



Examen 102 - Instalación de Linux y Gestión de Paquetes

102.4 Gestión de paquetes Debian

Teoría

Las distribuciones basadas en Debian (como la propia Debian, Ubuntu, Mint, etc.) utilizan un sistema de gestión de paquetes basado en archivos con la extensión `.deb`. Este sistema simplifica enormemente la instalación, actualización, configuración y eliminación de software.

Conceptos Fundamentales:

1. **Paquete (`.deb`):** Es un archivo de archivo (`ar`) que contiene:
 - Los archivos del programa a instalar (ejecutables, bibliotecas, archivos de configuración, documentación, etc.).
 - Metadatos sobre el paquete: nombre, versión, descripción, dependencias (otros paquetes que necesita para funcionar), conflictos (paquetes con los que no puede coexistir), scripts pre/post-instalación/eliminación, etc.
2. **dpkg:** Es la herramienta de bajo nivel para gestionar paquetes `.deb` individuales. `dpkg` trabaja directamente con los archivos `.deb` o con la base de datos local de paquetes instalados (`/var/lib/dpkg/`).
 - `dpkg -i <archivo.deb>`: Instala un paquete desde un archivo `.deb`. No resuelve dependencias automáticamente.
 - `dpkg -l [<patrón>]`: Lista los paquetes instalados. Puedes usar un patrón para filtrar (ej: `dpkg -l "*apache2*"`).
 - `dpkg -s <nombre_paquete>`: Muestra información detallada sobre un paquete instalado.
 - `dpkg -L <nombre_paquete>`: Lista los archivos instalados por un paquete.
 - `dpkg -r <nombre_paquete>`: Elimina un paquete, pero conserva sus archivos de configuración.
 - `dpkg -P <nombre_paquete>`: Elimina un paquete y sus archivos de configuración ("purge").
 - `dpkg -S <ruta_archivo>`: Busca qué paquete instaló un archivo determinado.
3. **APT (Advanced Packaging Tool):** Es la interfaz de alto nivel recomendada para la gestión de paquetes. APT no trabaja directamente con archivos `.deb` en la mayoría de los casos (aunque puede), sino que interactúa con **repositorios** de paquetes. APT se encarga de tareas complejas como la resolución de dependencias, la gestión de actualizaciones del sistema completo y la búsqueda de paquetes. Las herramientas más modernas dentro del ecosistema APT son `apt` y `apt-get`. `apt` está diseñado para ser más amigable para el usuario final, combinando funcionalidades comunes de `apt-get`, `apt-cache`, etc.

- `apt update`: Descarga la información más reciente sobre los paquetes disponibles desde los repositorios configurados. **Es crucial ejecutar esto antes de instalar o actualizar software.**
 - `apt upgrade`: Actualiza todos los paquetes instalados a sus versiones más recientes, respetando las dependencias. No elimina paquetes ni instala nuevos a menos que sea estrictamente necesario por dependencias.
 - `apt full-upgrade`: Similar a `upgrade`, pero puede eliminar paquetes viejos o instalar nuevos para manejar cambios complejos de dependencias.
 - `apt install <nombre_paquete>`: Instala un paquete (y todas sus dependencias requeridas) desde los repositorios.
 - `apt remove <nombre_paquete>`: Elimina un paquete, pero conserva sus archivos de configuración.
 - `apt purge <nombre_paquete>`: Elimina un paquete y sus archivos de configuración.
 - `apt search <término>`: Busca paquetes que coincidan con el término en sus nombres o descripciones.
 - `apt show <nombre_paquete>`: Muestra información detallada sobre un paquete disponible en los repositorios (no necesariamente instalado).
 - `apt autoremove`: Elimina paquetes que fueron instalados automáticamente para satisfacer dependencias, pero que ya no son necesarios para ningún paquete instalado.
4. **Repositorios:** Son servidores (HTTP, FTP) o directorios locales que almacenan colecciones de paquetes `.deb` organizados y firmados digitalmente.
- La lista de repositorios que APT utiliza se define en `/etc/apt/sources.list` y en archivos `.list` individuales dentro del directorio `/etc/apt/sources.list.d/`.
 - Cada línea en estos archivos especifica el tipo de repositorio (`deb` o `deb-src`), la URL del repositorio, la distribución o "suite" (ej: `buster`, `focal`, `jammy`), y las "secciones" o componentes (ej: `main`, `restricted`, `universe`, `multiverse` en Ubuntu; `main`, `contrib`, `non-free` en Debian).
5. **Gestión de Dependencias:** El sistema de paquetes Debian es muy robusto en cuanto a la gestión de dependencias. Cuando intentas instalar un paquete, APT verifica qué otros paquetes necesita y los instala automáticamente. Si intentas eliminar un paquete del que dependen otros, APT te advertirá o te impedirá hacerlo a menos que también elimines los paquetes dependientes.

Flujo de Trabajo Típico:

1. `sudo apt update`: Sincronizar la lista de paquetes disponibles con los repositorios.
2. `apt search <software>`: Buscar el nombre del paquete deseado.
3. `apt show <nombre_paquete>`: Obtener más detalles sobre el paquete.
4. `sudo apt install <nombre_paquete>`: Instalar el paquete.

5. (Opcional) `sudo apt remove <nombre_paquete>` o `sudo apt purge <nombre_paquete>`: Eliminar el paquete.
6. `sudo apt autoremove`: Limpiar dependencias no utilizadas.
7. `sudo apt upgrade`: Mantener el sistema actualizado.