Informática Forense y Auditoría

Grado en Ingeniería Informática del Software

Escuela de Ingeniería Informática de Oviedo

Práctica 2: Ejercicios de Práctica Forense I

Objetivos

En los siguientes ejercicios, pretendemos alcanzar, entre otros, los siguientes objetivos:

- Obtener la máxima información posible de un sistema en vivo en Linux.
- Comprender la nomenclatura que utiliza Linux para nombrar a los ficheros que representan dispositivos de almacenamiento y sus particiones.
- Realizar firmas hashs que permitan asegurar la integridad de la información.
- Realizar una imagen de un dispositivo.
- Realizar una imagen de la memoria RAM de una máquina Linux.
- Aprender a montar particiones tratando de no alterar la información presente en las mismas.
- Familiarizar al alumno con el contenido de archivos de registro.
- Localizar los principales archivos de registro de un sistema Linux.
- Monitorizar archivos de registro en tiempo real.
- Extraer información de los archivos de registro utilizando expresiones regulares.
- Realizar la búsqueda de cadenas en ficheros.
- Comenzar a utilizar Autopsy como herramienta de análisis forense.

Instrucciones comunes a todos los ejercicios

Deberá justifique la realización de los siguientes ejercicios con capturas de pantalla donde quede patente que el ejercicio se realiza con la cuenta del alumno creada en la práctica anterior, así como los comandos empleados en la resolución de los mismos y las explicaciones que considere oportunas.

Firmado por GARCIA TAMARGO, MARCO ANTONIO (FIRMA) el día 03/09/2024 con un certificado emitido por AC DNIE

Obteniendo información de un sistema en vivo

En caso de encontrarnos el sistema encendido, el primer paso que debemos realizar en estas situaciones es obtener las principales características del sistema afectado además de cualquier información volátil que pueda desaparecer en el momento que apaguemos el mismo. Para esto, los sistemas Linux disponen de un gran número de instrucciones que nos permiten realizar estas tareas de una forma sencilla.

Realice los siguientes ejercicios en la máquina virtual en la que ha instalado la versión escritorio de Ubuntu (**IFA-UD-XX**).

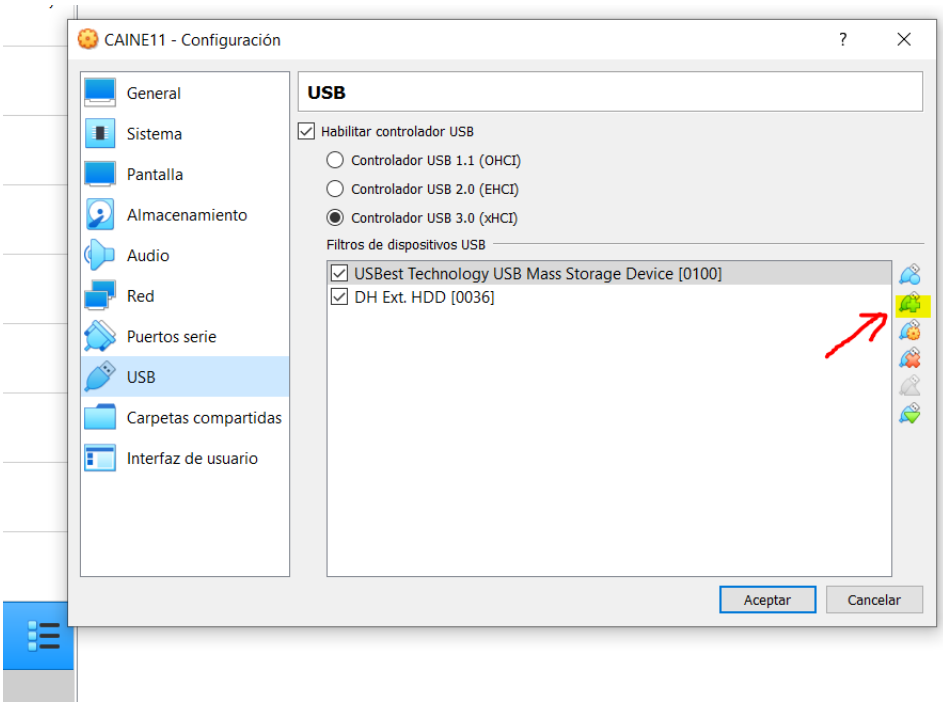
1. Abra una terminal y obtenga la fecha y la hora del sistema intervenido. Compare dicha hora con la hora oficial de España y anote cualquier diferencia.
2. Averiguar la versión del kernel y versión del sistema operativo.
3. Obtener un listado de todos los interfaces de red presentes en el sistema y sus características.
4. Obtener el listado de todas las conexiones existentes con la máquina, tanto aquellas de entrada como las de salida. Además, deberá mostrar información de cada una de ellas, como por ejemplo los procesos que las están utilizando
5. Averiguar qué usuarios están logueados y cuánto tiempo lleva el sistema arrancado.
6. Mostrar la información de la tabla de enrutamiento existente.
7. Mostrar los procesos que están corriendo en el sistema de todos los usuarios sin que se corte el nombre del proceso si es más largo que el ancho de la pantalla.
8. Visualizar cuál fue el último acceso de cada usuario al sistema y el registro de accesos de un usuario al sistema. Obtener la lista de accesos fallidos al sistema.
9. Mostrar los ficheros abiertos actualmente en el sistema.
10. Mostrar información de los sistemas de ficheros que utiliza el equipo en los diferentes discos y las diferentes opciones de montaje.
11. Obtener las diferentes particiones existentes en el disco duro /dev/sda, así como los sectores de inicio y fin de cada una de ellas.

Creación de imágenes y firmas hash

Una parte de la adquisición forense consiste en la creación de imágenes forenses, tanto de dispositivos de almacenamiento masivo completos, como de sus particiones. Realice los siguientes ejercicios en la máquina virtual en la que ha instalado CAINE (**IFA-AU-XX**).

Como paso previo, conecte a su máquina anfitrión (máquina Windows 10) un lápiz USB (cuanto más pequeño, mejor) del cual realizará posteriormente una firma hash y una imagen. En el apartado USB de la configuración de su máquina virtual habilite el controlador USB y seleccione como controlador USB el 3.0 (xHCI). A continuación,

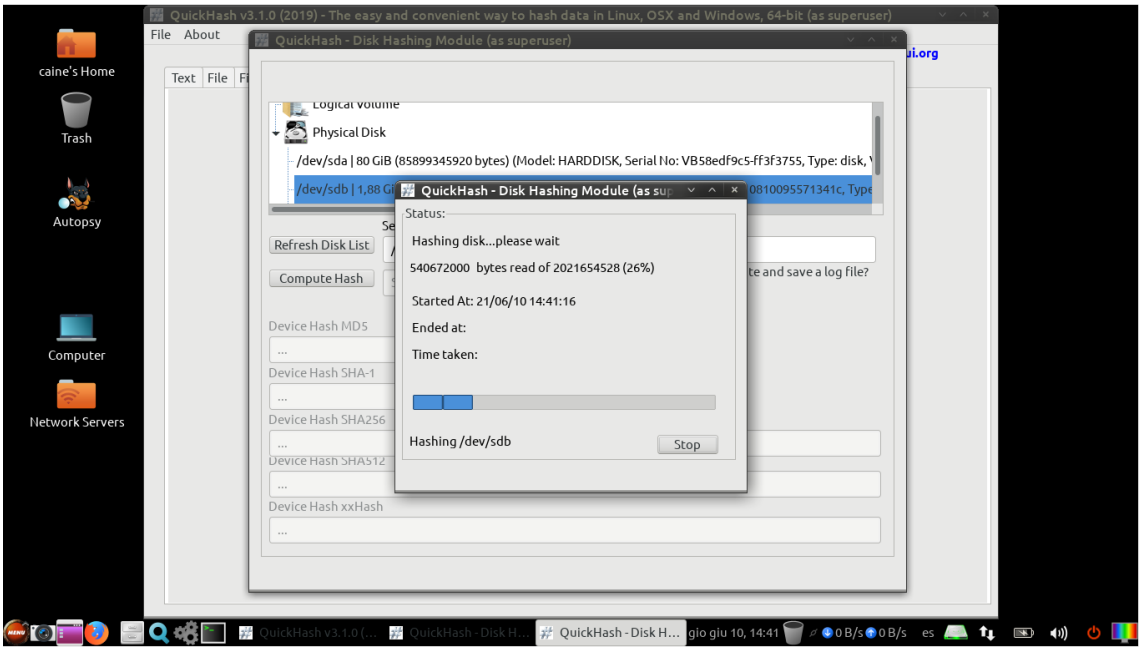
agrega un filtro para el lápiz de memoria que quieras conectar a tu máquina virtual, el cual, previamente deberá estar conectado al anfitrión (equipo en el que te has sentado).



Una vez hayas agregado el filtro para el nuevo dispositivo inicia la máquina virtual de CAINE. Una vez haya arrancado y te hayas logueado, presiona el botón izquierdo del ratón en la utilidad **Mounter** que se encuentra en la barra de tareas (representada por un icono de disco duro). Podrás ver si este programa reconoce las distintas particiones existentes en el dispositivo conectado. **Por defecto, CAINE tiene deshabilitado el automontaje de particiones, con lo cual, es poco probable alterar el contenido de las mismas, aunque para más seguridad debemos utilizar un write-blocker siempre que sea posible.**

12. Averigua qué comando de Linux nos proporciona información sobre si un dispositivo ha sido reconocido por el sistema.
13. Vaya al icono del menú en la esquina inferior izquierda de la pantalla. Seleccione **Herramientas forenses** y dentro de dicha categoría **Hash**. Lance la aplicación **QuickHash** y una vez dentro de la misma seleccione la pestaña **Discos**. Pulse el botón que encontrará en dicha pestaña. Despliegue en la ventana que aparecerá la categoría **Discos Físicos**. Seleccione el nombre lógico del dispositivo (/dev/sd**X**) correspondiente al lápiz de memoria y seleccione como función hash **SHA1**. Calcule el hash para dicho dispositivo y almacénelo. Anote el tiempo que tarda en calcular el hash para dicho dispositivo y calcule el tiempo de hash por GB copiado.

	Hash SHA1	Tiempo de realización	Tiempo por GB
Dispositivo Físico			



14. Obtén la firma hash del lápiz USB **desde línea de comando**. Realiza la firma hash utilizando el algoritmo SHA-1 y luego el algoritmo MD5. Almacena cada firma hash en un fichero distinto.

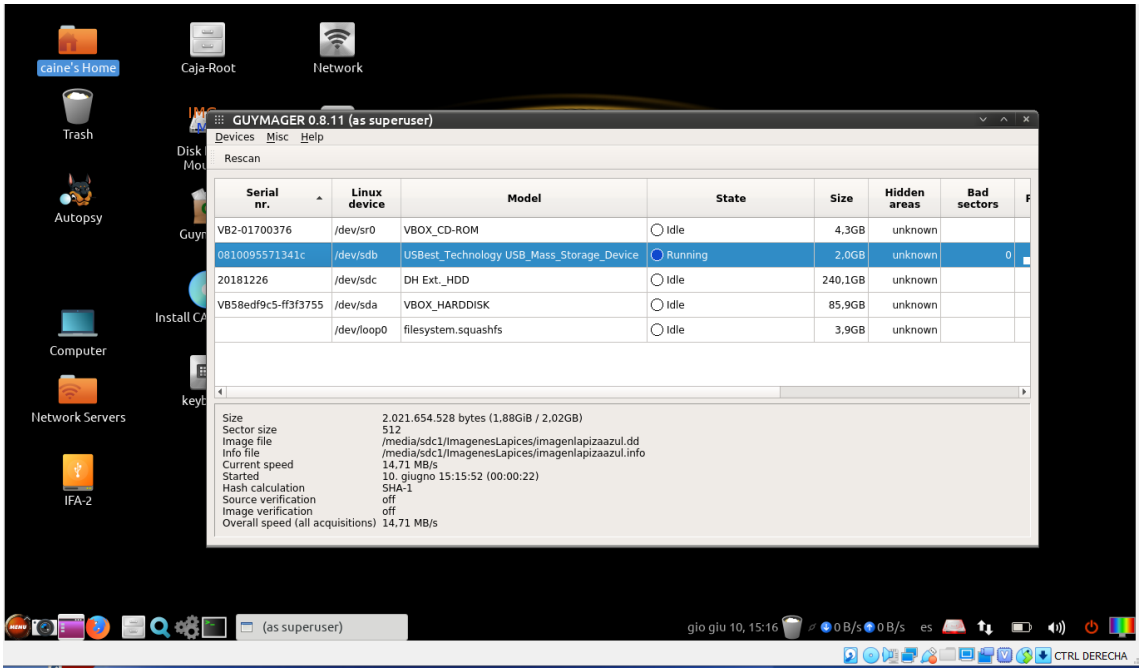
15. Averigua cuántas particiones tiene el lápiz USB. Para ello puede utilizar la herramienta **Mounter** o bien el comando que utilizaste en el ejercicio 11 de esta práctica. Realiza una firma hash del fichero de dispositivo correspondiente a dicha partición, primero utilizando el **hash SHA-1** y luego el **MD5** y almacena cada firma hash en un fichero distinto. Finalmente, monta dicha partición en **modo sólo lectura**. **Para montar la partición puedes utilizar bien la herramienta Mounter o bien puedes utilizar la línea de comandos**.

16. En caso de que el pendrive contuviese una única partición, ¿coincidiría la firma hash de ésta con la firma hash del fichero de dispositivo que representa al lápiz? **Razónese la respuesta y demuéstrese empíricamente**.

17. Realiza una imagen física completa del lápiz USB. Puedes realizar la imagen con la herramienta **GuyMager** o desde la línea de comando. Seleccione **Herramientas forenses** y dentro de dicha categoría **Discos**. Lance la aplicación **GuyMager**. Seleccione el dispositivo del cual va a realizar la imagen forense (primer lápiz USB conectado a su equipo anfitrión). Una vez seleccionado pulse con el botón derecho sobre el mismo y en el menú contextual seleccione la opción **Adquirir imagen**. En la ventana que aparecerá a continuación seleccione la opción **Linux dd raw image**. Deje marcada la opción de dividir el archivo de imagen en trozos de 2GB si la unidad donde va a almacenar su imagen está formateada en FAT32. Seleccione la ruta donde va a almacenar la imagen (puede almacenar dicha imagen en la carpeta

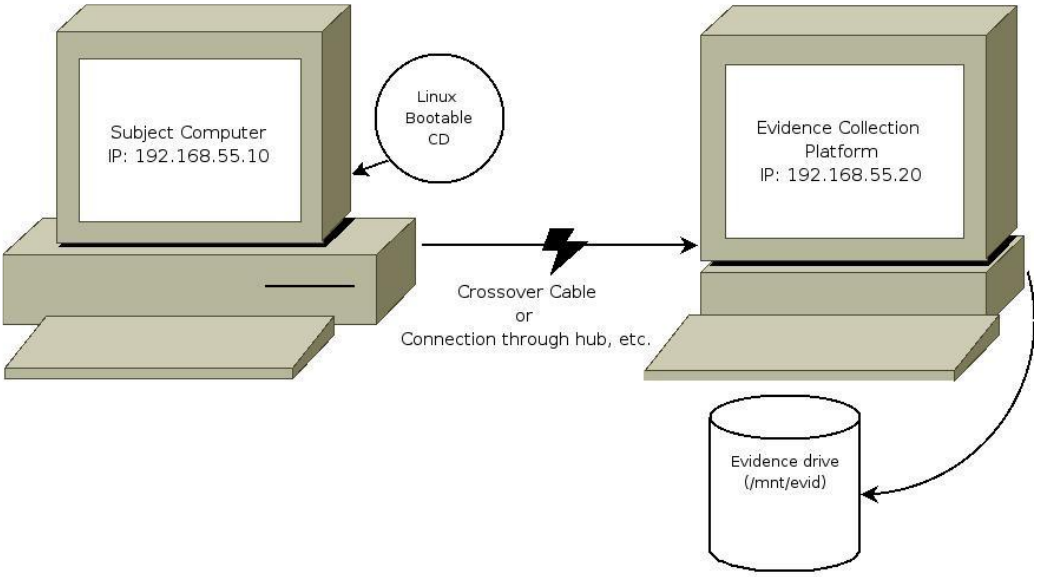
compartida por el equipo anfitrión o bien en un segundo lápiz de memoria o disco duro externo conectado a su máquina virtual). Introduzca el nombre del fichero donde va a almacenar la imagen, así como el nombre del fichero de información de la imagen. Marque la casilla correspondiente a calcular el hash SHA1 y MD5 así como la casilla de verificar la imagen después de la adquisición. Anote el tiempo que le lleva realizar la imagen. Una vez concluida la realización de la imagen compare el hash obtenido en el ejercicio 14 de esta práctica. ¿Son iguales las firmas hash?

	Hash SHA1	Tiempo de realización
Dispositivo (ejercicio 14)		
Imagen dispositivo (este ejercicio)		



18. Realice una imagen de la partición presente en el lápiz anterior con el nombre imagen-p**X**.lapiz (ej.: imagen-p1.kingston) utilizando el comando adecuado, donde **X** es el número de partición de la que realiza la imagen.
19. Compruebe si el fichero de imagen creada en el ejercicio 18 tiene la misma firma hash que la calculada en el ejercicio 15 para la partición.
20. Montar en bucle, en modo lectura y sin permiso de ejecución la imagen de la partición del lápiz creada en el ejercicio 18.
21. En algunas ocasiones es necesario adquirir las evidencias de un sistema utilizando un disco de arranque y una conexión de red a la cual está conectada la plataforma de recolección de evidencias. El ordenador del cual crearemos la imagen lo llamaremos ordenador "**objetivo**" y en el que almacenaremos la imagen lo llamaremos ordenador "**recolector de evidencias**". Para poder realizar la imagen a

través de la red necesitaremos en primer lugar hacer que el "**recolector de evidencias**" escuche el flujo de datos proveniente del ordenador "objetivo". Esto puede hacerse mediante el comando netcat (**nc**).



El primer paso será abrir una conexión de escucha en el "recolector de evidencias" y redirigir todos los datos recibidos en dicha conexión al comando **dd**. En la computadora objetivo debemos ejecutar el comando **dd** tomando como fichero de entrada el fichero que representa el disco (o la partición) del cual queremos hacer la imagen y en lugar de suministrar un fichero de salida canalizaremos la salida al comando **nc** en la dirección IP y puerto en la que está esperando el comando homónimo en la máquina "**recolector de evidencias**".

Para probar esta técnica, vamos a hacer una imagen de un dispositivo conectado a su máquina virtual **IFA-UD-XX** en su máquina virtual **IFA-AU-XX**. Para ello modifique los interfaces de red de ambas máquinas y colóquelos en modo "**adaptador puente**". En segundo lugar añada un filtro para el lápiz USB que va a conectar a la máquina **IFA-UD-XX**. Si ya tenía un filtro creado para dicho dispositivo en la máquina **IFA-AU-XX**, elimínelo primero. Una vez añadido el filtro, conecte dicho dispositivo a la máquina **IFA-UD-XX** y compruebe que ha sido detectado por el Sistema Operativo de dicha máquina. Haga un hash del dispositivo del cual va a crear la imagen antes de realizarla. Luego haga la imagen utilizando el procedimiento descrito anteriormente para lo cual tendrá que averiguar la IP de la máquina que asume el rol de "**recolector de evidencias**". Una vez concluido el proceso de realización de la imagen, haga un hash en destino del fichero de imagen y compruebe si coincide con el hash del dispositivo del cual ha realizado la imagen en origen.

22. Repita el ejercicio 17 repartiendo el fichero imagen en trozos de 100 MB a medida que se van generando. Dichos trozos tendrán como prefijo **trozo_pendrive.**

23. Reensamble las imágenes creadas en el ejercicio 22 en un único archivo denominado **imagen_nueva.dd**. Calcule el hash SHA1 de la imagen reensamblada y compáralo con el hash del lápiz obtenido en el ejercicio 17. ¿Son iguales?
24. Apague la máquina virtual de forma ordenada desde la opción Apagar del botón Menú que se encuentra en la esquina inferior izquierda de la pantalla. Repita de nuevo el ejercicio 13 de esta práctica utilizando el mismo lápiz USB. ¿Son iguales las firmas hash que obtuvo al realizar inicialmente el ejercicio 13 y al volver a realizarlo ahora?

	Hash SHA1
Ejercicio 13	
Ejercicio 24	

25. Realice una imagen de la memoria RAM de la máquina Linux. Para ello compruebe en primer lugar el valor del parámetro **CONFIG_STRICT_DEVMEM** que se encuentra en el fichero **/boot/config-versióndelkernel**. Averigüe la versión del kernel para lo cual repase el ejercicio 2 de esta práctica. Si el parámetro anterior toma como valor '**n'** podrá copiar el contenido de la memoria con el mismo comando que ha utilizado para copiar discos duros o particiones sin ninguna restricción. En caso de que este parámetro tome como valor '**y'**, solamente podrá copia el primer MB de la memoria RAM. En este segundo caso, para poder copiar toda la memoria RAM sin ninguna restricción deberá primero instalar el paquete **lime-forensics-dkms**. Una vez lo haya instalado, necesitará cargar el módulo del kernel **lime.ko** para lo cual deberá usar el comando **insmod**.

Extracción de información de los archivos de registro (log)

Los archivos de registro, son archivos que los sistemas operativos utilizan para registrar eventos. El sistema operativo, los programas, los procesos en segundo plano, los servicios o las transacciones entre servicios, pueden generar tales eventos. Los archivos de registro dependen de la aplicación que los genera. En las investigaciones forenses es conveniente saber dónde se encuentran tales ficheros y saber seleccionar información de los mismos mediante expresiones regulares.

26. Cualquier software puede conservar archivos de registro, incluido el propio sistema operativo. Por convenio, Linux utiliza el directorio **/var/log** para almacenar diversos archivos de registro (la mayoría de ellos gestionados por **rsyslog**), incluidos los del sistema operativo. Los sistemas operativos modernos son complejos componentes de software y, por ende, emplean varios archivos diferentes para registrar eventos.

    a) Busca información sobre el propósito de los ficheros de log que figuran en la siguiente tabla.

Fichero	Propósito
syslog	
auth.log	
kern.log	
mail.log	
lastlog	
btmp	
wtmp	

27. Localizar y examinar el fichero de log donde el kernel almacena las acciones realizadas por "cron" (programador de tareas en sistemas Unix/Linux) y que recibe la mayoría de eventos del sistema.

28. Descarga del campus virtual el fichero denominado Recursos de **Prácticas->Práctica 2-> logstash-tutorial.log**. Este fichero es un fichero de muestra de la salida de un servidor Web. Observe el contenido del mismo y realice los siguientes apartados.
    a) ¿Cuántas líneas tiene el fichero?
    b) El formato de las líneas de salida viene determinado por la aplicación que la genera y la configuración que haga de la misma su administrador. El formato de salida de los ficheros de log de Nginx es muy similar al formato de salida de los ficheros de Log generados por Apache.

```
86.1.76.62 - - [04/Jan/2015:05:30:37 +0000] "GET /style2.css HTTP/1.1" 200 4877
"http://www.semicomplete.com/projects/xdotool/" "Mozilla/5.0 (X11; Linux x86_64;
 rv:24.0) Gecko/20140205 Firefox/24.0 Iceweasel/24.3.0"
```

    Cada línea de salida de este fichero está formada por diferentes campos separados por espacios en blanco. La descripción de estos campos es la siguiente:

    - **IP origen**: Dirección IP del cliente que realiza la petición.
    - **Identidad del cliente**: Basada en el estándar **RFC 1413**. Esta información no es fiable salvo en redes internas. Si en su lugar hay un guion indica que esta información no se encuentra disponible.
    - **Identificador del usuario**: Identificador del usuario que realiza la petición como determina la autentificación HTTP. Si el recurso no está protegido por contraseña, aquí figurará un guion.
    - **Fecha/hora**: Fecha hora en que fue recibida la petición por parte del servidor Web.
    - **Solicitud**: La solicitud HTTP enviada por el cliente. La solicitud incluye el verbo HTTP empleado (GET, POST, PUT, etc.) así como el recurso solicitado y la versión del protocolo HTTP empleado por el cliente.
    - **Código de estado**: Enviado por el servidor al cliente.
    - **Número de bytes enviados desde el servidor al cliente**. Este valor no incluye el tamaño de las cabeceras de respuesta.
    - **Referer**: La página desde la cual se incluye o se refiere el recurso solicitado.

- **Identificador del navegador del cliente**.

c) A la hora de extraer información de archivos de registro es bueno saber cómo trabajar con expresiones regulares. Una expresión regular (regex) es un patrón de símbolos que describe datos que deben coincidir en una consulta o en cualquier otra operación. Las expresiones regulares se construyen en forma similar a las aritméticas, utilizando diversos operadores para combinar expresiones más pequeñas. Abre un navegador web y navega a https://regexone.com/. **Regex One** es un tutorial que ofrece lecciones para aprender sobre los patrones de las expresiones regulares. **Realiza el tutorial**.

d) Busca ayuda en el man sobre el comando **grep**. Utilizando el comando **grep**, aplica las expresiones regulares de la siguiente tabla al fichero **logstash-tutorial.log** y describe la información obtenida.

Patrón de expresión regular	**Descripción**	**Número líneas que hacen match**
^83		
[A-Z]{2,4}		
2015		
05:22:2[0-9]		
\.com		
complete|GET		
0{4}		

29. Los archivos de registro se pueden mostrar con muchas herramientas de presentación de texto. Aunque se puede utilizar **cat**, **more**, **less** y **vi** para trabajar con archivos de registro, no son adecuados para monitorearlos en tiempo real. Los desarrolladores de Unix diseñaron diversas herramientas que permiten monitorear archivos de registro en tiempo real. En los siguientes apartados de este ejercicio veremos el uso de la herramienta **tail** (busca información en el man sobre la misma), una herramienta simple pero eficiente que está disponible prácticamente en todos los sistemas basados en Linux/Unix.

a) El comando **tail** muestra por defecto las últimas 10 líneas del fichero que se le pasa como argumento. Utiliza el comando tail para mostrar las 10 últimas líneas del fichero **/var/log/syslog**

b) Utiliza la opción **–n** del comando **tail** para especificar el número exacto de líneas desde el final del archivo que quieres visualizar. Utiliza esta opción para visualizar las 5 últimas líneas del fichero **/var/log/syslog**

El comando tail también te permite monitorear en tiempo real los cambios que se van produciendo en un fichero de log. Para ello utiliza la opción **–f** (follow). Para probar esta opción vamos a trabajar con el servidor web **nginx**. Ni la máquina virtual de CAINE (**IFA-UD-XX**) ni la máquina Ubuntu (**IFA-UD-XX**) tienen este

servidor entre el software preinstalado. Instale el servidor **nginx** ejecutando el siguiente comando:

sudo apt install nginx

Comprueba si está en ejecución utilizando el siguiente comando:

sudo systemctl status nginx

El servidor Web **nginx** genera sus mensajes de log en el fichero **/var/log/nginx/access.log**. Abre una terminal y desde ella monitoriza de forma continua dicho fichero.

c) Abre una ventana del navegador web e introduce en la misma **127.0.0.1**. Deberás comprobar como por cada petición que realices a tu servidor web se genera una nueva línea de salida en el log con el formato descrito en el apartado b) del ejercicio 28.

30. En Linux, el proceso **init** es el primero que se carga cuando arranca la computadora. Directa o indirectamente, **init** es el padre de todos los procesos que se ejecutan en el sistema. **Systemd** es un moderno sistema **init** diseñado para unificar la configuración y el comportamiento de los servicios de Linux en todas sus distribuciones. **system-journald** (o simplemente **journald**) es el servicio de registro de eventos de **systemd** y utiliza archivos binarios "**append-only**" (solo anexar) que actúan como sus archivos de registro. Hay que mencionar que **journald** no impide que el uso de otros sistemas de archivos de registro como **syslog** o **rsyslog**. En este ejercicio trabajaremos con **journalctl**, una utilidad de **journald** que se utiliza para visualizar archivos de registro y monitorearlos en tiempo real.

a) En una terminal escribe el comando **journalctl** sin opciones para mostrar todas las entradas de registro de diario (puede ser una lista bastante larga). La salida comienza con una línea que señala la marca de hora en la que el sistema comenzó las operaciones de registro.

Journalctl incluye numerosas funcionalidades como desplazamiento de páginas y mensajes codificados por colores, entre otras. Utiliza las teclas de flechas hacia arriba y hacia abajo del teclado para desplazarte por la salida, una línea de cada vez vez. Utiliza las teclas de las flechas hacia la izquierda/derecha del teclado para desplazarte lateralmente y mostrar entradas de registro que se extienden por fuera de los límites de la ventana del terminal. La tecla **<INTRO>** muestra la siguiente línea, mientras que la barra espaciadora muestra la siguiente página de la salida. Presiona la tecla **q** para salir de **journalctl**.

b) **journalctl** incluye opciones que ayudan a filtrar el resultado. Utilice la opción **–b** para mostrar entradas de registro relacionadas con el último arranque.

c) Averigüe si es posible obtener entradas de registro relacionadas con arranques pasados. Muestre las entradas correspondientes a hace 2 arranques.
d) Lanza el comando **journalctl** con la opción **–list-boots** para generar una lista de arranques anteriores.
e) Utilice **--since "<time range>"** para especificar el intervalo de tiempo para el cual se deben mostrar entradas de registro. Muestre todos los registros generados desde hace 2 horas. Muestre todas las entradas de registro generadas en el último día.
f) Journalctl también permite mostrar entradas de registro relacionadas con un servicio específico si se utiliza la opción **–u**. Investiga esta opción para mostrar entradas de registro relacionadas con el servidor web nginx.
g) Similar a **tail -f**, journalctl también da soporte de monitoreo en tiempo real. Utilicen la opción **–f** para monitorear en tiempo real los registros generados por el sistema. Para ello, desde otra terminal lance el comando **ssh localhost** (instale para ello el servidor ssh si no está instalado) e introduzca su contraseña. Verá cómo se genera un nuevo registro para el nuevo inicio de sesión.

Búsqueda de archivos y cadenas

Una vez realizada la adquisición forense, gran parte del trabajo del investigador forense se resume en buscar información que se encuentra en las evidencias forenses, bien en el espacio asignado (accesible a través del sistema de ficheros) o bien en el espacio sin asignar (bloques no asignados a ningún archivo).

En muchos casos tendremos que realizar la búsqueda de cadenas de caracteres, que pueden estar codificados siguiendo diversos estándares como ASCII7, ISO-8859-1 o UNICODE. La búsqueda se puede realizar con herramientas de línea de comando (**grep**) o bien utilizando software forense que disponga de capacidades especiales para realizar este tipo de tareas.

31. Normalmente los usuarios Linux suelen tener como Shell el Bash. Localiza el archivo que contiene el historial de los comandos introducidos por un usuario. **NOTA: Realice esta práctica en la máquina virtual IFA-UD-XX con cualquier usuario que haya creado en ella.**
32. Haz una lista con todos los ficheros y directorios que contiene la partición montada en bucle en el ejercicio 20. Si desmontaste dicha partición, vuélvela a montar en modo bucle y en sólo lectura, tal y como hiciste en el citado ejercicio. Los ficheros deberán mostrar su número de inodo y la lista deberá estar ordenada por tiempo de acceso.
33. Almacene en un fichero el tipo real de todos los ficheros contenidos en la partición montada en bucle en el ejercicio 20.
34. Descargue del campus virtual el fichero denominado **Recursos de Prácticas->Práctica 2->logs.v3.tar.gz**. Destarea y descomprime el anterior archivo en una

carpeta denominada logs. Deberás ver 5 ficheros de log de diferentes sistemas Unix. Estos ficheros de log contienen entradas correspondientes a una gran variedad de fuentes, incluyendo el kernel y otras aplicaciones. Crea un pipeline que muestre las fechas (mes y día) en las que ha habido apuntes en los respectivos logs de forma descendente (de más reciente a menos reciente) y que elimine las entradas múltiples (las repetidas para una misma fecha).

35. Busca, en los fichero de log anteriores, todas las entradas correspondientes al **13 de Noviembre**.
36. Busca, en los ficheros de log anteriores, todas las entradas en las que aparezca "**Did not receive identification string from**" y mostrar para cada una de ellas el mes, día, hora e IP.
37. Descargue del Campus Virtual una imagen denominada **Recursos de Prácticas->Práctica 2->minimagen.dd**. Suponga que dicha imagen es la de un dispositivo que pertenece a un empleado de una multinacional que amenaza a la misma con desatar un virus si no se le paga un rescate de **50000$**. Para resolver este ejercicio utilizaremos en primer lugar herramientas de línea de comandos. Busque en el contenido de la imagen (en el espacio no asignado y en el espacio de fragmentación interna de los archivos que se puedan encontrar) palabras clave como "**$50,000**", "**virus**" o "**ransom**" (rescate) o variaciones de las mismas, tanto en mayúsculas como e minúsculas. Investigue para ello las posibilidades que le ofrece el comando grep y sus diferentes opciones. Almacene los resultados obtenidos en un fichero de salida para luego analizar lo encontrado.

38. Descarga del campus virtual **Recursos de Prácticas->Práctica 2->RTR.E01**. Se trata de una imagen de un disco duro Western Digital realizada con el software forense EnCASE. Almacénelo en una carpeta de Evidencias. Abra la utilidad Autopsy desde **Menú->Forensic Tools->Autopsy**. Cree un nuevo caso desde el interfaz de Autopsy. Llame al nuevo caso de la siguiente manera: **EjercicioXX_StringSearch_Apellidos_Nombre**, donde XX es el número del ejercicio que está realizando en este momento. En el número de caso ponga DDMMAAAA-XX donde DD es el día en el que realiza el ejercicio, MM es el número del mes actual y AAAA es el año actual, siendo XX el número del ejercicio que está realizando. Ponga su nombre y sus datos en la información del examinador (ponga en el correo su dirección de email de uniovi). Añada al caso la evidencia **RTR.E01** como Imagen de Disco o fichero VM.

    Investigue qué posibilidades ofrecen los módulos de ingestión de Autopsy siguientes: **File Type Identification, Keyword Search** y añádalos como módulos de ingestión de evidencia asociados al proyecto. En el módulo de ingestión **Keyword Search** cree una nueva lista de palabras denominada **P2-EJ38**. El caso que se está investigando es el robo y la destrucción de la nueva carta de menú de un restaurante ruso. Lo único que recuerda el personal del restaurante son las

secciones que contendría dicho menú, pero desconoce mayormente el contenido de cada una de ellas. Las cadenas a buscar son expresiones en inglés codificadas en Unicode **UTF16BE** y son las siguientes:

- Appetizers
- Soup
- Pancakes
- Meat pies and dumplings
- Meat and fish
- Cheese and milk products
- Beverages
- Dessert

Configure, en función de la información aportada, el módulo de ingestión de búsqueda de cadenas de Autopsy e intente reconstruir, tanto como le permita Autopsy, el menú robado, tanto en inglés como los nombres de las diferentes entradas en ruso.

39. Repita el ejercicio 37 de esta práctica, pero en este caso realice la búsqueda de las palabras clave desde Autopsy. Cree un nuevo caso en Autopsy y busque en el contenido de la imagen palabras clave como "**$50,000**", "**virus**" o "**ransom**" (rescate) o variaciones de las mismas que puedan relacionar al empleado con el chantaje. Responda además a las siguientes cuestiones, para lo cual es posible que tenga que añadir a su proyecto en Autopsy módulos de ingestión adicionales:
    a) ¿Cuántas cadenas con formato de URL fueron detectadas en la imagen forense?
    b) ¿Cómo se llama el fichero borrado en el que se encuentra la carta amenazante?
    c) ¿Qué tamaño ocupa en bytes?
    d) ¿En qué sector empieza dentro de la imagen?
    e) ¿Cuántos sectores ocupa?
    f) Indique cuál es el destinatario de la carta amenazante.
    g) ¿Quién firma la carta?
    h) ¿Qué codificación tiene el archivo?
    i) ¿En qué fecha y hora fue modificado el archivo?
    j) Localice si en el disquete del sospechoso hay algún archivo que pueda contener un virus. Para ello examine las direcciones de correo.
    k) Analice el archivo/s sospechosos con el ClamTK (instale este paquete si no viene instalado con CAINE) e indique, en caso de encontrar un archivo con virus, de qué tipo es y en qué lenguaje está escrito.