ADMINISTRACIÓN DE SISTEMAS Y SERVICIOS TELEMÁTICOS

TRABAJO FINAL

LA HERRAMIENTA DUMP

- Backup de Sistemas -

Alumno: JOSÉ JUAN PINA CAMACHO

Cuenta de laboratorio: las99020

E-mail: jjpina@sgt.es

INDICE

NOTA "EXCULPATORIA": En este documento se emplea ampliamente el anglicismo "backup", que en español equivale a la expresión "copia de seguridad", por su extendido uso en nuestro país, al menos en el ámbito de la administración de sistemas.

1. INTRODUCCION: *BACKUP* DE SISTEMAS

Para muchos usuarios, hacer *backups* es una actividad absurda con la que se pierde el tiempo. Lejos de ser real, cualquier administrador de sistemas, y en general cualquiera que emplee una máquina monousuario, debe dedicar un tiempo considerable a pensar seriamente sobre la salvaguarda de sus datos. Existen gran número de errores que pueden producir pérdida o modificación no deseada de los datos: el usuario *root* puede modificar o borrar cualquier fichero del sistema, se pueden producir caídas del sistema con la consecuente pérdida de datos no almacenados en disco, se pueden producir averías en componentes *hardware* como el disco duro, etcétera. Esto es especialmente importante en máquinas claves dentro de una organización como puedan ser servidores, *firewalls*, etcétera.

Un buen *backup* para un sistema debe cumplir una serie de criterios: debe ejecutarse automáticamente sin la intervención del administrador, en la medida que sea posible; los soportes para *backup* deben poder ser gestionados por los propios programas de *backup*, sin necesidad de configuración especial; y la restauración de los datos guardados debe ser fácil y eficiente.

Con la estructura de directorios de Linux y sus potentes herramientas es relativamente fácil realizar esta labor para los ficheros de configuración y de usuario.

En un *backup* es fundamental comprobar que el soporte físico empleado para el almacenamiento de los datos no tiene ningún error, e incluso es adecuado realizar una comprobación rutinaria de la restauración de los datos para comprobar que en un hipotético caso de destrucción de datos, ésta se llevaría a cabo sin problemas.

Para la realización de *backups*, existen programas muy diversos que van desde simples herramientas incorporadas al sistema que leen del disco duro y almacenan los datos en el medio adecuado (programas de bajo nivel), hasta potentes herramientas que permiten una mayor administración y configuración de las técnicas de almacenamiento (programas de alto nivel, p. Ej. Amanda).

En este documento se presenta una herramienta entre bajo y alto nivel, *dump*, de muy fácil configuración y ampliamente usada en cualquier sistema Unix. También se presenta la herramienta *restore*, que es la encargada de restaurar los datos almacenados con *dump*.

2. OBJETIVOS Y DESCRIPCIÓN DE LA HERRAMIENTA

Dump examina los ficheros de un sistema de ficheros y determina cuáles de ellos precisan una copia de seguridad. Estos ficheros son copiados al medio físico que se emplee (disco, cinta o cualquier otro) para un mantenimiento seguro.

Es una herramienta muy difundida entre los sistemas Unix y, aunque accede a la estructura interna de ficheros del sistema, es necesaria una versión específica de *dump* para cada tipo de sistema. Para Linux sólo es posible su empleo con sistemas de ficheros tipo *ext2* y *Minix*, si bien el primero es el propio de toda partición Linux.

La herramienta *dump* es de bajo nivel en cuanto es empleada por programas de mayor nivel como por ejemplo *amanda*. Pero al mismo tiempo es de alto nivel en cuanto almacena la fecha y nivel del último *backup* en el fichero */etc/dumpdates*, de forma que pueda ser empleado para la gestión de diferentes niveles de *backup* (técnicas incrementales).

La restauración de los datos almacenados con *dump* se realiza mediante la herramienta *restore*.

El programa *dump* depende de que el formato de */etc/fstab* sea adecuado, en el sentido en que no contenga la entrada *defaults* entre sus opciones.

Como ejemplo de su uso, un *backup* completo del directorio *home* a una cinta flexible se puede obtener sin más que ejecutar el siguiente comando:

(linux):~> ***dump 0Bbuf 120000 1000 /dev/rft0 /home***

El parámetro *0* (nivel de *dump*) indica copia completa del sistema de ficheros. Los diferentes niveles pueden ser empleados para *backups* incrementales o diferenciales. La opción *u* se emplea para la actualización del fichero */etc/dumpdates*, que mantiene un histórico de todas las ejecuciones de *dump* exitosas y, como ya se ha comentado, sirve además para realizar *backups* incrementales. Las opciones *B* y *b* se incluyen para a continuación pasar como parámetros el tamaño y número de bloques.

3. REQUISITOS

En este apartado se describen los requerimientos de la herramienta para poder ser utilizada. Concretamente nos centramos en los soportes sobre los que puede ser realizado el *backup*.

Con el tamaño actual de los discos duros, es inviable pensar en un *backup* completo en *diskettes*. El gasto en tiempo y soporte físico excedería con creces las necesidades de seguridad a resolver. No obstante, para pequeñas cantidades de datos que requieran *backups* frecuentes la opción de los *diskettes* sí tiene sentido.

De cara a un *backup* automático y conveniente es necesario el empleo de un medio que haga innecesaria la presencia del administrador. En ese sentido son muy útiles las cintas automáticas que cambian solas de una a otra cuando se llenan. De esta forma es posible la realización de *backups* durante la noche que no interfieran con la actividad humana diurna.

El soporte más utilizado para la realización de *backups* es la cinta magnética. Son muy baratas y, a pesar de que el acceso es secuencial, suficientemente rápidas. Hay diferentes estándares para cintas magnéticas y los correspondientes dispositivos o interfaces a los que pueden ser conectadas. De entre estos estándares, el más conocido es *QIC (Quarter Inch Cartridge).*

4. ARQUITECTURA

Tanto bajo Linux como bajo cualquier otro sistema Unix, estas técnicas de *backup* pueden almacenar los datos en dispositivos conectados a otra computadora y que pueden ser empleados a través de la red. Así, para una red con gran número de máquinas puede ser muy útil tener un servidor de *backup* que, conectado a potentes máquinas de trabajo con cintas y otros dispositivos, permita hacer copias de seguridad de cualquier máquina de la red.

Esto hace posible centralizar las copias de seguridad de la red, de forma que la gestión del soporte físico de almacenamiento deba ser realizada en un único punto de la red y no en cada máquina. E incluso se pueden realizar copias de seguridad globales, seguras y bien estructuradas, como parte de la política de gestión de la red.

Si bien es la arquitectura más empleada en cualquier red de ordenadores, tiene algunas desventajas propias de este tipo de servicios, como son la coordinación y configuración del servicio, además del aumento de carga de datos en la red. Además, un *backup* puede tardar a través de la red un tiempo significativamente mayor que si se realiza mediante un potente dispositivo local; sin embargo, cuando se trata de realizar copias de seguridad para un gran número de máquinas en red, la arquitectura comentada es la más adecuada.

5. CONFIGURACIÓN

Normalmente se suelen automatizar las tareas de *backup* mediante *scripts* sencillos que se ejecutan de forma automática.

Básicamente existen dos tipos de *backup*: completo cuando queremos almacenar todos los ficheros e incremental cuando queremos almacenar sólo aquellos que hayan cambiado desde la última vez.

Para realizar un *backup* mediante *dump* debe usarse la orden:

```
dump [0123456789ackMSu] [B records] [b blocksize] [d density]
      [e inode number] [f file] [F script] [h level] [L label]
      [s feet] [T date] filesystem/directory
```

- 0123456789: indica el nivel de *backup*: 0 completo; mayor que 0 incremental, salvándose sólo aquellos ficheros modificados respecto a *dump* anterior de nivel inferior
- a: fuerza a escribir hasta el final del soporte
- c: usar cinta "cartridge"
- k: usar autenticación *kerberos* para uso con servidor remoto
- M: para múltiples volúmenes (usando prefijo dado por *f*)
- S: estima, sin ejecutarlo, cuánto espacio es necesario para hacer el *backup*
- u: actualiza el fichero de monitorización */etc/dumpdates*
- B: indica el número de bloques (*records*) del dispositivo
- b: indica el tamaño del bloque (*blocksize*) del dispositivo
- d: determina la densidad (density) del dispositivo
- e: excluye el inodo *inode* del *backup*
- f: hace el *backup* en fichero (tiene que ser un fichero asociado a un dispositivo)
- F: indica el *script* a usar para confirmar cambios de cinta
- h: configura el nivel a partir del que puede hacerse *backup*
- L: etiqueta a la que luego puede acceder *restore*
- s: estima la cantidad de cinta necesaria para una densidad particular (*feet*), solicitando otra si se queda corta
- T: establecen momento a realizar *backup* en lugar de los dados en */etc/dumpdates*

Un aspecto muy interesante es el de la configuración remota, que permite guardar el *backup* en un fichero de otra máquina. Para ello se debe invocar con la opción *f* y el parámetro *host:file* o usr@host:file. De esta forma se guarda en el directorio */etc/rmt* de dicha máquina. Esta configuración sólo es posible emplearla trabajando como *root*, pues tiene riesgos de seguridad asociados.

6. PRUEBAS

En toda copia de seguridad, la mejor forma de probar que se ha completado con éxito es realizar una restauración inmediata de los datos empleados en algún lugar auxiliar para comprobar que, en caso de un fallo real, se restaurarían los datos sin mayores problemas.

Así pues, la mejor forma de probar los resultados obtenidos con la herramienta *dump* es empleando su homólogo para la restauración de los datos, *restore*, que se describe en el último apartado de este documento.

7. GESTIÓN DIARIA

En este apartado se describe un poco el día a día del administrador en lo que a *backups* se refiere, en el sentido de la estrategia que debe seguir para la realización de sus copias de seguridad.

Las estrategias de *backup* dependen de muchos factores. El administrador del sistema debe decidir qué datos requieren ser salvados, con qué intervalos temporales, así como qué tipo y número de dispositivos físicos deben ser empleados. La secuencia de *backups* completos e incrementales debe ser especificada. En *backups* completos, todos los ficheros salvo escasas excepciones son almacenados, mientras las técnicas incrementales sólo afectan a aquellos ficheros que han sido modificados desde la última copia de seguridad.

A la hora de realizar *backup*, deben seguirse los pasos siguientes:

- Las partes exactas o subdirectorios afectados deben ser minuciosamente determinados por el administrador del sistema, en función de los requerimientos existentes.

- Se debe intentar llevar a cabo en tiempos de tráfico de datos bajo, pues por un lado menor número de datos están sometidos a un posible cambio, y por otro lado la ejecución requiere ciertos recursos del sistema por lo que debe tratar de interferirse lo mínimo posible.

- Además debe tratar de realizarse el *backup* en modo monousuario, aunque esto sea muy difícil en la mayoría de los casos, pues normalmente se trata de servidores dentro de una determinada red.

A la hora de decidir qué ficheros deben ser salvados, normalmente se siguen los siguientes criterios:

- Los datos de usuario del directorio */home* suelen cambiar muy a menudo, por lo que un *backup* frecuente del mismo es adecuado.

- Los ficheros de configuración en */etc* y */var* son especialmente delicados en cuanto suelen llevar detrás una minuciosa labor de estudio y conocimiento, por lo que también es importante realizar copias de seguridad cada vez que se cambian.

- Finalmente, las particiones */usr* y */usr/local* normalmente contienen aplicaciones bastante estáticas, pero no está de más realizar copias de seguridad cada cierto tiempo.

Finalmente otro aspecto importante es el número de medios empleados para el *backup*. Normalmente con dos es suficiente, de forma que siempre se almacena en el más antiguo, pero en ocasiones es conveniente el uso de un tercero para mayor seguridad.

8. LA HERRAMIENTA *RESTORE*

El comando *restore* realiza la función inversa a *dump*: un *backup* completo realizado mediante la herramienta *dump* puede ser restaurado y los subsiguientes *backups* incrementales montados sobre él. Asimismo, se pueden restaurar ficheros o subdirectorios individuales a partir de *backups* completos o parciales.

Para el ejemplo que veíamos al describir la herramienta *dump*, la instrucción para restaurar el *backup* de la cinta sería:

(linux):~> restore rf ***/dev/rft0***

La opción *r* indica restauración automática sin necesidad de intervención del administrador. Para una restauración interactiva se emplearía la opción *i*, que desplegaría una *shell* simple en la que se puede navegar mediante *cd* y *ls*; para ficheros o directorios individuales se emplearía *add* (pudiendo ser borrados posteriormente mediante *delete*); para hacer efectiva la restauración de los elementos seleccionados se emplearía *extract*.

Restore se basa en el fichero */etc/dumpdates* para restauración de *backups* incrementales.

9. REFERENCIAS

- "Linux Copanion for System Administrators" (Cap. 7)
  Jochen Hein. Ed. Addison-Wesley. 1999

- Páginas "*man*" de Linux: dump(8) y restore(8)

- http://dump.sourceforge.net

- http://www.debian.org