Gestión y Administración de Usuarios en Linux

20 de febrero de 2025

Objetivos de la Presentación

- Comprender los conceptos fundamentales sobre usuarios y pseudo-usuarios.
- Revisar los ficheros de configuración clave: `/etc/passwd`, `/etc/shadow`, `/etc/group` y `/etc/gshadow`.
- Conocer el proceso de creación, modificación y eliminación de cuentas de usuario.
- Analizar la gestión de grupos y la asignación de permisos.
- Explorar aspectos de seguridad, autenticación y auditoría en la administración de usuarios locales.

Introducción a la Gestión de Usuarios

- Un **usuario** es una entidad (persona o proceso) que interactúa con el sistema.
- Un **pseudo-usuario** es una cuenta utilizada para ejecutar programas o servicios.
- Características de un usuario:
  - **Nombre de usuario** (username)
  - **UID** (Identificador numérico único)
  - **Grupos** a los que pertenece (GID principal y secundarios)

Ficheros de Configuración de Usuarios

- **/etc/passwd**: Lista las cuentas de usuario.
  - Formato: `nombre:password:UID:GID:gecos:home:shell`
  - Ejemplo: `juan:x:1001:1001:Juan Pérez:/home/juan:/bin/bash`
- **/etc/shadow**: Contiene las contraseñas encriptadas y políticas de expiración.
- **/etc/group**: Define los grupos del sistema y sus miembros.
- **/etc/gshadow**: Información de seguridad de los grupos.
- **/etc/login.defs** y **/etc/skel**: Valores por defecto y ficheros de inicialización para nuevos usuarios.

Creación y Gestión de Usuarios

- La herramienta `adduser` (o `useradd`) crea una nueva cuenta de usuario.
- Se asigna el primer UID libre (generalmente a partir de 500 o 1000) y se crea el directorio `HOME`.
- Se copian los ficheros de `/etc/skel` al directorio `HOME` del usuario.

Ejemplo:

```
adduser juan
passwd juan
```

Gestión de Grupos y Permisos

- Los grupos permiten asignar permisos de forma colectiva.
- El fichero `/etc/group` almacena la información de cada grupo.
- Un usuario puede pertenecer a un grupo primario (definido en `/etc/passwd`) y a grupos secundarios.

Ejemplo: Añadir el usuario `juan` al grupo `desarrolladores`

```
groupadd desarrolladores
usermod -aG desarrolladores juan
```

Configuración del Shell y Autenticación

- El último campo en `/etc/passwd` especifica el shell por defecto.
- Los shells permitidos están listados en `/etc/shells`.
- Se puede cambiar el shell con el comando `chsh`.

Ejemplo: Cambiar el shell del usuario `juan` a `/bin/zsh`

```
chsh -s /bin/zsh juan
```

Resumen del Proceso de Creación de un Usuario

- Al ejecutar:

  `useradd juan`

  se realizan los siguientes pasos:

  1. Se añade una línea en `/etc/passwd` similar a:

     `juan:x:1001:1001:Juan Pérez:/home/juan:/bin/bash`

  2. Se añade una línea en `/etc/shadow` para gestionar la contraseña y políticas de expiración.
  3. Se crea un grupo primario para `juan` en `/etc/group`.
  4. Se añade una entrada en `/etc/gshadow` para la seguridad del grupo.
  5. Se crea el directorio `/home/juan` y se copian los ficheros de `/etc/skel`.

Configuración de sudoers y Privilegios

- El archivo `/etc/sudoers` define los permisos para ejecutar comandos con privilegios elevados.
- Se recomienda editarlo mediante el comando `visudo` para evitar errores de sintaxis.
- **Ejemplo:** Permitir al usuario `juan` ejecutar todos los comandos sin pedir contraseña:

```
juan ALL=(ALL) NOPASSWD: ALL
```

- También se pueden asignar permisos a grupos completos.

Uso de PAM (Pluggable Authentication Modules)

- PAM es un marco modular que permite gestionar la autenticación y otros mecanismos de seguridad de forma flexible.
- Los módulos PAM son bibliotecas compartidas que se cargan en tiempo de ejecución y se aplican a distintos servicios.
- La configuración de PAM se organiza en cuatro categorías:
  - **auth:** Verifica la identidad del usuario (por ejemplo, solicitando una contraseña).
  - **account:** Controla restricciones de acceso, como horarios o vencimiento de cuentas.
  - **password:** Gestiona el cambio y la complejidad de contraseñas.
  - **session:** Configura el entorno del usuario tras iniciar sesión (por ejemplo, variables de entorno o registro de sesiones).
- Cada servicio (SSH, login, su, etc.) tiene su propio archivo de configuración en `/etc/pam.d/`.

Uso de PAM (Pluggable Authentication Modules)

- Los parámetros de control (`required`, `requisite`, `sufficient`, `optional`) determinan la criticidad y el comportamiento en la pila de autenticación.
- **Ejemplo:** Fragmento de configuración en `/etc/pam.d/login`, mostrando cómo se integran distintos módulos para controlar la autenticación, las restricciones de cuenta, la gestión de contraseñas y la configuración de sesión.

```
#%PAM-1.0
auth        required     pam_securetty.so
auth        include      system-auth
account     required     pam_nologin.so
account     include      system-auth
password    include      system-auth
session     required     pam_env.so
session     include      system-auth
session     optional     pam_mail.so standard noenv
session     include      system-auth
```

Seguridad en la Gestión de Usuarios

- Pueden implementarse políticas de contraseñas seguras y caducidad.
- Se utiliza `chage` para gestionar los periodos de expiración de las contraseñas.
- **Ejemplo:** Configurar al usuario `maria` para que cambie su contraseña cada 90 días, con un aviso 14 días antes:

```
chage -M 90 -m 7 -W 14 maria
```

- Además, es recomendable bloquear cuentas inactivas para reducir riesgos.

Auditoría y Seguimiento en la Gestión de Usuarios

- Se pueden monitorizar los accesos y cambios en las cuentas de usuario.
- Herramientas útiles:
  - `lastlog`: Muestra el último acceso de cada usuario.
  - `faillog`: Registra intentos fallidos de autenticación.
  - `auditd`: Sistema de auditoría que registra eventos críticos.
- **Ejemplo:** Consultar el último acceso de los usuarios:

```
lastlog
```

- **Ejemplo:** Verificar intentos fallidos de login:

```
faillog -a
```