183870] [] prepare_namespace+0x16d/0x1a6
[ 5.183907] [] kernel_init+0x153/0x15b
[ 5.183936] [] kernel_thread_helper+0x4/0x10
[ 5.183965] [] ? start_kernel+0x3bd/0x3bd
[ 5.183993] [] ? __gs_change+0x13/0x13
=
```



Consejos generales

- **Prevención:** Ante cualquiera de los escenarios de pérdida de información debemos tener la capacidad de recuperarnos inmediatamente o en un corto lapso de tiempo.
- Una opción es utilizar **copias de seguridad**.



Consejos generales

Consejos generales:

- Etiquetar siempre las copias realizadas.
- Elegir correctamente la frecuencia de copias.
- Usar particiones distintas para el sistema de ficheros.
- Hacer que el *backup* diario quepa en la unidad.
- Llevarse la copia a otro lugar y proteger ese lugar.
- Limitar la carga computacional durante el proceso de *backup*.
- No esperar a que ocurra un problema para verificar las copias.
- Tener en cuenta el tiempo de vida de los dispositivos.
- Prepararse para lo peor.



Factores I

Factores a considerar en una estrategia de copias de seguridad:

- ¿Qué ficheros se deben copiar y dónde están esos ficheros?.
- Conocer qué es lo más importante del sistema.
- ¿Quién hará la copia?
 - ¿el **administrador** o el **propietario** de los ficheros?.
- ¿Dónde, cuándo y bajo qué condiciones se deben hacer?
 - Mejor hacer las copias cuando no haya usuarios trabajando (por la noche, a la hora de comer...).
- Frecuencia de cambios en los ficheros \Leftrightarrow Frecuencia de las copias.



Factores II

Factores a considerar en una estrategia de copias de seguridad:

- ¿Cada cuánto tiempo habrá que recuperar ficheros dañados o perdidos? (muy difícil saberlo).
- ¿Dónde se restaurarán los datos?.
- Rutinas de restauración sencillas.
- Proteger las copias de seguridad contra escritura.
- Seguridad de las copias:
 - Lugar donde se almacenan, condiciones ambientales, propiedades de los medios empleados...



Estrategias de copias de seguridad I

Copia de seguridad completa

- Se guardan todos los archivos asociados a un ordenador.
- La restauración necesita un solo fichero pero **mucho tiempo**.
- Puede ser difícil recuperar un archivo suelto.
- Si los ficheros no cambian muy a menudo: no hay justificación.
- Si cambian mucho y son vitales para el trabajo de mucha gente: están justificadas incluso a diario.
- Hacerla ante grandes cambios: nuevo *software*, nuevo SO, . . .



Estrategias de copias de seguridad II

Copia de seguridad parcial

- Se copia sólo algunos archivos específicos (por ejemplo, la carpeta/etc).
- Proceso de restauración sencillo, ya que hay menos archivos implicados.
- Problema: nos dejamos archivos sin copiar.



Estrategias de copias de seguridad III

Copia de seguridad incremental

- Solo aquellos ficheros que hayan cambiado desde la última copia.
- Se deben realizar casi a diario.
- Se mantiene una copia completa del sistema, y se incorporan cambios muy pequeños, de los que se irán haciendo copias incrementales.



Estrategias de copias de seguridad IV

- Copias incrementales organizadas por niveles.
 - Nivel 0 → *Backup* completo.
 - Nivel 1 → Todos los ficheros que han cambiado desde el último *backup* de nivel 0.
 - Nivel 2 → Todos los ficheros que han cambiado desde el último *backup* de nivel 1.
 - ...
- Posibilidades de estrategias:
 - Lunes: nivel 0. Resto de días: nivel 1.
 - Lunes: nivel 0. Martes: nivel 1. Miércoles: nivel 2. Jueves: nivel 1. Viernes: nivel 2.
- También hay que asociar una [estrategia de restauración](#).



Soportes para realizar las copias I

- Guardar la copia de seguridad en el mismo disco, o en otro disco conectada a la máquina, no es seguro.
- Multitud de dispositivos:
 - Cintas magnéticas (`/dev/st0`, normal, o `/dev/nst0`, *non-rewinding*, para unidades de cinta SCSI):



Soportes para realizar las copias II

- Discos extraíbles (disco duro que puedes extraer sin apagar la máquina).



- CD-Roms o DVD's regrabables.



- Disquetes.



Soportes para realizar las copias III

- Librería de cintas o *jukeboxes*, *stackloaders* y similares...



- etc.



Criterios para elegir el soporte

- Coste: no solo del dispositivo sino también del soporte físico de almacenamiento.
- Soporte del kernel para el dispositivo.
- Capacidad de almacenamiento de datos de los soportes físicos.
- Tasa de transferencia de datos para realizar copias de seguridad.
- Mecanismo de cargador automático.
 - Cuando se llena una cinta se inserta otra automáticamente.
 - Permite las copias no supervisadas de grandes volúmenes.



Comando tar (Tape ARchiver)

- Realiza copias de seguridad de ficheros o “dispositivos”.
- Algunas opciones son:
 - **c** → Crea un fichero contenedor.
 - **x** → Extrae ficheros de un fichero contenedor.
 - **v** → Modo *verbose* (mayor cantidad de mensajes).
 - **f** → Permite especificar el nombre del fichero contenedor.
 - **z** → Comprime o descomprime mediante *gzip*.
 - **j** → Comprime o descomprime mediante *bz2*.
 - **p** → Conserva los permisos de los ficheros.
 - **P** → Guarda los ficheros con su ruta absoluta.
 - **N** → Considera **solo** archivos cuya fecha sea superior al argumento.



Comando tar (Tape ARchiver)

- `tar cPf /dev/nst0 /home` ⇒ copia todos los ficheros del directorio `/home` en la unidad de cinta.
- `tar czvf /dev/sda1 /home` ⇒ ¿qué sucede con la partición `/dev/sda1`?
- `tar czvf /dev/nst0 /dev/sda1`
- `tar czvf practicas.tgz prac_pas`
- `tar tzvf practicas.tgz` ⇒ listar el contenido de la copia de seguridad realizada en el fichero.
- `tar xzvf practicas.tgz` ⇒ descomprimir.
- `tar xzvf practicas.tgz prac_aso/boletin1.pdf` ⇒ recuperar el fichero `boletin1.pdf` (observa que hay que indicar la ruta con la que tar lo almacenó).



Comando cpio

- Copias de seguridad de conjuntos de ficheros seleccionados arbitrariamente.
 - Empaqueta los datos en una cinta más eficientemente que tar (al restaurar es capaz de saltar trozos de la cinta defectuosos).
 - Lee de la entrada estándar el nombre de los ficheros a guardar, para usarlo enlazado con otras órdenes con tuberías.
 - Algunas opciones:
 - **o** → Copiar “fuera” (*out*) (crear la copia).
 - **i** → Copiar “dentro” (*in*) (descomprimir).
 - **m** → Conserva fecha y hora de los ficheros.
 - **t** → Muestra la tabla de contenidos, es decir, muestra el contenido de la copia.
 - **A** → Añade ficheros a un contenedor existente.
 - **d** → Crear directorios al descomprimir.
 - **v** → Modo *verbose*.
 - **F** → Crear la copia en un fichero.



Comando cpio

- `find /home | cpio -o > /dev/nst0` → se copia en la unidad de cinta.
- `find /home | cpio -o -F h.cpio` → la copia la realiza en un fichero.
- `cpio -i < h.cpio` → restaura la copia de seguridad de ese fichero.
- `cpio -i -F h.cpio fichero` → restaura sólo el fichero indicado.



Comando dump

- Hace copias de seguridad de un sistema de ficheros Ext2, Ext3 o Ext4 de forma **selectiva** (copia una partición completa)
- Permite realizar **copias de seguridad por niveles**: desde el nivel 0, copia completa, al nivel 9 (que es el valor por defecto).
- Actúa solo a nivel de dispositivo.
- `/etc/dumpdates` → información sobre las copias de seguridad de cada SF y de qué nivel son:
`/dev/sda1 0 Mon Feb 14 09:56:44 2013 +0100`
- Algunas opciones son:
 - **0-9** → Nivel de la copia de seguridad, no requiere argumento.
 - **-u** → Actualiza `/etc/dumpdates`, no requiere argumento.
 - **-f** → Indica fichero destino diferente al usado por defecto, sí requiere argumento. Por defecto, se usa la unidad de cinta.



Comando restore

- Restaura copias de seguridad creadas con dump.
- Permite recuperar ficheros, directorios y SF enteros.
- Se ha de recuperar el más reciente de cada nivel empezando por el 0. ¡Mucho cuidado con las fechas!
- Para recuperar SF → crear y montar un SF limpio y vacío, entrar en el punto de montaje y deshacer el *backup*.
- Algunas opciones son:
 - `-r` → Restaura la copia completa, no requiere argumento.
 - `-f` → Indica el dispositivo o archivo donde está el *backup*, sí requiere argumento.
 - `-i` → Modo interactivo, no requiere argumento.
 - `-x` → Extrae los archivos y directorios desde el directorio actual.
 - `-t` → Imprime los nombres de los archivos de la copia, no requiere argumentos.



Ejemplos de dump y restore

- `dump 0 -u -f /dev/nst0 /dev/sda1 →`
Copia de nivel 0 de /dev/sda1 en la unidad de cinta, actualizando /etc/dumpdates.
- `dump 1 -u -f /dev/nst0 /dev/sda1 →`
Copia de nivel 1 de /dev/sda1 en la unidad de cinta, actualizando /etc/dumpdates.
- `dump 0 -f jj.dump /dev/sda1 →`
Copia de nivel 0 de /dev/sda1 en el fichero jj.dump.
- `restore -t -f fichero_backup →` listado de la copia.
- `restore -x -f fichero_backup practicas/smallsh.c →` restaura sólo el fichero practicas/smallsh.c.
- `restore -r -f /dev/nst0 →` restaura una copia completa.
- `restore -i -f /dev/nst0 →` permite restaurar ficheros interactivamente (con ls, cd, pwd, add y extract).



Restauración del sistema

- Si se tiene una copia de todo el sistema:
 - ➊ Arrancar desde un dispositivo distinto (p.e. un DVD).
 - ➋ Si es necesario, crear los ficheros especiales de dispositivos para los discos (/dev/sda1, etc.).
 - ➌ Preparar el disco duro, e.d., crear las particiones.
 - ➍ Crear el sistema de ficheros en la partición donde se restaurarán los datos y montarlo en un directorio.
 - ➎ Restaurar la copia de seguridad sobre ese sistema de ficheros.
 - ➊ Restaurar la copia más reciente de nivel 0.
 - ➋ Restaurar la copia más reciente del nivel más bajo después del último restaurado.
 - ➌ Si quedan más copias por restaurar, volver al paso anterior.
 - ➏ Desmontar el sistema de ficheros restaurado.
 - ➐ Volver al paso 2, para restaurar otros SF adicionales.



Restauración del sistema

- De las siguientes copias realizadas, se restaurarían las que están en rojo:

- 0 0 0 0 0 0 0.
- 0 5 5 5 5 5.
- 0 3 2 5 4 5.
- 0 9 9 5 9 9 3 9 9 5 9 9.
- 0 3 5 9 3 5 9.



Referencias



Nemeth, Snyder y Seebass.

Linux Administration Handbook

Prentice Hall. Segunda Edición. 2007.



Programación y Administración de Sistemas

8. Copias de seguridad y restauración

Pedro Antonio Gutiérrez

Asignatura "Programación y Administración de Sistemas"
2º Curso Grado en Ingeniería Informática
Escuela Politécnica Superior
(Universidad de Córdoba)
pagutierrez@uco.es

19 de abril de 2015

