🔐 Examen 110 - Seguridad

Este examen cubre la realización de tareas administrativas de seguridad, la configuración de seguridad del host y el aseguramiento de datos con cifrado básico.

110.1 Realizar tareas de administración de seguridad

Teoría

Las tareas de administración de seguridad en Linux buscan proteger la **confidencialidad** (evitar acceso no autorizado a datos), la **integridad** (proteger datos de modificaciones no autorizadas) y la **disponibilidad** (asegurar que los recursos están accesibles cuando se necesitan) del sistema y los datos que aloja. Este objetivo se centra en las prácticas administrativas básicas para mantener la seguridad del host.

1. **Principios de Seguridad Básicos:**
    - **Actualizaciones Regulares:** Mantener el sistema operativo y las aplicaciones actualizadas es fundamental para corregir vulnerabilidades de seguridad conocidas. Utiliza el gestor de paquetes (`apt`, `dnf`/`yum`).
    - **Contraseñas Fuertes:** Obligar o alentar el uso de contraseñas largas, complejas y únicas. Configurar políticas de antigüedad de contraseñas (`chage`, `/etc/login.defs`).
    - **Principio de Menor Privilegio:** Otorgar a usuarios y servicios solo los permisos estrictamente necesarios para realizar sus funciones. No operar como root a menos que sea indispensable.
    - **Auditoría y Monitorización:** Revisar regularmente los logs del sistema (108.2) para detectar actividades sospechosas.
    - **Firewalls:** Controlar el tráfico de red entrante y saliente (110.2).
    - **Backups:** Realizar copias de seguridad regulares y verificarlas para recuperarse en caso de pérdida de datos o corrupción por ataques (fuera del alcance detallado de LPIC-1, pero es una práctica de seguridad clave).
    - **Eliminar Servicios Innecesarios:** Deshabilitar o desinstalar servicios que no se utilizan para reducir la superficie de ataque.
2. **Control de Acceso Basado en Usuarios y Grupos:** (Revisitado de 107.1 y 104.5)
    - La correcta gestión de usuarios, grupos y permisos de archivos (`chmod`, `chown`, `chgrp`) es la primera línea de defensa para controlar quién puede acceder a qué en el sistema local.
    - El usuario `root` (UID 0) tiene bypass a la mayoría de los permisos. Limitar el uso directo de la cuenta root es vital.
3. **Gestión Segura de Privilegios Elevados (`sudo` vs `su`):**

- **`su`:** Cambia al usuario root (o a otro usuario) solicitando la contraseña del *usuario destino*. Si muchos administradores conocen la contraseña de root, es difícil auditar quién hizo qué.
- **`sudo`:** Permite a usuarios autorizados ejecutar comandos como root (o como otro usuario) utilizando *su propia contraseña*. La configuración en `/etc/sudoers` permite especificar granularmente qué usuarios o grupos pueden ejecutar qué comandos, desde qué terminales, y como qué usuarios. Es el método preferido para otorgar privilegios limitados y proporciona un registro de auditoría (en los logs del sistema) de quién ejecutó qué comando con `sudo`.
- **Archivo `/etc/sudoers`:** Contiene las reglas de `sudo`. **¡Siempre debe editarse usando el comando `visudo`!** `visudo` verifica la sintaxis antes de guardar, previniendo errores que podrían bloquear el acceso a root. El archivo tiene un formato específico para definir alias de usuarios, hosts, comandos y reglas que vinculan quién puede hacer qué, dónde y como quién.
- **Grupos para acceso sudo:** Es común otorgar acceso sudo completo o parcial a miembros de un grupo específico en lugar de a usuarios individuales.
    - **Diferencias Debian vs. Red Hat (Grupo sudo por defecto):**
        - **Rama Debian/Ubuntu:** El grupo común con acceso sudo es `sudo`. Los usuarios se añaden a este grupo para permitirles usar `sudo`.
        - **Rama Red Hat/CentOS/Fedora:** El grupo tradicional con acceso sudo es `wheel`. La configuración por defecto en `/etc/sudoers` a menudo ya permite que los miembros del grupo `wheel` ejecuten cualquier comando como root. Los usuarios se añaden a este grupo para darles acceso sudo.

4. **Monitorización de Actividad de Login:**
    - **`/var/log/auth.log` (Debian) / `/var/log/secure` (Red Hat):** Contienen registros de intentos de inicio de sesión (exitosos y fallidos), uso de `sudo`, `su`, etc. Esencial para detectar actividad sospechosa.
    - **`last`:** Muestra un historial de los últimos inicios de sesión de usuarios en el sistema (leyendo `/var/log/wtmp`). Útil para ver quién se ha conectado y cuándo.
    - **`who`:** Muestra quién está actualmente logueado en el sistema.
    - **`w`:** Similar a `who`, pero también muestra qué comando está ejecutando cada usuario logueado.
5. **Verificación de Integridad de Archivos/Paquetes:**
    - **Sums de Verificación:** Utilizar herramientas como `md5sum`, `sha256sum` para generar y verificar sumas de hash de archivos. Si la suma de un archivo cambia inesperadamente, podría indicar una modificación no autorizada.
    - **Verificación de Paquetes:** Los gestores de paquetes (`apt`, `rpm`, `dpkg`, `dnf`) tienen funcionalidades para verificar la integridad de los archivos instalados como parte de un paquete.

- `dpkg --verify <paquete>` o `dpkg -V <paquete>` (Debian).
- `rpm -V <paquete>` (Red Hat).