Universidad de los Andes
Ingeniería de Sistemas y Computación
ISIS 2203 – Infraestructura Computacional
Taller – Control de Acceso

Propósito. Familiarizarse con el funcionamiento del sistema de protección de Linux/Unix.

Reconocimiento del terreno. En Linux un usuario tiene asociado un nombre y un número para identificación. Además, un usuario pertenece a un grupo primario y puede pertenecer a varios grupos. Cada grupo tiene asociado un nombre y un número para identificación.

1. Inicie la máquina virtual indicada y a continuación inicie una terminal.
2. Establezca su identificación
   2.1. ¿Cuál es su identificador de usuario? Averigüe nombre y número. (use el comando id)
   2.2. ¿Cuál es su identificador de grupo primario? Averigüe nombre y número.
   2.3. ¿A cuántos grupos pertenece el usuario? Averigüe nombres y números.
   2.4. Cuál es su directorio home? (use los comando cd y pwd o echo $HOME)
   2.5. Linux ofrece la utilidad sudo para que un usuario regular pueda correr programas que requieren permisos de administrador(root). Busque información sobre este comando, entienda cuándo se usa y qué hace.
   2.6. Un administrador define en el archivo de configuración de sudo (/etc/sudoers) cuáles usuarios tienen permiso para correr sudo y ejecutar programas con permisos de administrador. Observe la configuración del archivo sudoers.
   2.7. Ejecute el comando

   ```
   sudo more /etc/sudoers
   ```

   para revisar y analizar el contenido del archivo sudoers. Identifique qué parte de la configuración de este archivo es la que le permite a su usuario ejecutar el comando sudo para obtener temporalmente permisos de administrador.
   2.8. ¿Qué ventajas ofrece el uso del comando sudo, comparado con crear un usuario administrador?
   2.9. Para el resto del taller use sudo si requiere correr comandos con permisos de root. Para correr un comando <x> con permisos de root ejecute:

   ```
   sudo <comandox>
   ```

Sistema de Protección. El mecanismo de control de acceso en Unix toma decisiones con base en los identificadores asociados al usuario o proceso que hace un pedido de acceso y los permisos del objeto al que quiere acceder

3. Directorios y archivos
   3.1. El archivo /etc/passwd almacena información de los usuarios en un sistema Linux.

3.1.1. Qué información se almacena acerca de los usuarios?

3.1.2. Ejecute el comando “ls –la /etc/passwd” y observe el resultado. Quién es el propietario del archivo passwd?

3.1.3. Corra el comando cat /etc/passwd. Usted puede ver el contenido aunque el archivo no es suyo. Por qué?

3.2. El archivo /etc/shadow almacena información de las claves de los usuarios del sistema.

3.2.1. Describa la diferencia en permisos de el archivo shadow con el archivo passwd.

3.2.2. Cambiar la clave de usuario es una actividad protegida. Cada usuario puede cambiar su propia clave y solo root puede cambiar las claves de otros usuarios. Cuando un usuario cambia su información (datos de usuario y/o clave), los archivos passwd y shadow cambian.

Cómo se consigue este resultado si los usuarios no tienen permisos de escritura sobre los archivos passwd ni shadow? Ayuda: observe los permisos y el bit setuid del programa passwd e interprételos.

3.3. Manejo y verificación de permisos

3.3.1. Permisos sobre directorios:

3.3.1.1. Cree un directorio y verifique permisos:

(a) cree un directorio ddd en su directorio home (use el comando mkdir),

(b) cambie los permisos para que el dueño del archivo tenga permisos rw, el grupo r y el resto r,

(c) intente entrar al directorio ddd,

(d) cuál es el resultado y por qué?

3.3.2. Permisos sobre directorios.

3.3.2.1. Cambie los permisos del directorio ddd y verifique de nuevo:

(a) adicione permisos de ejecución (+x) al dueño,

(b) intente de nuevo entrar al directorio,

(c) cuál es el resultado y por qué?

3.3.3. Permisos sobre archivos (y directorios):

3.3.3.1. (a) asigne los permisos rwxr-xr-x al directorio ddd,

(b) cree un archivo dentro del directorio: touch f.txt,

(c) salga del directorio y cambie (de nuevo) los permisos del directorio ddd al valor r-xr-xr-x,

(d) liste el contenido del directorio, cuál es el resultado y por qué?

(e) intente borrar el archivo f.txt, cuál es el resultado y por qué?

4. Usuarios

4.1. Cree un usuario nuevo usuario2.

4.1.1. Revise el manual en línea para el comando adduser

4.1.2. Cree el nuevo usuario: (observe que debe remplazar /home/usuario2 será el directorio home para el usuario y /bin/bash es el shell asignado al usuario)

```
adduser --home /home/usuario2 –shell /bin/bash usuario2
```

Necesitará permisos de root para correr el comando.

4.2. Adicione su usuario al grupo primario del segundo usuario (usuario2). Para esto, edite el archivo /etc/group y adicione su usuario al grupo correspondiente. Necesitará permisos de root al editar el archivo.

4.3. Inicie una sesión como usuario2

4.3.1. cree un archivo con touch g.txt y temine la sesión

4.4. Vuelva a su usuario original e intente los siguientes comandos

4.4.1. ls /home/usuario2/g.txt, cuál es el resultado y por qué?

4.4.2. rm /home/usuario2/g.txt, cuál es el resultado y por qué?

4.5. Cree otro usuario usuario3

4.5.1. Asígne el mismo password que asignó a usuario2. Necesitará permisos de root para crear el usuario.

4.5.2. Como es de esperar, Linux no almacena las claves de los usuarios en texto plano, en lugar de esto almacena el hash de la clave. Para tener una barrera adicional usa sal. Verifique el hash que se almacena para cada uno de los nuevos usuarios:

```
sudo grep usuario /etc/shadow
```

4.5.3. Qué tipo de ataques se mitigan con la adición de la sal?