

## **LPIC-2 / Examen 206 - Mantenimiento del Sistema**

Este examen (en la nomenclatura oficial de LPI, estos objetivos están bajo el Examen 202) cubre la realización de copias de seguridad y la notificación a los usuarios.

### **206.1 Realizar copias de seguridad**

#### **Teoría**

Las copias de seguridad (backups) son copias de tus datos que se almacenan en un lugar seguro para poder recuperarlos en caso de pérdida, daño o corrupción de los datos originales. La pérdida de datos puede ocurrir por fallos de hardware, errores humanos (borrado accidental), ataques de malware (ransomware), desastres naturales, etc.

#### **Estrategia Básica de Backup:**

Una estrategia de backup efectiva responde a:

- **¿Qué respaldar?** Archivos de configuración del sistema (`/etc`), datos de usuarios (`/home`), datos de aplicaciones (bases de datos, archivos de sitios web), archivos importantes del sistema (`/var`, `/opt`).
- **¿Dónde almacenar los backups?** En un medio de almacenamiento diferente al original (otro disco interno, disco externo, almacenamiento en red, cinta, nube). La regla 3-2-1 (3 copias de datos, en 2 tipos de medios diferentes, con 1 copia fuera del sitio) es una buena práctica.
- **¿Con qué frecuencia?** Depende de la criticidad de los datos y la frecuencia con la que cambian (diaria, semanal, mensual, continua).
- **¿Cómo verificar los backups?** Regularmente, intenta restaurar algunos archivos o directorios desde el backup para asegurarte de que los datos se pueden recuperar correctamente.
- **¿Cuánto tiempo conservar los backups?** Define una política de retención.

#### **Tipos de Copias de Seguridad:**

- **Backup Completo (Full Backup):** Copia todos los datos seleccionados. Es la base de cualquier estrategia. Es el más lento de crear y el que ocupa más espacio, pero el más rápido de restaurar.
- **Backup Incremental:** Copia solo los datos que han cambiado desde el *último backup de cualquier tipo* (completo o incremental). Es rápido de crear y ocupa menos espacio, pero la restauración puede ser lenta, ya que requiere restaurar el último backup completo y luego todos los incrementales subsiguientes en orden.
- **Backup Diferencial:** Copia solo los datos que han cambiado desde el *último backup completo*. Es más rápido de crear que uno completo y la restauración es más rápida que con incrementales (solo necesitas el último completo y el último diferencial). Ocupa más espacio que un incremental pero menos que un completo.

#### **Herramientas de Línea de Comandos para Backups:**

## 1. **tar (Tape Archiver):**

- Herramienta versátil para crear archivos (paquetes de archivos y directorios) y extraerlos. A menudo se combina con compresión. No es inherentemente una herramienta de backup incremental/diferencial a menos que se combine con `find` u otras lógicas.
- **Opciones Comunes:**
  - `-C`: Crear un archivo.
  - `-x`: Extraer de un archivo.
  - `-t`: Listar contenido de un archivo.
  - `-f <archivo>`: Especificar el nombre del archivo de archivo (o dispositivo).
  - `-v`: Mostrar verbosidad (lista los archivos procesados).
  - `-P`: No eliminar las barras / al principio de las rutas (preserva rutas absolutas).
  - `--exclude=<patron>`: Excluir archivos o directorios que coincidan con el patrón.
- **Compresión:**
  - `-Z` (o `--gzip`): Comprimir/descomprimir con gzip (`.tar.gz` o `.tgz`).
  - `-j` (o `--bzip2`): Comprimir/descomprimir con bzip2 (`.tar.bz2` o `.tbz`).
  - `-J` (o `--xz`): Comprimir/descomprimir con xz (`.tar.xz` o `.txz`).
- **Backup a archivo/dispositivo:** `tar -cvf /ruta/a/backup.tar /ruta/a/respaldar`. Para comprimir: `tar -czvf /ruta/a/backup.tar.gz /ruta/a/respaldar`. Para respaldar directamente a otro disco (¡cuidado!): `tar -cvf /dev/sdb1 /ruta/a/respaldar`.
- **Restaurar:** `tar -xvf /ruta/a/backup.tar [-C /ruta/destino]` (extrae en el directorio actual o en el especificado por `-C`).

## 2. **dd (dataset dump):**

- Herramienta de bajo nivel para copiar datos a nivel de bloque. Ideal para crear imágenes exactas de discos o particiones enteras. No es eficiente para copias de seguridad a nivel de archivo o copias incrementales. Copia todos los bloques, incluyendo el espacio vacío.
- **Opciones Comunes:**
  - `if=<origen>`: Archivo o dispositivo de entrada (input file).
  - `of=<destino>`: Archivo o dispositivo de salida (output file).
  - `bs=<tamaño>`: Tamaño de los bloques a leer/escribir.
  - `count=<número>`: Número de bloques a copiar.
- **Backup de partición a archivo:** `sudo dd if=/dev/sda1 of=/ruta/a/backup_sda1.img bs=4M`.

- **Restaurar partición desde archivo (¡PELIGROSO!):** `sudo dd if=/ruta/a/backup_sda1.img of=/dev/sda1 bs=4M`. El dispositivo de destino debe ser del mismo tamaño o mayor.

### 3. **rsync (remote sync):**

- Herramienta potente y flexible para sincronizar archivos y directorios, local o remotamente. Es muy eficiente para copias de seguridad incrementales, ya que solo transfiere las partes de los archivos que han cambiado utilizando un algoritmo de diferencias binarias.
- **Sintaxis:** `rsync [opciones] <origen> <destino>`
  - **<origen>:** Archivo o directorio local o remoto (`user@host:/ruta`).
  - **<destino>:** Archivo o directorio local o remoto (`/ruta, user@host:/ruta`).
- **Opciones Comunes:**
  - **-a (archive mode):** Modo archivo, un atajo para varias opciones (`-r lptgoD`) que preserva permisos, tiempos, propietario, grupo, enlaces simbólicos, dispositivos, recursivamente.
  - **-v:** Verbose, muestra qué archivos se transfieren.
  - **-h:** Output legible por humanos para tamaños.
  - **--delete:** Elimina archivos en el destino que no existen en el origen (sincronización exacta). **¡Usar con cuidado!**
  - **--exclude=<patron>:** Excluir archivos o directorios.
  - **--include=<patron>:** Incluir archivos o directorios (útil con `--exclude` para definir un conjunto específico).
  - **-z:** Comprimir datos durante la transferencia (útil para backups remotos).
  - **-P:** Mostrar progreso durante la transferencia.
- **Backup incremental local:** `rsync -avh --delete /home/mi_usuario/documentos/ /ruta/a/backup/documentos/`. La primera ejecución copia todo; las subsiguientes solo copian los cambios y eliminan archivos borrados.
- **Backup incremental remoto (vía SSH):** `rsync -avzP --delete /home/mi_usuario/documentos/ usuario_remoto@servidor_backup:/ruta/a/backup/documentos/`.

### 4. **dump / restore:**

- Herramientas tradicionales a nivel de sistema de archivos, principalmente para ext2/3/4. Permiten realizar copias de seguridad a nivel de sistema de archivos (no de bloque como `dd`).
- **dump:** Crea backups de un sistema de archivos. Soporta niveles de backup (0 para completo, 1-9 para incrementales). Lee el campo `dump` en `/etc/fstab`.
- **restore:** Restaura backups creados con `dump`.

- Son menos comunes hoy en día en comparación con `tar` o `rsync` para backups generales de archivos/directorios, pero LPIC-2 puede mencionarlos.

### Programación de Backups:

Las copias de seguridad manuales no son fiables. Deben programarse para que se ejecuten automáticamente.

- **cron:** El demonio de programación clásico. Puedes usar `crontab -e` para el usuario, o archivos en `/etc/cron.d/`, `/etc/cron.hourly/`, `/etc/cron.daily/`, etc. (revisitado en 108.3). Coloca tus scripts de backup (que usan `tar`, `rsync`, etc.) en estas ubicaciones o configúralos en `crontab`. **La ubicación exacta de los directorios `cron.*` y su gestión puede variar ligeramente entre distribuciones.**
- **Systemd Timers:** El método moderno basado en `systemd` para programar tareas. Proporciona más flexibilidad y mejor integración con `systemd` que `cron`. Se crean archivos de unidad `.timer` y `.service`. **Este método es estándar en ambas ramas con `systemd`.**

### Verificación y Restauración:

- **Siempre prueba a restaurar datos.** Un backup que no se puede restaurar no sirve para nada.
- Verifica la integridad de los archivos de backup (ej: sumas de verificación) si es posible.
- Mantén los registros (logs) de las ejecuciones de los backups para verificar si tuvieron éxito o si hubo errores.

### Diferencias Debian vs. Red Hat (Backups):

- Las herramientas principales (`tar`, `dd`, `rsync`, `dump/restore`) son las mismas y se instalan desde paquetes con el mismo nombre o similares.
- La forma de programar backups es la misma (`cron` o `systemd` timers). La diferencia puede estar en la organización de los directorios `cron.*` por defecto.
- La gestión de los campos `dump/pass` en `/etc/fstab` es estándar, aunque el uso de `dump/restore` en sí mismo es menos común hoy en día.