ULPGC
Universidad de
Las Palmas de
Gran Canaria

ESCUELA DE
INGENIERÍA INFORMÁTICA

Administración de Sistemas Operativos

Tema 4.3. Notas sobre seguridad lógica

Contenidos

- Ámbitos y políticas de seguridad lógica
- La BIOS
- El cargador (ej. GRUB)
- Cuentas de usuarios y contraseñas
- Cuenta del administrador (root)
- Bits setuid/setgid
- Rutinas de monitorización y auditoría

Objetivos

- Garantizar el acceso al sistema sólo a usuarios autorizados.
- Garantizar que los usuarios acceden solamente a la información y los recursos a los que tienen derecho.
- Un objetivo crucial es **evitar los accesos indebidos** (especialmente las **intrusiones**).

Políticas de seguridad: ámbitos

- ***PERSONAS.*** Qué personas tienen derechos.
- ***RECURSOS.*** Sobre qué recursos tienen derechos (ficheros, procesos, espacio en disco, dispositivos de E/S…)
- ***IDENTIFICACIÓN.*** ¿Cómo se sabe quién es la persona que trata de acceder a un recurso?
- ***PRIVILEGIOS.*** ¿Qué tipos de operaciones están sujetas a restricciones? ¿Cómo se implementan las restricciones?
- ***MONITORIZACIÓN.*** ¿Cómo vigilamos sobre el terreno el cumplimiento de nuestras políticas?
- ***AUDITORÍA.*** ¿Cómo revisamos periódicamente el estado del sistema y los posibles incidentes que hayan ocurrido en el pasado?

Plan de seguridad lógica

- **Perfiles de acceso.** Tipos de usuarios, grupos, privilegios, altas y bajas.
- **Autenticación.** Contraseñas: complejidad mínima, caducidad. Sistemas biométricos, certificados electrónicos.
- **Uso de recursos.** Permisos de acceso, cuotas de espacio y de tiempo, etc.
- **Monitorización y auditoría.** Qué recursos se vigilan, con qué periodicidad.

La BIOS

- BIOS = Basic Input/Output System
- Memoria regrabable con la rutina de arranque y la configuración básica del *firmware.*
- Riesgo: que el usuario arranque desde un dispositivo no convencional (ej. CD, puerto USB)

La BIOS

- La BIOS permite prohibir dispositivos de arranque: USB, disco óptico, etc.
- También inhibir dispositivos de E/S que generan riesgo de intrusión, ej. puertos serie y paralelo.
- La configuración de la BIOS puede protegerse con contraseñas.
- Ojo: las contraseñas se pierden si se retiran las baterías de la placa base. ¡Colocar un candado en la unidad central!

El cargador del SO (GRUB)

- Riesgo: el usuario entra en modo edición y elige arrancar en modo *single user*, o desde una partición elegida por él.
- Solución: proteger con contraseña.

  # grub-md5-crypt

  Editar **/boot/grub/grub.conf** y colocar la contraseña:

  directiva **password --md5** ***contrseña cifrada***

Contraseñas de usuarios

- Unix guarda las contraseñas cifradas en **/etc/shadow**. Algoritmo MD5 (bien).
- Riesgo: contraseñas débiles (adivinables con *ataques de diccionario*)
- Solución: requisitos de dificultad de contraseña
  - mezclar letras, números y signos de puntuación
  - prohibir palabras registradas en un diccionario
  - prohibir palabras que estén en el perfil del usuario (ej. nombre, apellidos, teléfono, cumpleaños…)
- Se configura con PAM: **pam_passwdqc**

Caducidad de contraseñas

- Riesgo: el usuario mantiene durante mucho tiempo la misma contraseña… esto eleva la probabilidad de que alguien la descubra.
- Solución: caducidad forzosa de contraseñas (campos del **/etc/shadow**, **chage**...)

Sesiones abiertas

- Riesgo: un usuario abandona su puesto y se deja una sesión abierta. Oportunidad de oro para un intruso.
- Soluciones:
  - Finalización automática de sesiones ociosas
  - Bloquear la terminal al cabo de un tiempo (y pedir contraseña para desbloquear)

Cuentas *zombis*

- Riesgo: mantenemos cuentas antiguas que ya no se deberían usar. Alguien podría aprovecharlas para entrar como intruso.
- Solución: caducidad de cuentas (**/etc/shadow**)
- Solución: eliminar cuentas no necesarias

Cuentas eliminadas

- Riesgo: si se elimina una cuenta de usuario y alguno de sus archivos permanece en el sistema, un nuevo usuario que adquiera el mismo UID del eliminado podría acceder a ese archivo viejo.
- Solución: buscar archivos sin propietario y eliminarlos (o cambiarles la propiedad)
  - **find / -nouser …**

Cuenta del root

- Riesgo: el administrador entra alegremente como **root** para cualquier cosa. Aumentan los riesgos de intrusión (ej. con troyanos).
- Soluciones:
  - Disciplina: sólo usar **root** cuando sea necesario
  - Temporizador para la sesión de root
  - Usar **/etc/securetty** para restringir las terminales desde las que se puede abrir sesión
  - Usar **/etc/ssh/sshd_config** para restringir SSH
  - En **/etc/passwd** poner como shell **/sbin/nologin**

Programas «su» y «sudo»

- **su** abre un *shell* como **root** (u otro usuario)
  - se puede limitar quién puede ejecutar **su**, si existe el grupo **wheel**. Sólo sus miembros pueden ejecutar **su**. Hay que tocar un módulo PAM.
- **sudo *orden*** ejecuta *orden* como **root**
  - **/etc/sudoers** contiene la lista de usuarios autorizados a ejecutar **sudo**

Permisos «setuid» y «setgid»

- Un proceso Unix maneja dos identidades:
  - **UID real (RUID).** El del usuario que lanzó el proceso.
  - **UID efectivo (EUID).** El que realmente se utiliza para aplicar los permisos de acceso a los recursos.
- Normalmente RUID=EUID. Pero se puede cambiar el EUID si el fichero ejecutable tiene activado el **bit SUID o «setuid»**:
  - Si un fichero ejecutable F propiedad del usuario U tiene activado el bit SUID, cuando alguien ejecute F, el EUID del proceso será el del usuario U.

setuid: ejemplo

- El programa **passwd** cambia la contraseña del usuario. Accede a **/etc/shadow**, que es un fichero que sólo **root** puede tocar.
- Si vemos los permisos de /usr/bin/passwd:
- `-rwsr-xr-x 1 root root 32168 Aug 22 /usr/bin/passwd`
- La “s” significa «setuid activado»
- Si un usuario ejecuta **passwd**, este proceso se ejecuta con el EUID del **root** !!!
- Por suerte, passwd es un programa que no hace *cosas raras*…

setgid

- Igual que existen el *usuario real (RUID)* y el *usuario efectivo (EUID)*, también existen el *grupo real (RGID)*, el *grupo efectivo (EGID)* y el permiso **setgid**.
- El significado es análogo al modelo setuid, pero aplicado a grupos.

Manejar suid/sgid

- Activar/desactivar suid/sgid en un fichero:
  - **chmod u+s miprograma**
  - **chmod g+s miprograma**
  - **chmod 4755 miprograma** *(suid)*
  - **chmod 2755 miprograma** *(sgid)*
- Buscar ficheros suid/sgid:
  - **find /bin /usr/bin -perm -4000** *(suid)*
  - **find /bin -perm -2000** *(sgid)*

Riesgos de setuid/setgid

- Riesgo: alguien consigue colocar un troyano con SUID (ej. una versión modificada de /usr/bin/passwd)
- Solución:
  - Monitorizar qué programas SUID/SGID hay en el sistema, y si hay cambios respecto a los originales (ej. chequeando sumas MD5)

Monitorización y auditoría

- Objetivo: detectar anomalías en la configuración del sistema, o indicios de posibles intrusiones
- Algunas rutinas:
  - Comprobar propietarios y permisos de los archivos de configuración
  - Comprobar propietarios y permisos de los directorios del sistema
  - Verificar la integridad de los binarios del sistema (ej. confrontar con sumas MD5 de los originales)
  - Verificar la presencia o ausencia de archivos importantes

Rutinas de monitorización (ver Tema 6)

- Observar los registros de uso:
  - Intentos fallidos de entrada en el sistema (**lastb**)
  - Intentos de entradas como **root**
  - **.bash_history** del **root**
  - Lo mismo con los usuarios normales
  - Registros del sistema en **/var/log**, sobre todo **/var/log/secure**

ULPGC Universidad de Las Palmas de Gran Canaria | ESCUELA DE INGENIERÍA INFORMÁTICA

Administración de Sistemas Operativos

Tema 4.3. Notas sobre seguridad lógica