AYUDA A REALIZAR LA ACTIVIDAD 1 : ADMINISTRACIÓN BÁSICA LINUX

- Recurso: VM CyberOps Workstation
- Instrucciones: Entregar en la plataforma de Netacad un único documento en formato .pdf con una captura de pantalla y los comentarios correspondientes por cada uno de los puntos del enunciado.

Usando como la máquina virtual CyberOps Workstation del curso resolver las siguientes cuestiones:

1. ¿Qué tamaño tienen los dispositivos de bloques que referencian a particiones de los discos duros que tiene la máquina virtual CyberOps?

   Se utilizan los comandos: lsblk o fdisk -l
2. Crear el punto de montaje /ACT-A con los siguientes permisos:
   - propietario: analyst
   - grupo: analyst
   - permisos:
     - propietario: rwx
     - grup: rw_
     - otros: rw_

   NOTA: El directorio ACT-A ha de colgar de /

   Un punto de montaje no es más que una carpeta normal, donde se podrá mapear una unidad o volumen (como un disco) para poder acceder a sus contenidos a través de esa carpeta. Para crear una carpeta, usamos el comando mkdir seguido de la ruta y el nombre. Pero, dado que la vamos a crear en la raíz del sistema de archivos (/) necesitamos permisos de administrador con sudo.

   Para cambiar el propietario, usamos el comando chown seguido del nombre de usuario y el archivo o carpeta. Para cambiar el grupo, se usa el comando chgrp junto con el nombre del grupo y la ruta. Como la carpeta fue creada por el usuario administrador, este cambio de propietario debe hacerse con sudo.

   En cuanto a los permisos, hay dos formas de asignarlos mediante el comando chmod:

   • Simbólica o relativa: más intuitiva de entender, consiste en usar una combinación de 3 parámetros:

   ◦Clase de acceso: escogemos entre u (usuario), g (grupo) y o (otros).

   ◦Operador: escogemos entre + (añadir permisos), - (quitar permisos) y = (establecer permisos).

   ◦Tipo de acceso: escogemos entre r (lectura), w (escritura) y x (ejecución)

En un mismo comando se pueden dar los mismos permisos a distintas clases, y se pueden añadir unos permisos y quitar otros a la vez, pero no se puede dar distintos permisos a diferentes clases.

Por ejemplo, “ug+rw-x” añadiría permisos de lectura y escritura al usuario y al grupo, y quitaría el de

ejecución. Este formato es más fácil de entender, pero puede requerir varias invocaciones para

establecer muchos permisos.

• Absoluta: se especifica un número de 3 cifras, cada una en base octal (del 0 al 7). La primera cifra indica los permisos del usuario, la segunda los del grupo y la tercera los de otros. Cada cifra se obtiene sumando los posibles valores de 4 (lectura), 2 (escritura) y 1 (ejecución), siendo un 0 ningún permiso. Por ejemplo, “764” daría todos los permisos al usuario, los de lectura y escritura al grupo y sólo el de lectura a otros. Este formato es más difícil de entender, pero basta una única invocación para establecer todos los permisos.

Por tanto, las dos posibilidades que tenemos son:

chmod usuario=permsos /carpeta

o bien:

chmod xyz /carpeta, siendo x,y,z los permisos en octal de x para el propietario, y para el grupo, y z para otros.

3. Montar el dispositivo de bloques que referencia al segundo disco en el directorio /ACT-A/

   Usaremos en comando:

   mount particion carpeta

4. Hacer una copia del fichero existente en /ACT-A/ con el nombre saludo.txt

Desde la terminal, copiar archivos es tan fácil como usar el comando cp seguido de la ruta completa del origen y el destino. Para facilitar las operaciones, cambiaremos primero al directorio /ACT-A con el comando cd seguido de la ruta (y comprobaremos con pwd y ls que se ha hecho correctamente). De ese modo, ya no tendremos que dar la ruta absoluta cuando trabajemos con archivos dentro de ese directorio, bastará con el nombre.

Si intentamos copiar el archivo ahora mismo, nos encontraremos con un error: al montar una unidad, Linux toma los permisos del sistema de archivos, no de la carpeta donde ha sido montada. Si lo comprobamos ahora veremos que el usuario “root” (administrador) es ahora el dueño de /ACT-A y se han cambiado los permisos a “rwxr-xr-x”. Tendremos que volver a cambiar el propietario, grupo y permisos, usando además el switch -R (recursivo) para que se aplique a todo el contenido del dispositivo. Esto sólo será necesario hacerlo una vez, si volvemos a montar la unidad en un futuro, se recordarán los permisos.

5. Editar el fichero saludo.txt y hacer que sólo aparezca el mensaje "HOLA MUNDO" en el contenido del fichero

Existen dos maneras de hacerlo desde la terminal: la larga sería abrir el .txt con un editor de texto (como por ejemplo vi, vim o nano), hacer los cambios y guardarlo (en nano, sería mediante CTRL+O e INTRO para guardar, y CTRL+X para salir del editor).

Sin embargo hay otra manera más corta y directa, y es usar el comando echo para imprimir “HOLA MUNDO”, y el operador de redirección >, que hace que la terminal tome la salida del último comando y, en vez de imprimirlo en pantalla, lo guarde en un archivo sobreescribiendo los datos anteriores (el operador >> agregaría el texto al contenido previo ya existente en el .txt, pegándolo al final, en vez de sustituirlo todo).

6. Cambiar el nombre del fichero saludo.txt a HOLA.txt

Como se ha visto en el punto 5 y el 6, el comando cat (concatenar) imprime en pantalla los contenidos de un archivo. La desventaja de este comando es que, si el fichero es muy largo, luego tendremos que desplazar la vista de la terminal para poder leerlo entero. Una alternativa mejor para estos archivos con mucho contenido es usar los comandos more o less, que no muestran más de una pantalla a la vez y permiten desplazarse con libertad por el contenido usando los cursores, la barra espaciadora, INTRO, retroceder y avanzar página, y Q para salir ("quit"). En este caso no es necesario pues HOLA.txt sólo tiene 1 línea.

7. Listar el contenido de HOLA.txt

Como se ha visto en el punto 5 y el 6, el comando **cat** (concatenar) imprime en pantalla los contenidos de un archivo. La desventaja de este comando es que, si el fichero es muy largo, luego tendremos que desplazar la vista de la terminal para poder leerlo entero. Una alternativa mejor para estos archivos con mucho contenido es usar los comandos more o less, que no muestran más de una pantalla a la vez y permiten desplazarse con libertad por el contenido usando los cursores, la barra espaciadora, INTRO, retroceder y avanzar página, y Q para salir ("quit"). En este caso no es necesario pues HOLA.txt sólo tiene 1 línea.

8. Comprobar si el servicio ssh está arrancado en el sistema para permitir el acceso en remoto a nuestro sistema.

Tenemos 2 formas de comprobar si el servicio de SSH (sshd, siendo la d por "daemon" o "demonio", un proceso que se ejecuta de fondo y hacer sus tareas sin necesidad de entrada del usuario) está arrancado.

La primera es usar el comando ps para imprimir una lista de procesos en ejecución. Con los parámetros a (ver procesos de todos los usuarios), u (mostrar el usuario propietario de cada proceso) y x (mostrar también procesos que no se están ejecutando en una terminal), el comando nos imprimirá la lista completa, como el Administrador de tareas de Windows pero en texto. Para filtrar esta lista, usamos el comando grep, una herramienta muy útil de Linux que permite encontrar y mostrar lo que queremos en un archivo o salida de comando. Pidiéndole que busque el texto "sshd", imprimirá sólo las líneas que contengan eso.

La segunda opción es, aprovechando que SSH es un servicio dado de alta en systemd (el sistema de gestión de servicios de Linux usado por distribuciones modernas), interrogar por su estado actual con la herramienta systemctl (control de systemd).

9. Simular una acceso remoto, haciendo un ssh desde la máquina virtual hacia nuestra propia máquina virtual. Acceder con el usuario analyst. Mostrar el contenido del directorio /ACT-A

Tenemos 2 formas de comprobar si el servicio de SSH (sshd, siendo la d por "daemon" o "demonio", un proceso que se ejecuta de fondo y hacer sus tareas sin necesidad de entrada del usuario) está arrancado.

La primera es usar el comando ps para imprimir una lista de procesos en ejecución. Con los parámetros a (ver procesos de todos los usuarios), u (mostrar el usuario propietario de cada proceso) y x (mostrar también procesos que no se están ejecutando en una terminal), el comando nos imprimirá la lista completa, como el Administrador de tareas de Windows pero en texto. Para filtrar esta lista, usamos el comando grep, una herramienta muy útil de Linux que permite encontrar y mostrar lo que queremos en un archivo o salida de comando. Pidiéndole que busque el texto "sshd", imprimirá sólo las líneas que contengan eso.

La segunda opción es, aprovechando que SSH es un servicio dado de alta en systemd (el sistema de gestión de servicios de Linux usado por distribuciones modernas), interrogar por su estado actual con la herramienta systemctl (control de systemd).