**Filtra contenido en Linux**

En esta lectura, continuarás explorando los comandos de Linux, que pueden ayudarte a filtrar la información que necesitas. Conocerás un nuevo comando de Linux, **find**, es útil si deseas buscar archivos y directorios para obtener información específica.

**Filtrado para obtener información**

Antes, exploraste cómo el filtrado para obtener información es una habilidad importante para las/los analistas de seguridad. El **filtrado** es la selección de datos que cumplen una determinada condición. Por ejemplo, si tuvieras un virus en el sistema que solo afectara a los archivos **.txt**, podrías usar el filtrado para encontrar estos archivos rápidamente. El filtrado te permite buscar en función de criterios específicos, como la extensión de archivo o una cadena de texto.

**grep**

El comando **grep** busca un archivo especificado y devuelve todas las líneas del archivo que contienen una cadena especificada. El comando **grep** suele combinarse con dos argumentos: una cadena específica a buscar y un archivo específico en el que buscar.

Por ejemplo, si ingresas **grep OS updates.txt**, obtendrás todas las líneas que contienen **OS** en el archivo **updates.txt**. En este ejemplo, **OS** es la cadena específica a buscar y **updates.txt** es el archivo específico en el que buscar.

**pipe (pleca)**

Para acceder a este comando, se utiliza el carácter pleca, o *pipe* (**|**). El comando ***pipe*** envía la salida estándar de un comando como entrada estándar a otro comando, para su posterior procesamiento. Como recordatorio, la **salida estándar** es la información devuelta por el sistema operativo a través del shell, mientras que la **entrada estándar** es la información recibida por el sistema operativo (SO) a través de la línea de comandos.

El carácter pleca (**|**) puede estar en distintas partes de un teclado, pero en general, se encuentra en la misma tecla que el carácter de barra invertida (**\**). En algunos teclados, el carácter **|** puede verse diferente, con un pequeño espacio en el medio de la línea. Si no encuentras el carácter **|**, busca en Internet dónde está ubicado en tu teclado.

Cuando se utiliza con **grep**, la pleca puede ayudarte a encontrar directorios y archivos que contienen una palabra específica en el nombre. Por ejemplo, **ls /home/analyst/reports | grep users** devuelve los nombres de archivos y directorios en el directorio **reports** que contienen la palabra **users**. Antes de la pleca, **ls** indica que debe hacerse una lista de los nombres de los archivos y directorios en **reports**. Luego, esta salida se envía, como entrada, al comando ubicado después de la pleca. En este caso, **grep users** devolverá todos los nombres de archivos o directorios que contienen **users** de la entrada recibida.

**Nota:** El comando **pipe** es una forma general de redireccionamiento en Linux y puede utilizarse para múltiples tareas, además del filtrado. Puedes considerarlo como una herramienta general a utilizar, siempre que quieras que la salida de un comando se convierta en la entrada de otro comando.

**find**

El comando **find** sirve para buscar directorios y archivos que cumplan con criterios especificados. Hay una amplia variedad de criterios que se pueden especificar con **find**. Por ejemplo, puedes buscar archivos y directorios que:

* Contengan una cadena específica en el nombre;
* Tengan un determinado tamaño de archivo; o
* Se hayan modificado por última vez dentro de un período de tiempo determinado.

Cuando usas **find**, el primer argumento después de **find** indica dónde comenzar a buscar. Por ejemplo, al ingresar **find /home/analyst/projects** se busca todo lo que comienza en el directorio **projects**.

Después de este primer argumento, debes indicar tus criterios para la búsqueda. Si no incluyes un criterio de búsqueda específico con tu segundo argumento, es probable que tu búsqueda devuelva una gran cantidad de directorios y archivos.

La especificación de criterios implica opciones. Las **opciones** modifican el comportamiento de un comando y, por lo general, comienzan con un guión (**-**).

**-name e -iname**

Un criterio clave que las/los analistas podrían usar con **find** es buscar nombres de archivos o directorios que contengan una cadena específica. Debes ingresar la cadena específica que estés buscando entre comillas después de las opciones **-name** o **-iname**. La diferencia entre estas dos opciones es que **-name** distingue entre mayúsculas y minúsculas, mientras que **-iname** no lo hace.

Supongamos que quieres buscar todos los archivos en el directorio **projects** que contienen la palabra “log” en el nombre. Para hacerlo, ingresarías **find /home/analyst/projects -name "\*log\*"**. También podrías ingresar **find /home/analyst/projects -iname "\*log\*"**.

En estos ejemplos, la salida estaría conformada por todos los archivos del directorio **projects** que contengan **log** rodeada de cero o más caracteres. La parte del comando que dice **"\*log\*"** es el criterio de búsqueda que indica que se debe buscar la cadena “log”. Cuando la opción es **-name**, no se devolverán los archivos con nombres que incluyan **Log** o **LOG**, por ejemplo, porque esta opción distingue entre mayúsculas y minúsculas. Sin embargo, sí se devolverán si la opción es **-iname**.

**Nota**: El asterisco (**\***) se usa como comodín para representar cero o más caracteres desconocidos.

**-mtime**

Las/los analistas de seguridad también pueden usar **find** para buscar archivos o directorios modificados por última vez, en un período de tiempo determinado. Para esta búsqueda, se puede utilizar la opción **-mtime**. Por ejemplo, al ingresar **find /home/analyst/projects -mtime -3**, se devuelven todos los archivos y directorios del directorio **projects** que han sido modificados en los últimos tres días.

La búsqueda de la opción **-mtime** se basa en días, por lo que la entrada **-mtime +1** indica todos los archivos o directorios que se modificaron por última vez hace más de un día y la entrada **-mtime -1** indica todos los archivos o directorios que se modificaron por última vez hace menos de un día.

**Nota:** Si quieres basar la búsqueda en minutos en lugar de días, puedes usar la opción **-mmin** en lugar de **-mtime**.

**Conclusiones clave**

Saber filtrar información con comandos de Linux es una habilidad importante para las/los analistas de seguridad, ya que les permite personalizar los datos de modo que se ajusten a sus necesidades. Tres comandos clave de Linux para esto son **grep**, el uso de la pleca (**|**) y **find**. Estos comandos sirven para navegar por el sistema de archivos y filtrar para obtener información en este.