ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

ELABORACIÓN DE RECOMENDACIONES DE BUENAS PRÁCTICAS A PARTIR DEL ESTUDIO DE LOS PRINCIPALES TIPOS DE MALWARE RANSOMWARE QUE HAN ATACADO EN ECUADOR A LAS ESTACIONES DE TRABAJO CON SISTEMA OPERATIVO WINDOWS MEDIANTE ANÁLISIS DINÁMICO Y ESTÁTICO

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN "ELECTRÓNICA Y TELECOMUNICACIONES"

ANDRADE VALDEZ JENNYFER ALEXANDRA

jennyferandradev@gmail.com

GALARZA ZURITA GIOVANNY PAUL

giovanny.galarza.ec@outlook.com

DIRECTOR: ING. WILLAMS FERNANDO FLORES CIFUENTES Msc.

Quito, febrero 2019

AVAL

Certifico que el presente trabajo fue desarrollado por Jennyfer Alexandra Andrade Valdez y Giovanny Paúl Galarza Zurita, bajo mi supervisión.

ING. WILLAMS FERNANDO FLORES CIFUENTES Msc
DIRECTOR DEL TRABAJO DE TITULACIÓN

DECLARACIÓN DE AUTORÍA

Nosotros, Jennyfer Alexandra Andrade Valdez y Giovanny Paúl Galarza Zurita, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

JENNYFER ALEXANDRA ANDRADE VALDEZ	GIOVANNY PAÚL GALARZA ZURITA

DEDICATORIA

A mi mami, porque pase lo que pase siempre me ha apoyado y ha estado junto a mí, te amo mamá. A mi papá y mi ñaño que son los hombres de mi vida. A Denisse por ser esa fuerza que me impulsa a ser mejor.

Jennyfer

DEDICATORIA

A Alma, los cíen paisajes que visitamos flotando y el lugar de la puerta roja con tapete de “Bienvenido”.

A Carmita, personificación entera del amor desprendido.

Giovanny

AGRADECIMIENTO

Agradezco a mi familia, que siempre han estado para mí y han aguantado esta década en la Universidad.

Agradezco a mis amigos que estuvieron en los momentos buenos y malos. Gracias también por presionarme y preguntar a cada rato como me va con la tesis.

A Gio Jazz, gracias por aguantar y adaptarte a mis horarios, un súper buen compañero de tesis.

Finalmente agradezco al ingeniero Fernando Flores que desde un inicio tuvo la apertura para guiarnos en la realización de este trabajo.

… Por fin se acabó …

Jennyfer

AGRADECIMIENTO

A mis padres, por todo el apoyo durante estos años.
A mis hermanos y sobrinos, por las sonrisas.
A Javier, Miguel, Roberto y Romina por haberme brindado un oído cuando lo necesité.
A Richard por haberme tendido su mano entonces, y aconsejado que tendiera la mía a quien la necesitara.

Giovanny

ÍNDICE DE CONTENIDO

RESUMEN

Durante los años 2015, 2016, 2017, el número de familias y variantes de Ransomware ha crecido vertiginosamente, infectando a usuarios finales, como también a grandes corporaciones a nivel del mundial.

Mediante la investigación se elaboró un ranking de amenazas Ransomware en Ecuador a equipos Windows, del cual se tomaron 4 cuatro muestras que son utilizadas para el análisis. Adicionalmente, se describe la metodología propuesta basada en recomendaciones de SANS y MARE y las herramientas utilizadas para el análisis.

Se realizó el análisis estático y dinámico de las muestras seleccionadas y se exponen los resultados obtenidos centre los cuales están el uso de librerías, funciones y recursos por parte del malware, así como también las conexiones que realizan a nivel de red para la propagación de la infección. Además, se realiza la comparación del estudio de una de las muestras con un trabajo previo.

Finalmente se entrega un compendio de recomendaciones de buenas prácticas informáticas, con el fin de evitar infecciones de malware o mitigar el impacto que estas provocan.

PALABRAS CLAVE: Ransomware, Análisis estático, Análisis dinámico, WannaCry, Spora, CTBLocker, Locky, Seguridad informática.

ABSTRACT

During the years 2015, 2016, 2017 there was an accelerated growth of families and variants of Ransomware, which infected end users and big corporations all around the world.

Through research, a ranking of Ransomware threats in Ecuador was developed for Windows computers, of which, four samples were taken to be used for the analysis. Additionally, the proposed methodology based on SANS and MARE recommendations and the tools for the analysis are described.

The static and Dynamic Analysis of the chosen samples is performed and the obtained results are exposed, which are the libraries' use of libraries, functions and resources, as well as the connections they make at the network level for the propagation of the infection. Also, a comparison of the study of one of the samples to one of a previous project is made.

Finally, a best computer practice recommendations compendium is delivered with the purpose of preventing malware infection and mitigate the impact caused by them.

KEYWORDS: Ransomware, Static Analysis, Dynamic Analysis, WannaCry, Spora, CTBLocker, Locky, Computer Security, Virus, Malware.

1. INTRODUCCIÓN

Actualmente las tecnologías de seguridad informática avanzan rápidamente; sin embargo, los ataques informáticos son cada vez más robustos, lo que enfrenta a los profesionales en tecnología informática a ataques de seguridad de la información más complejos.

Durante el año 2015 según notificaciones de Kaspersky Lab se registraron:

- 1'996.324 intentos de infección de malware con el objetivo de acceder a cuentas bancarias en línea.
- 753.684 infecciones a terminales de usuarios.
- 179.209 intentos de encriptación de datos [1].

Para el año 2016 fueron reconocidas más de 54.000 modificaciones de Ransomware y 62 nuevas familias de malware, arrojando finalmente un registro de 1'445.434 ataques de encriptación a usuarios finales donde el 22.6% fueron dirigidos a usuarios corporativos [2].

En el año 2017 existió un aumento en las modificaciones de Ransomware detectándose así 960.000 de estas y 38 nuevas familias de malware [3].

De acuerdo al reporte de FireEye "Looking Agead Cybersecurity 2018" se espera que en los siguientes años se incremente el número de ataques de Ramsomware debido a los problemas a nivel de administración y actualización de los equipos.

Ecuador se alinea a las tendencias globales, durante el 2017 del total de infecciones se calcula que el 45.6% fueron infecciones de malware. Estas cifras nos ubican en el Top 5 de países con mayor número de infecciones en la región [4].

Este significativo incremento acelerado de malware ejecutado por cibercriminales coloca a empresas y usuarios finales vulnerables a cualquier ataque. Entonces:

Si nos encontramos en el ranking de países latinoamericanos con mayor número de ataques por usuarios conectados, es válido cuestionarse si las empresas y usuarios finales están realmente protegidos.

También es válido analizar a profundidad el concepto de Ransomware y su aparición en el año 1989. Durante ese año se distribuyó el primer tipo de programa capaz de encriptar la información del usuario y pedir una recompensa a cambio de desencriptarla.

Para la distribución de este programa llamado PC Cyborg (también conocido como AIDS Information Trojan) se enviaron a través del servicio postal diskettes que lo contenían, haciendo pasar este diskette como información pertinente al virus del SIDA.

A partir de entonces y durante la primera mitad de la década de los 90s, empezaron a desarrollarse criptovirus, que no exigían un rescate por los archivos del usuario, sino que simplemente los encriptaban. Ejemplos de estos son los malware “One-Half” y “KOH” [5].

Durante los primeros meses del año 2017 se difundieron noticias de ataques de Ransomware en los cuales la información era encriptada en archivos de extensión “.WCRY”. Esta infección se extendió por varios lugares del mundo, principalmente en Rusia, Ucrania, España y el Reino Unido, donde 16 instituciones médicas se vieron afectadas.

La familia de ataques procedentes de este exploit (programa o código que "explota" una vulnerabilidad del sistema o parte de él para aprovechar esta deficiencia en beneficio del creador del mismo) [6], se conoce como “Wannacry”.

Para conseguir sus propósitos Wannacry se aprovecha de un Exploit de Windows conocido como “Eternalblue”. A partir de este punto se procede a la encriptación, que se realiza mediante una herramienta de cifrado de descargada.

Debido al incremento continuo de ataques de malware en la actualidad, este trabajo de titulación se apoya en el análisis de tipos de encriptación de 4 diferentes variantes de malware Ransomware con mayor número de incidencias que afectan a las estaciones de trabajo en el Ecuador, para este análisis se incluirán vectores de infección, su modo de ejecución y adicionalmente se describirán las recomendaciones necesarias para evitar la propagación del malware y futuros ataques.

Este trabajo propuesto también espera reportar las amenazas en las estaciones de trabajo mediante un manual de buenas prácticas, que mejore la cultura de seguridad informática,

con el fin de evitar el riesgo de la información de los usuarios tanto a nivel personal como empresarial.

1.1 Objetivos

El objetivo general de este proyecto es elaborar recomendaciones de buenas prácticas a partir del estudio de los principales tipos de malware Ransomware que atacan en Ecuador a las estaciones de trabajo con sistema operativo Windows mediante análisis dinámico y estático

Los objetivos específicos de este Proyecto Integrador son:

- Identificar los tipos de malware Ransomware más difundidos en Ecuador para estaciones de trabajo con sistema operativo Windows, mediante estudio de reportes de seguridad.
- Implementar un ambiente aislado y seguro para el análisis de malware siguiendo los lineamientos de SANS Institute.
- Realizar el análisis dinámico para conocer el impacto que puede llegar a tener el malware.
- Determinar mediante el análisis estático de las propiedades el funcionamiento del malware.
- Elaborar un manual de buenas prácticas para usuarios finales esperando disminuir la vulnerabilidad en futuros ataques.

1.2 Alcance

Se identificarán los 10 tipos de malware Ransomware en el Ecuador con mayor presencia en los últimos años, los datos se obtendrán mediante una investigación basada en reportes de herramientas de seguridad informática.

De este listado previo se obtendrán 3 muestras de malware, puesto que para cada muestra que es necesario preparar un ambiente de pruebas individual y realizar un análisis de equipo previo a la infección, se seleccionarán las muestras de malware tomando en cuenta que no hayan sido analizadas en trabajos previos y serán utilizadas en el desarrollo del trabajo.

Adicional a las 3 muestras previamente indicadas, se realizará el análisis de una cuarta muestra que ya haya sido probada en trabajos previos con el objetivo de contrastar resultados.

Se implementará una máquina virtual con sistema operativo Windows versión 7 en un ambiente aislado y seguro siguiendo los lineamientos de SANS Institute [7] evitando así que una infección se propague en la red.

Previo a la infección se procederá a documentar el estado inicial del sistema con el fin de contrastar la situación después de realizar la infección. En el equipo Windows se ejecutarán las muestras de malware con la finalidad de conocer el comportamiento de las mismas.

Se realizará un análisis dinámico completamente automatizado mediante herramientas de libre acceso (Regshot, CwSandbox, Wireshark, ThreathExpert, Buster Sandbox Analyzer) que ayudarán a conocer el impacto que tendría el malware en caso de haberse ejecutado sobre un sistema operativo compatible.

Este trabajo pretende determinar mediante el análisis estático de las propiedades el funcionamiento del malware y ciertas propiedades del mismo obteniendo mayor información sobre las muestras.

Al final del análisis correspondiente se realizará la generación de un manual de buenas prácticas a partir del análisis de 3 diferentes variantes de malware Ransomware para equipos Windows con la finalidad de proteger la información que se encuentra en los equipos del usuario final, evitando presentar vulnerabilidades frente a ataques de malware y futuras infecciones.

Dicho manual contendrá sugerencias de protección en herramientas Endpoint tanto gratuitas como de pago y lineamientos de uso adecuado del equipo en cuanto a seguridad informática.

1.3 Marco Teórico

Tipos de Ataques

Existen varias técnicas de ataques independientemente del objetivo del mismo. Entre las más conocidas y comunes se puede tener:

- **Denegación de servicio:** es normalmente conocido como DoS que son las siglas en inglés Denial of Service, la cual consiste en buscar una vulnerabilidad en el sistema informático. Generalmente se lo realiza sobrecargando los recursos del servicio o provocando la pérdida de conexión de la red por un alto consumo del ancho de banda [8].

- **Ingeniería social:** son técnicas diseñadas para que usuarios finales por falta de conocimientos, den accesos a sus datos confidenciales provocando una infección sus equipos o abriendo enlaces a sitios que se encuentran infectados o que no son legítimos [9].

- **Suplantación:** consiste en trasplantar la identidad de un usuario legítimo capturando las credenciales originales del inicio de sesión a un portal o servicio [10].

- **Exploits:** implica buscar un vacío en la seguridad del software o sistema operativo generado, que no fue detectado durante el desarrollo del mismo.

- **Ataques de datos:** es una secuencia de comandos que permiten al intruso depositar un código malicioso.

- **Debilidades de infraestructura:** consiste en aprovechar que el administrador de la red no ha realizado un despliegue de parches de seguridad de sus equipos o no se han analizado los servicios y protocolos que se encuentran permitidos, con lo cual el atacante puede ingresar a la red de la víctima.

Definición de Malware:

La palabra malware es la abreviatura en inglés de “malicious software” que se traduce al español como software malicioso [11].

Este software está diseñado para infectar un equipo apoderándose del control o generando daños del mismo, por lo general sin que el usuario conozca de la infección.

Por lo general un software es considerado malware dependiendo de las intenciones del desarrollador y no de las características reales. Actualmente es creado para obtener beneficios mediante publicidad, robo de información, difusión de correo no deseado o para la extorsión de dinero.

Clasificación de Malware:

El malware puede clasificarse de varias formas, por lo que se han creado 3 grupos para entender de una mejor manera los tipos de malware que existen.

- **Por su forma de propagación:**

Los tipos de malware dependiendo de la manera de propagación se puede dividir en dos categorías: los que necesitan un programa host y los que son independientes.

Los primeros, son fragmentos de programas que no pueden existir independientemente de algún programa de aplicación real, utilidad o sistema, por ejemplo los Virus.

El malware independiente como su nombre lo indica no depende de un sistema operativo y su principal representante son los Gusanos [12].

  - **Virus:** Es un programa informático que puede infectar a otros sistemas modificándolos, depende de otros programas y generalmente requiere la interacción humana para propagarse [13].

    Un virus informático puede ser desarrollado con las mismas herramientas que cualquier otro software, con el fin de que sea rápido y pequeño.

Dentro de los virus también existen varios tipos, entre los que se tienen:

- ✓ Del sector de arranque
- ✓ Infecciones de archivos
- ✓ Virus macro y scripts

- **Gusanos:** Es un programa informático que no se reproduce de manera parasita, es decir, que se puede ejecutar de manera independiente. Normalmente un gusano toma el control de un equipo y lo usa como punto de partida para controlar otros sistemas vulnerables [12].

- **Por su capacidad de ocultarse:**

La infección pasa desapercibida dentro de un sistema, sin que el usuario final esté consciente de la infección.

- **Troyanos:** Es un programa que parece tener una función útil, pero que en realidad tiene una función oculta la cual es maliciosa evadiendo los mecanismos de seguridad del equipo [12].

- **Backdoors:** También conocido como puerta de trampa, es un programa con una entrada oculta que quien conoce de su existencia puede ingresar al equipo infectado sin necesidad de pasar por los protocolos de seguridad.

- **Rootkits:** Es un software malicioso diseñado para infectar un equipo y permitiendo que el atacante instale un conjunto de herramientas otorgando acceso remoto al equipo.

Por lo general se oculta en el sistema operativo y está diseñado para evadir la detección de antimalware de aplicaciones. Puede contener herramientas que registran las pulsaciones de teclado, robo de contraseñas o módulos que roban información de tarjetas de crédito o acceso a bancos en línea [14].

- **Por su manera de lucrar**

Este tipo de malware tiene fines lucrativos, con el fin de generar ingresos económicos a los atacantes.

- **Bot:** Viene de la palabra robot y se caracteriza por ser un ataque automatizado. Normalmente el objetivo de un bot es infectar al equipo para conectarse a un servidor central, en el que el bot ejecuta instrucciones para realizar un ataque globalizado contra un objetivo específico.

- **Adware:** Genera anuncios automáticamente en un servicio o aplicación presentada al usuario. Puede obtener dos tipos de ingresos: uno es para la visualización del anuncio y otro cuando el usuario da clic en el anuncio presentado [15].

- **Spyware:** Su objetivo es recopilar información sin el conocimiento de la persona u organización atacada, con el fin de vender la información obtenida a un tercero [15].

Ransomware:

Ransomware es un tipo de malware con el fin de extorsionar víctimas, exigiendo pagos para deshacer los cambios realizados en los archivos infectados. Los cambios pueden ser:

- Cifrar la información de la víctima evitando que pueda acceder a su información.
- Bloquear el acceso al equipo de la víctima [16].

El Ransomware se caracteriza por ser totalmente evidente por su víctima.

- **Vías de infección de Ransomware:**

Las maneras más comunes en que se instala un Ransomware son:

- ✓ Por correos electrónicos falsos conocidos como phishing.
- ✓ Al ingresar a sitios web con un programa malicioso.

Una vez infectado el equipo ya sea cifrando la información o evitando que acceda al sistema, se suele mostrar un mensaje de rescate solicitando el pago para recuperar la información [16].

- **Estadísticas de Ransomware en la actualidad:**

Desde sus primeras apariciones hasta los registros del año 2017, los tipos de Ransomware han aumentado exponencialmente en el número de familias y variantes. Como se indica en la Figura 1.1.

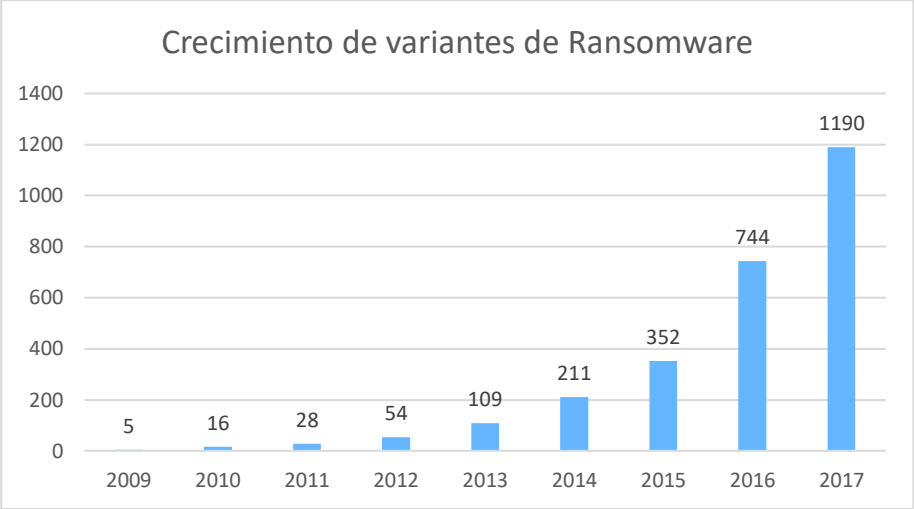

Figura 1.1. Crecimiento de Ransomware de los últimos 8 años *[17]*.

- **Estadísticas de Ransomware del año 2015:**

En el año 2015 se pudo observar que Ransomware empezó a ser un servicio alojado en redes anónimas y a utilizar de monedas virtuales o criptomonedas, para solicitar el rescate de la información sin dar a conocer la identidad del atacante [18]. Como se indica en la Figura 1.2.

Figura 1.2. Familia de Ransomware del año 2015. *[18]*

Durante el año 2015 las variantes más conocidas fueron:

- CryptoWall 3.
- CTB-Locker.
- CryptoLocker.

Mientras que los archivos más infectados fueron:

- Microsoft Office.
- Adobe PDF.
- Archivos gráficos.

Kaspersky pudo registrar el porcentaje de ataques detectados de Trojan-Ransom a nivel mundial como se puede observar en la Figura 1.3.

	Country*	% of users attacked by Trojan-Ransom**
1	Kazakhstan	5,47
2	Ukraine	3,75
3	Russian Federation	3,72
4	Netherlands	1,26
5	Belgium	1,08
6	Belarus	0,94
7	Kyrgyzstan	0,76
8	Uzbekistan	0,69
9	Tajikistan	0,69
10	Italy	0,57

Figura 1.3. Países más atacados de Trojan-Ransomware del año 2015 *[1]*.

- **Estadísticas de Ransomware del año 2016**

En el año 2016 el número de usuarios atacados por Ransomware fue de 1'445.434 y se detectaron 54.000 modificaciones con un total de 62 nuevas familias de este malware [2]. Como se indica en la Figura 1.4.

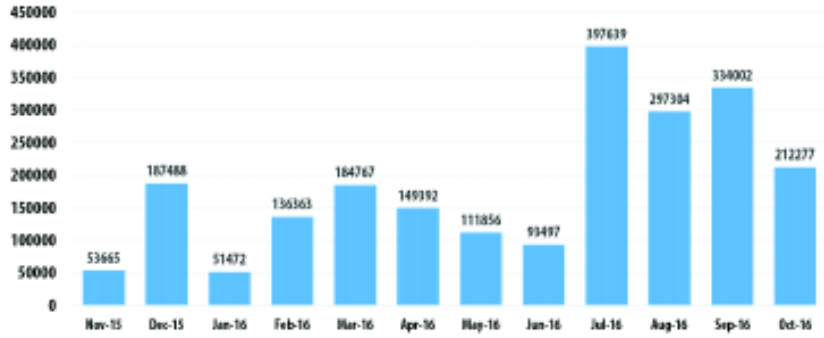

Figura 1.4. Usuarios atacados por Ransomware del año 2016 *[2]*.

Estos valores reflejan una parte de la verdadera cantidad de ataques que existieron a nivel mundial, puesto que reflejan los resultados de detecciones de heurísticas y basadas en firmas por parte de Kaspersky Security Network.

El crecimiento de familias y de variantes de Ransomware en el año 2016 fue casi del 100% y muchas de estas variantes fueron creadas por desarrolladores amateurs por lo que fue fácil detectar los ataques. Como se observa en la Figura 1.5.

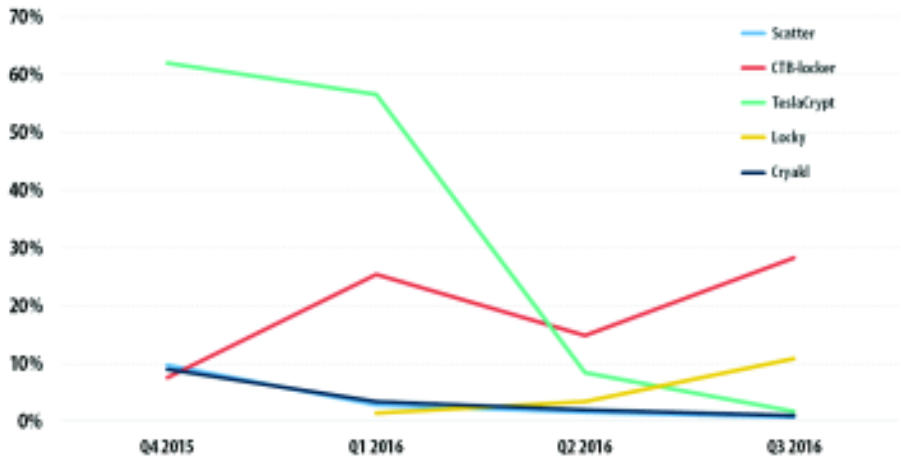

Figura 1.5. Familias de Ransomware con mayor presencia en el año 2016 *[2]*.

- **Estadísticas de Ransomware del año 2017**

Durante el año 2017 se lograron detectar más de 96.000 modificaciones de Ransomware, descubriéndose así 38 nuevas familias.

Este fue el año con mayor evolución de Ransomware siendo WannaCry su mayor representante acumulando un total de 700.000 víctimas [19].

WannaCry se presentó el 12 de mayo de 2017, siendo uno de los ataques cibernéticos más grandes de la década, al infectar tanto a usuarios finales, como a grandes corporaciones como Telefónica en España, Renault en Francia, varios hospitales en Reino Unido, entre otros.

Su método de infección aprovecha un vacío de seguridad en Windows, que fue parchado en marzo del mismo año, pero que varios administradores de red no habían utilizado [19].

En cuanto a América Latina en el año 2017 la familia de Ransomware con más detecciones fue WannaCry con un 23% de presencia en ataques, como se puede observar en la Figura 1.6 [19].

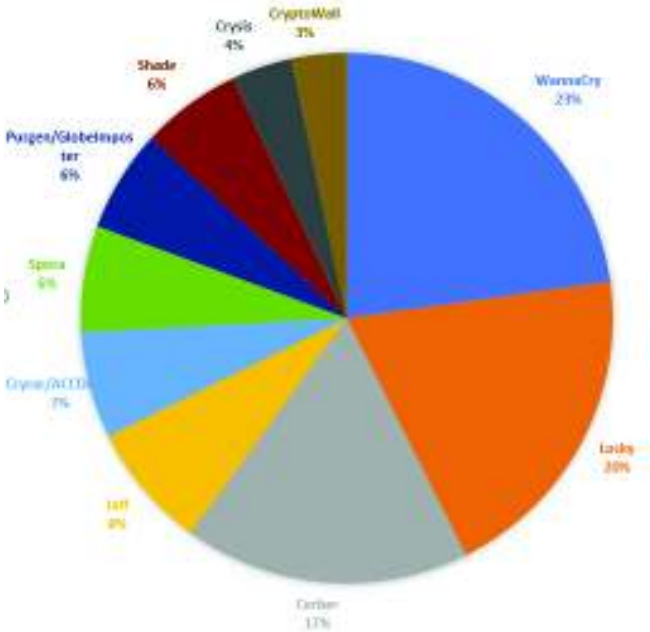

Figura 1.6. Variantes de Ransomware 2017 *[19].*

- **Clasificación de Ransomware**

En este trabajo se han definido 3 tipos de Ransomware:

- **Locksreen:**

Este tipo de Ransomware impide el acceso a los recursos del equipo, bloqueando la interfaz gráfica del equipo infectado y solicitando un pago para desbloquear el equipo.

Este tipo de malware no encripta la información y deja al equipo con capacidades limitadas para que la víctima se pueda comunicar únicamente con el atacante y realizar el pago solicitado.

Puesto que este Ransomware generalmente se puede eliminar entrando en modo seguro al equipo, tiende a utilizar ingeniería social para presionar a la víctima a realizar el pago, disfrazándose de autoridades policiales o regulatorias.

- **Cryptolockers:**

Cryptolockers están diseñados para cifrar la información del equipo e inhabilitar el acceso a los documentos infectados a menos que se obtenga una contraseña de descifrado, la cual el atacante provee una vez realizado el pago para liberar la información.

Este ataque se considera exitoso cuando la víctima considera como valiosa la información encriptada y no tiene un respaldo de la misma.

Una vez instalado el malware este busca los archivos, evitando ser detectado, hasta encriptar la mayor cantidad de ellos. Puesto que el objetivo no es bloquear el acceso al equipo infectado, la víctima puede ingresar a las funcionalidades de la computadora.

- **Ransomware para dispositivos móviles:**

Para los dispositivos móviles un equipo se puede infectar de Ransomware al realizar la instalación de una aplicación o de la misma manera que en los equipos PC a través de phishing o accesos a páginas web infectadas.

En el año 2017 se descubrieron 544.107 paquetes infectados de Ransomware para móviles, duplicando la cantidad que se detectó en el año 2016. El mayor representante de este es Trojan-Ransom.AndroidOS.Congur [20]. Como se observa en la Figura 1.7.

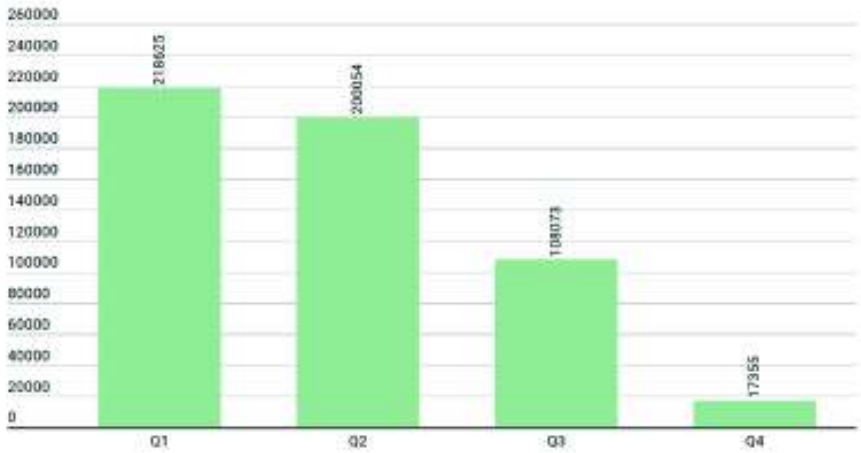

Figura 1.7. Número de paquetes de instalación de Trojan-Ransomware móvil 2017 *[20]*.

Patrones de Infección:

Cuando un equipo se infecta con Malware tiende a presentar comportamientos y signos ajenos a su funcionamiento normal.

Los fabricantes de software advierten de este tipo señales y que la presencia de uno o varios de los mismos pueden ser el resultado de una infección de software malicioso.

A continuación, se enlista una recopilación de comportamientos sospechosos de sistemas computacionales: [21]

- Incremento en el tiempo de respuesta del sistema.
- Congelamiento del sistema o reinicio inesperado al procesar instrucciones.
- Mensajes de error del sistema anunciando archivos faltantes o corruptos.
- Inaccesibilidad a programas de configuración del sistema operativo.
- El funcionamiento del producto de seguridad antivirus ha sido deshabilitado, o se interrumpe sin la interacción directa del usuario.
- Comportamiento inusual durante la navegación de internet, tales como aparecimiento de ventanas emergentes con contenido publicitario recurrente, redireccionamiento, incapacidad de acceder a sitios determinados, barras de búsqueda o configuración nuevas y cambio de la página de inicio del navegador.
- Perdida de archivos y documentos.
- Alto uso espacio en disco

- Al iniciar el computador, otros equipos en la misma red experimentan una desaceleración en el acceso a internet.
- Aparecimiento de nuevos íconos en el escritorio [22].

Es necesario recalcar que algunos de estos síntomas no están ligados con la presencia de malware en el sistema, ya que pueden ser el resultado de otro tipo de desperfectos existentes con el software o hardware.

Sin embargo, la presencia conjunta de estos indicadores son señales que pueden apuntar a la presencia de software malicioso [23].

Metodologías de análisis de malware:

El análisis de Malware puede ser definido como el arte de estudiar minuciosamente el software malicioso con el objetivo de comprender sus componentes y su funcionamiento.

El análisis de Malware proporciona las herramientas necesarias para identificar, neutralizar y/o eliminar dicho programa malicioso. El análisis de Malware es particularmente útil para casos pocos conocidos de Malware, ya que existen casos en que sus códigos son desarrollados para
una operación en específico, y las claves para identificar su comportamiento dentro del sistema no han sido analizadas por herramientas antimalware convencionales [24].

El análisis de Malware consiste en la aplicación de un conjunto de técnicas útiles para identificar los recursos utilizados, cambios en el sistema operativo, intentos de comunicación por red, infección de archivos o equipos vecinos y demás comportamientos destructivos que pudieran ser causados por software malicioso.

Este conjunto de técnicas se agrupan en dos categorías según su forma de interactuar con el programa sospechoso de malignidad y toma en cuenta la ejecución necesaria del malware, de esta manera se puede realizar análisis estáticos o análisis dinámicos. Además, dentro de estas categorías se puede distinguir el nivel de análisis necesario, que se divide en básico o avanzado [25]. Como se observa en la Figura 1.8.

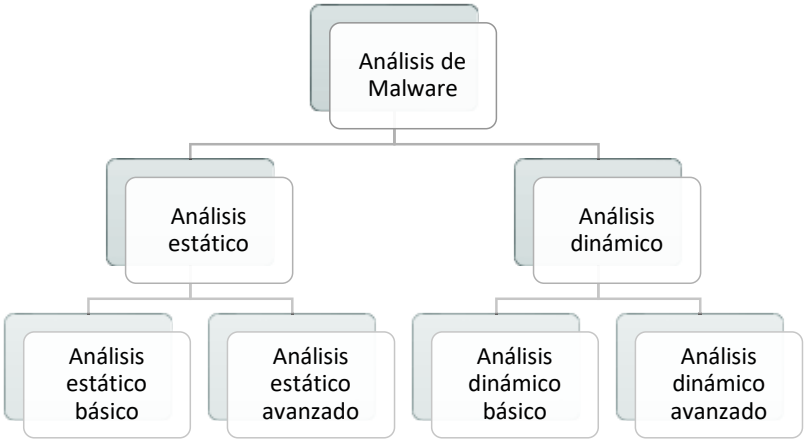

Figura 1.8. Metodologías de análisis de Malware.

A continuación, se dará descripciones a grandes rasgos de estos métodos.

Las descripciones a mayor detalle de los mismos y los tipos de técnicas que se incluyen en cada una de estas etapas se explicarán en el capítulo 2 [26].

- **Ambiente virtual para el análisis de malware**

Para realizar las prácticas correspondientes a un análisis de malware es necesario instalar un “laboratorio” virtual.

Dicho laboratorio virtual debe ser seguro tanto para el computador donde se realizará el análisis, como para otros equipos que pueden afectarse en la misma red de computadoras.

El laboratorio virtual para análisis de malware puede realizarse sobre máquinas físicas o virtuales. Los laboratorios montados en máquinas físicas permiten una ejecución más natural en respuesta a malware capaces de detectar ambientes virtuales. Sin embargo, el malware puede ser más difícil de eliminar de las máquinas físicas, lo cual resulta inconveniente si se necesitan analizar distintas muestras a partir de las mismas condiciones iniciales. Es por esto que a pesar de que el análisis puede ser realizado en máquinas físicas, la práctica común es hacerlo en máquinas virtuales.

Para crear un laboratorio virtual para el análisis de malware primero se debe elegir la herramienta de virtualización, que gestionará la creación, funcionamiento e interacción de las máquinas virtuales.

Las herramientas más comunes para estos fines son VMWare y VirtualBox. Al momento de elegir qué ambiente utilizar se debe tomar en cuenta las prestaciones que dichos ambientes permiten (toma de capturas de estado de las máquinas virtuales, ambiente de red independiente, entre otras).

Es importante tener presente las tecnologías de virtualización existentes. La virtualización nativa permite al sistema operativo huésped ejecutar código utilizando de manera directa los recursos de Hardware.

Para esto es necesario que ciertos recursos se dediquen a esta máquina de manera exclusiva. Por su lado la paravirtualización supone el uso de una interfaz similar a una API que permite al sistema operativo huésped interactuar con el hardware de una manera indirecta.

De acuerdo a las fuentes la paravirtualización permite una mejora en el rendimiento del sistema operativo virtualizado, sin embargo, supone un problema adicional, ya que pueden virtualizarse únicamente sistemas operativos acordes para interactuar con la interfaz anteriormente mencionada (Interfaz Binaria de Aplicaciones), lo cual limita la virtualización de sistemas operativos comerciales para su virtualización.

Es recomendable que la herramienta de virtualización permita un uso inteligente del espacio de almacenamiento. Los malware presentan en su mayoría actividad de red, cuyo análisis es muy útil para el análisis del mismo. La opción recomendable es crear una red de acceso local virtual, accesible únicamente para las máquinas virtuales pertenecientes al laboratorio de análisis de malware, pero que no conecte con otras máquinas físicas.

Es recomendable un laboratorio virtual en el que dos computadores virtuales estén conectados entre sí mediante una LAN virtual, pero que no estén conectados a la máquina física ni a la red. De esta manera una de las máquinas virtuales actuará como “víctima” mientras que la otra proveerá a esta un conjunto de servicios (HTTP, DNS, entre otros), los cuales serán necesarios para que el malware interactúe de manera normal con su ambiente.

- **Análisis estático básico:**

Debido a las formas en que el malware se presenta en los equipos víctima, no es común tener acceso al código fuente del malware. Debido a esto el análisis estático básico se realiza sin leer los comandos y comprende una serie de pasos que incluyen [24]:

- Uso de herramientas antimalware: De primera mano este proceso supone una búsqueda para verificar si la muestra corresponde a un malware que ya ha sido identificado por un software antimalware o herramientas online dedicadas a este propósito.

- Hashing: Supone ingresar la muestra sospechosa de malignidad a una herramienta criptográfica que, a su salida, ofrece una secuencia de caracteres mediante la cual se puede obtener más información sobre la muestra mediante una búsqueda online.

- Búsqueda de Strings: Este proceso permite encontrar series de caracteres almacenados dentro de la muestra, los cuales pueden dar indicios sobre la funcionalidad del programa.

- Análisis de ofuscación o empaquetamiento: Permite saber si la información de la muestra ha sido empaquetada o cifrada mediante algún algoritmo

- Búsqueda de funciones y librerías citadas.

- Búsqueda de ejecutables portables.

Ninguno de estos procesos implica la ejecución del software malicioso, pero si la extracción de información contenida en el mismo mediante el uso de herramientas diseñadas para tales fines.

- **Análisis dinámico básico:**

El análisis dinámico básico incluye técnicas que necesitan ejecutar el código malicioso para poder examinar su comportamiento y efectos. Este tipo de análisis no requiere ahondar en la programación utilizada para la creación del software malicioso y se centra en el uso de

técnicas, utilidades y herramientas capaces de detectar acciones realizadas durante la ejecución. Dichas técnicas incluyen:

- **Sandboxes:**

Son herramientas que incluyen múltiples funcionalidades que permiten al malware actuar en un ambiente seguro virtualizado. Estas herramientas entregan a su salida información de actividad de archivos, red y registros.

- **Ejecución del Malware:**

Una manera en que se puede entender el Malware es ejecutándolo en un ambiente controlado, y de esta manera observar directamente los efectos causados por este. Durante la ejecución del malware deben realizarse:

- ✓ Monitoreo de procesos
- ✓ Comparación de Strings
- ✓ Comparación de capturas de registro.
- ✓ Monitoreo de actividad de registro.

Este tipo de análisis puede realizarse mediante plataformas mayoritariamente automatizadas llamadas Sandboxes o mediante la ejecución manual del Malware y el análisis de los efectos causados en el sistema con lo cual se podrán conocer a más profundidad los detalles sobre el funcionamiento del malware.

- **Análisis Estático Avanzado:**

El análisis estático avanzado puede resultar de un nivel elevado de complejidad, ya que para realizarlo es necesario utilizar técnicas de desensambaje e ingeniería inversa del software a analizarse, para acceder a las instrucciones del programa y poder analizarlas línea a línea.

Esta técnica puede revelar exactamente lo que el Software malicioso puede hacer, ya que permite analizarlo en su totalidad; sin embargo, este tipo de análisis requiere conocimientos avanzados de programación, así como también del funcionamiento del sistema operativo Windows, además de un trabajo arduo de análisis de código en lenguaje de bajo nivel.

El análisis estático avanzado parte de un archivo binario en el que obtiene a la salida las instrucciones en lenguaje ensamblador; es decir, instrucciones de bajo nivel relacionadas con el computador y su arquitectura.

Este proceso es conocido como desensamblaje y se ejecuta a través de software especializado para estos fines llamado densensamblador.

Es importante señalar que existen distintos dialectos de código ensamblador dependiendo del tipo de arquitectura del sistema en cuestión.

La arquitectura más utilizada hasta el momento de elaboración del presente documento es X86, desarrollada por Intel en el año de 1978. La versión de 64 bits basada en X86, llamada X86-64 (x64 o AMD64) y desarrollada por AMD es retrocompatible con el set de instrucciones en ensamblador existente inicialmente en los procesadores X86 de 32 bits.

Por los motivos expuestos, el conocimiento de lenguaje ensamblador para procesadores de arquitectura X86 es una herramienta útil para el proceso de análisis de malware avanzado [27].

- **Análisis dinámico avanzado.**

Este tipo de análisis se basa en utilizar debugging para conocer el estado del software mientras este es utilizado.

El debbuging o depuración es un proceso mediante el cual se analiza la ejecución de un programa instrucción a instrucción, observando los pormenores de los efectos del mismo. Este tipo de análisis se ejecuta usualmente en conjunto con el análisis estático avanzado, ya que un debugger permite observar la manera en que las instrucciones del código malicioso interactúan con las funciones, registros y espacios de memoria del sistema.

Es importante hacer mención de que los debuggers utilizados para este proceso trabajarán con el código a nivel de ensamblador y no de código fuente (el cuál no se posee).

Tanto para el análisis estático avanzado como para el dinámico es importante tomar en cuenta las funciones de sistema operativo a las que el software malicioso hace llamado. A

más de ofrecer una guía al funcionamiento del malware, dichas funciones permiten conocer el modo en que el programa analizado se encuentra ejecutándose.

El modo de usuario es en el que se ejecutan la mayoría de programas. En este modo el programa no tiene acceso directo para manipular el hardware, sino que todos sus llamados deben hacerse a través del Windows API. Por otro lado, el modo de Kernel (que es en el que el sistema operativo y los drivers del hardware se ejecutan) puede interactuar con el hardware directamente, así como también a la totalidad de espacios de memoria disponibles.

Si el Malware se ejecuta en modo de Kernel no solo es más peligroso por la cantidad de acciones que puede realizar (algunas de las cuales se verían restringidas para un código que se ejecute en modo de usuario) sino que puede ser más difícil de detectarse por un software antimalware que no realice monitoreo a nivel de Kernel.

Herramientas Antivirus:

Las herramientas de antivirus utilizan distintas maneras de identificar comportamiento malicioso. Una vez que un software malicioso ha penetrado en el sistema o red víctima, dicho malware debería ser neutralizado por un sistema antimalware, el mismo que deberá procurar realizar las siguientes acciones con la finalidad de mitigar los efectos causados por programas malignos [12].

- Detección: este mecanismo determina y notifica que ha ocurrido una infección a su vez que identifica dentro del sistema los archivos que han sido infectados.

- Identificación: El presente mecanismo se ejecuta posterior a que la presencia de malware ha sido detectada y consiste en advertir qué tipo de malware es el que se encuentra presente en el equipo identificándolo con precisión.

- Eliminación: Dicho mecanismo se lleva a cabo una vez que el malware ha sido detectado e identificado. La eliminación es el proceso mediante el cual se intenta suprimir todas las muestras de software dañino dentro de los sistemas infectados de tal suerte que la infección no pueda propagarse.

Clasificación de Sistemas de Detección de Malware:

Las técnicas de detección de virus pueden tener un comportamiento reactivo si basa su detección en comparación con características de Software malicioso previamente conocido; o proactivo si es capaz de identificar software malicioso con el que no ha interactuado antes.

En la Figura 1.9. se indica como los sistemas de detección de malware pueden actuar tanto en los equipos terminales (host) como también en los sistemas de seguridad periféricos, tales como Firewalls y Sistemas de Detección de intrusos (IDS).

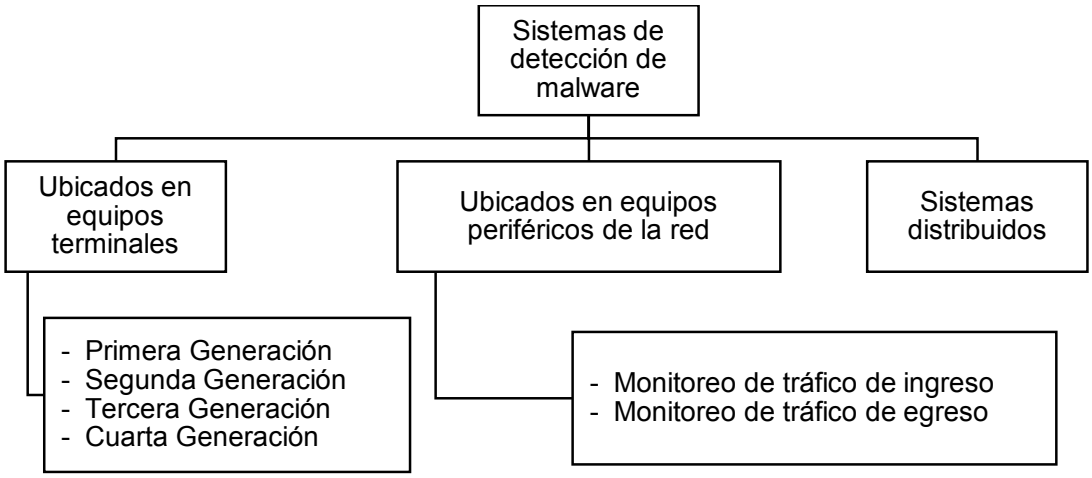

Figura 1.9. Clasificación de Sistemas de Detección de Malware.

- **Buscadores de Malware ubicados en hosts:**

El uso de software antivirus en cada equipo terminal permite un análisis en detalle de los archivos que pudieran estar infectados.

Dado que a lo largo del tiempo la complejidad de los distintos tipos de malware ha ido creciendo, es necesaria la evolución en los productos de detección antivirus.

Dicha carrera tecnológica ha llevado al desarrollo de diferentes técnicas de tratamiento de malware, las mismas que han definido los productos antimalware. Con estas consideraciones, y de acuerdo a Stallings [12], se puede clasificar los antivirus en cuatro generaciones hasta el momento de la publicación del presente documento.

- **Primera Generación:**

En su intento por difundirse en un sistema, los virus buscan no copiarse a sí mismos en un programa que ya ha sido infectado.

Para que esto sea posible, dejan en los archivos víctima un patrón de bits que puedan reconocer, si encuentran dicha marca o "firma" en un programa, lo ignorarán ya que dicho programa ya ha sido infectado. Los motores de búsqueda de virus de primera generación buscan identificar firmas de virus conocidos por estos en los programas existentes en el sistema. Para realizar dicha detección es necesario para los antivirus contar con una base de datos de las firmas conocidas.

Una desventaja evidente de este tipo de mecanismo de detección de virus es su funcionamiento, ya que es necesario que el antivirus se encuentre actualizado permanentemente y que la base de datos del antivirus contenga las firmas de todos los virus conocidos. Esto implica un trabajo arduo al agregar las firmas de los nuevos virus que se descubren y que los programas antivirus estén siempre un paso atrás de los desarrolladores de software malicioso.

- **Segunda Generación:**

Los antivirus de segunda generación dejaron de depender de "firmas" para la detección de virus.

Un tipo de antivirus de segunda generación analiza secciones de código que realizan acciones sospechosas tales como lazos de encriptación, pudiendo ser capaces incluso de romper dicha encriptación.

Este tipo de técnicas se basa en reglas que permiten puntuar sobre 10 qué tan posible es que el programa analizado sea malicioso. Si el archivo en cuestión obtiene una calificación alta, el motor antivirus toma acciones al respecto. Este tipo de técnicas se denominan "análisis heurístico", lo que significa que es un análisis basado en reglas, pero no estricto, sino que permite una suerte de "ensayo y error". Este tipo de sistemas de detección presentan como desventaja casos de falsos positivos, es decir que existen casos en que identifican erróneamente como malicioso a software de código benéfico.
Durante esta generación también se desarrollaron técnicas de verificación de integridad. Dicha técnica se basa en el uso de un algoritmo lógico o matemático para realizar detección de errores. Dicho algoritmo genera una secuencia de bits o símbolos tales que se

correspondan con el contenido del archivo en cuestión. Si el virus modifica el contenido del archivo sin modificar la secuencia calculada se puede verificar que el programa ha sido corrompido. Existen sin embargo softwares maliciosos más avanzados que intentan reemplazar también esta secuencia de bits para evitar la detección. Razón por la cual los motores de búsqueda de malware también han tenido que evolucionar.

- **Tercera Generación**

Los motores de búsqueda de malware de tercera generación se centran en analizar el comportamiento de los programas activos en la memoria más allá de su estructura. La ventaja de estas técnicas es que es más eficiente para analizar las instrucciones del sistema que pueden representar malignidad en lugar de llevar a cabo un análisis de firmas o evaluaciones heurísticas del posible software malicioso.

- **Cuarta Generación**

La cuarta generación de motores de detección de malware incluye técnicas de detección como desencriptación genérica y bloqueo de comportamiento.

La técnica de desencriptación genérica detecta los archivos encriptados y los ejecuta a través de un emulador, el mismo que analiza las instrucciones obtenidas del software sospechoso en lugar de ejecutarlas directamente.

El objetivo de dicho emulador es engañar al software malicioso para que se comporte tal como si estuviera ejecutándose y de esta manera analizar sus acciones y realizar sobre el mismo un análisis basado en firmas una vez que este se ha desencriptado.

Es importante recalcar que al desencriptarse el malware e interpretarse, no hay riesgo de que se ejecuten instrucciones dañinas en el sistema, ya que el malware es interpretado en un ambiente completamente aislado. Por otro lado, una desventaja de este tipo de antimalware es que su análisis del archivo sospechoso causa demoras en la ejecución deseada, y que, al no poder realizar un análisis completo debido al tiempo, puede causar conductas maliciosas que no son detectadas.

En la técnica de bloqueo de comportamiento el sistema de detección de Malware trabaja de manera integrada con el sistema operativo del ordenador, de manera que pueda monitorear las acciones realizadas por los programas y, si intentan ejecutar instrucciones peligrosas, bloquearlos para que no afecten al sistema.

Esta técnica tiene como ventaja su eficiencia ya que se fija únicamente en las instrucciones de sistema que pueden resultar nocivas. Por otra parte, pudiera no ser eficaz, ya que puede ejecutar acciones nocivas de carácter no crítico (tales como renombrar ficheros) sin que el malware sea bloqueado.

- **Sistemas de detección de Malware en dispositivos periféricos de la red.**

Otra posibilidad para realizar control de malware es colocar sistemas de detección ubicados en el Firewall o el Sistema de Detección de intrusos de una red. Este tipo de software antivirus puede funcionar como parte de los servicios de análisis de tráfico o de Proxy.

El software de detección de malware basado en dispositivos periféricos presenta la ventaja analizar el tráfico que transita por la red, bloqueando incluso flujos de información maliciosa. Por otro lado, carece de capacidad de análisis basado en comportamiento, al tener únicamente acceso al contenido de la información mientras se halla en la red.

Este tipo de análisis de malware puede realizarse mediante dos tipos de monitoreo. El monitoreo de ingreso se realiza entre el proveedor de internet y la red empresarial, en dispositivos como routers de bordeo firewalls externos. Por otra parte, el monitoreo de egreso de tráfico puede localizarse tanto en los puntos de salida de las LANs individuales dentro de la empresa como en el borde entre la red y el proveedor de internet.

La búsqueda de virus en sistemas periféricos es de principal utilidad para lidiar con botnets y detectar gusanos debido a la actividad que este tipo de software malicioso realiza sobre la red y el tráfico que genera.

- **Sistemas de detección de virus en sistemas distribuidos.**

Finalmente, la detección antivirus puede realizarse de una manera distribuida entre distintos equipos combinando las acciones de detección en host y en dispositivos periféricos.
La información reunida de dichos sensores es analizada por un sistema centralizado, el mismo que desarrolla acciones heurísticas para neutralizar y/o eliminar el software malicioso.

Un ejemplo de sistema de búsqueda de virus en un ambiente distribuido es el presente en el Digital Immune System de IBM. Este sistema detecta al software malicioso de manera temprana cuando este entra en el sistema distribuido.

El software sospechoso es enviado a un sistema centralizado para su análisis. En respuesta, dicho sistema crea información útil para la detección, neutralización y eliminación del virus, misma que es enviada al equipo remitente original y a todos los equipos de la organización.

- **Valoración de Herramientas Endpoint.**

Para realizar un análisis de las herramientas de antivirus que se utilizarán se ha tomado en cuenta el criterio de NSS Labs y Gartner, que se encargan de analizar distintas marcas de seguridad de la información:

- **NSS Labs:**

NSS Labs, cada año presenta un mapa de valores de seguridad, el cual está dividido por líneas punteadas que representan la relación entre efectividad y costos, para de esta manera determinar 3 niveles.

- **Recomendado:** son los productos que proporcionan un alto nivel de detección y una buena relación calidad-precio [28].
- **Precaución**: son aquellos que ofrecen un valor limitado por el dinero dado [28].
- **Neutral**: los productos que se encuentran en las secciones superior izquierda o inferior derecha pueden ser buenas opciones para organizaciones con requisitos específicos de seguridad o presupuesto [28].

Como se observa en la Figura 1.10.

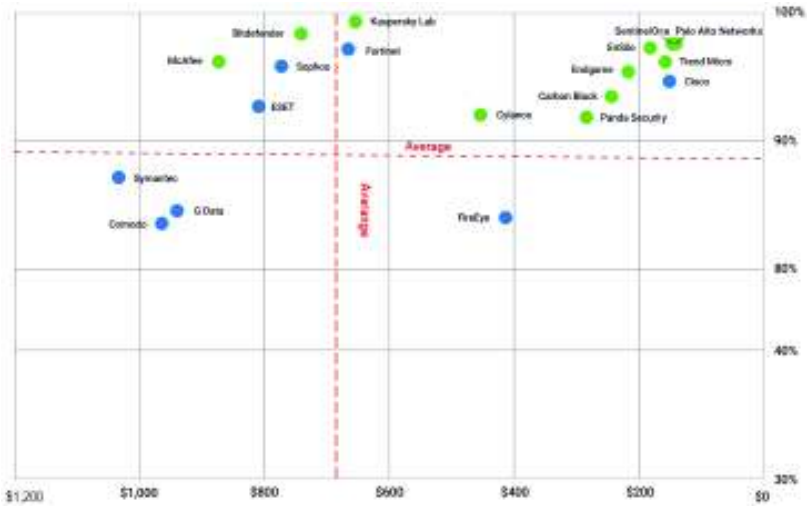

Figura 1.10. NSS Labs de Antivirus 2017 *[29]*.

- **Cuadrante de Gartner:**

El cuadrante mágico de Gartner indica el posicionamiento en el mercado de proveedores de tecnología, el cual está dividido por 2 ejes, el eje X mide el conocimiento de cada marca de cómo aprovechar el momento actual del mercado y de cuantos usuarios finales continúan con dicha marca, mientras que el eje Y mide la capacidad que tiene una marca para ejecutar con éxito su visión del mercado, creando 4 categorías de marcas o proveedores, las cuales son [30]:

- **Líderes**: son las marcas o productos que ejecutan bien su visión y que están bien posicionados para el futuro.
- **Visionarios**: son aquellas marcas que entienden hacia dónde va el mercado, pero aún no ejecutan su visión correctamente.
- **Jugadores de nicho:** se enfocan con éxito en un segmento pequeño o específico del mercado, pero no logran superar a las otras marcas.
- **Desafiantes o retadores:** son aquellos que pueden dominar un segmento grande del mercado, pero no logran comprender hacia dónde va del mercado.

Como se observa en la Figura 1.11.

Figura 1.11. Cuadrante de Gartner Antivirus 2017 *[30]*.

2. METODOLOGÍA

Con el fin de elaborar las recomendaciones de buenas prácticas a partir del estudio de los principales tipos de malware Ransomware que han atacado en Ecuador a las estaciones de trabajo con sistema operativo Windows se realiza el análisis dinámico y estático, para lo cual se obtiene una lista top 10 de los tipos de Ransomware que más ataques han realizado en el país.

Dicha lista se la realiza después de analizar las páginas Hybrid Analysis [31] y SecureList [32]. Posterior a ello se recolectan 4 muestras de Ransomware en base a la lista anteriormente obtenida y a la disponibilidad de las mismas.

Luego de ello, se realiza el análisis de las muestras obtenidas, integrando algunos lineamientos de SANS Y MARE.

2.1. Obtención de un ranking de amenazas Ransomware en Ecuador con más impacto a equipos Windows.

Para obtener el ranking de amenazas Ransomware en Ecuador con mayor impacto se utilizaron los sitios web Hybrid Analysis [31] y Securelist [32].

En la Figura 2.1 se observa que según las estadísticas de Kaspersky [32] Ecuador tiene un 18.7% de usuarios infectados durante un mes.

Entre las 10 clases de malware que más atacan en Ecuador en las estadísticas de Kaspersky [32], no se encuentran muchos registros del tipo Ransomware como se observa en la Figura 2.1, por lo que para tener mayor información se optó por utilizar información del Hybrid Analysis.

Para encontrar las infecciones a causa del malware Ransomware en Hybrid Analysis, es necesario tener una cuenta registrada en el sitio y también indicar cuál es la razón por la se necesita acceso a la página.

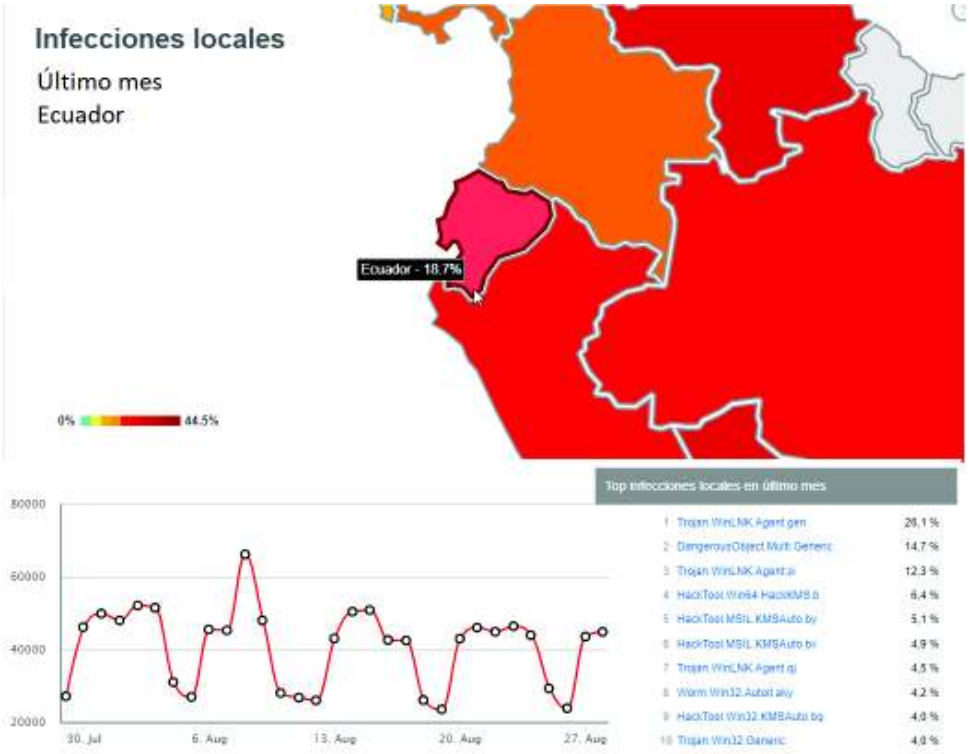

Figura 2.1. Estadísticas de infecciones Locales en Ecuador del último mes, tomada el 21 de Agosto del 2018 *[32]*.

Una vez validada la cuenta por parte del equipo de soporte de Hybrid Analysis, se filtró la búsqueda como se puede observar en la Figura 2.2.

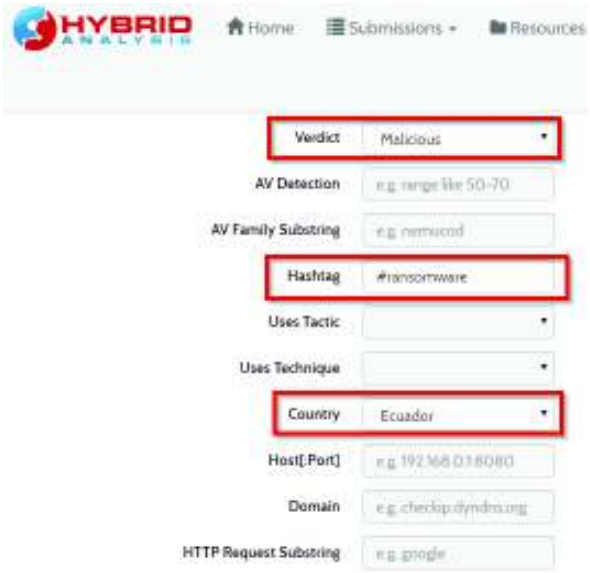

Figura 2.2. Opciones de búsqueda de reportes de Ransomware en Ecuador *[31]*.

Se tomaron en cuenta 115 muestras reportadas como Ransomware en Ecuador desde el 7 de marzo del 2018 hasta el 21 de agosto del 2018, con lo cual se las clasificó como se muestra en la Tabla 2.1.

Tabla 2.1. Muestras reportadas en Hybrid Analysis de Ransomware en Ecuador [31].

Ransomware	Número de muestras reportadas
WannaCryptor	27
Cryptowall	19
Cerber	15
Crysis	14
Locky	12
Satan	10
TeslaCrypt	8
Spora	6
Uiwix	3
CTBLocker	1

WannaCryptor:

Conocido como WannaCy es un Ransomware que se propaga automáticamente a través de redes internas y de Internet pública. Durante el año 2017, fue el ataque con mayor número de víctimas y contabilizó ataques a grandes empresas a través de una vulnerabilidad a nivel de Windows, que podría evitarse con uno de los parches de Microsoft.

WannaCry tiene dos componentes, uno que es la funcionalidad del ransomware y el otro que es utilizado para la propagación.

Cryptowall:

Cryptowall es un malware tipo Ransomware, que se distribuye a través de correo electrónico no deseado o a través de descargas como actualizaciones falsas de Flash Player.

Cryptowall modifica los datos en el equipo infectado, evitando que dichos datos sean utilizados o evitando que el equipo funcione correctamente. Cuando los datos son encriptados, el usuario recibe un mensaje en el que se solicita dinero a cambio de la información comprometida, y una vez realizado el pago la victima podrá recuperar los datos.

Cerber:

Cerber es uno de los malware tipo Ransomware más populares. Es diferente de otro malware ya que se ha actualizado varias veces y tiene muchas variantes.

Utiliza diferentes métodos de distribución y también es posible que cualquiera cree su propia versión y que parte de las ganancias sean entregadas a los creadores originales. Este malware cifra los archivos y solicita el pago a través de BitCoins, que es un tipo de divisa electrónica descentralizada que permite realizar transacciones anónimas.
Una de las características de Cerber es que analiza los archivos más utilizados, e intenta que sean los primeros en ser encriptados y puede hacerlo estando fuera de línea. Cuando los archivos son comprometidos toman la extensión .cerber.

Crysis:

Crysis se encontró entre los 5 tipos de Ransomware que más han atacado en Latinoamérica. Como casi todos estos tipos de malware solicitan dinero a cambio de rescatar la información.

Crysis utiliza una fusión de cifrado entre RSA y AES haciendo que el método de descifrado sea casi imposible, su método de distribución es por medio de archivos ejecutables maliciosos.

Locky:

Locky es un malware tipo Ransomware que se propaga a través de correo electrónico con un documento adjunto de Microsoft Office, el cual tiene un macro por detrás que el usuario no puede detectar. Cuando el equipo ya se encuentra infectado el primero paso es eliminar

un punto para que se pueda realizar una restauración del equipo, evitando que el usuario pueda recuperar la información sin necesidad de la clave que entrega el cyber delincuente.

Los documentos originales que han sido encriptados toman la extensión .locky, y utiliza los algoritmos de encriptación RSA y AES.

Satan:

Este malware agrega la extensión .satan a los archivos infectados y genera un archivo con una nota de rescate de la información.

En la actualidad existen muchas variantes de este malware, puesto que se ha creado un servicio que permite a cualquier persona crearse una cuenta y crear una versión personalizada del malware, en donde el delincuente decide la manera de distribución del mismo.

TeslaCrypt:

TeslaCrypt es un malware de tipo de Ransomware que fue detectado por primera vez en el año 2015.

Utiliza el algoritmo de encriptación AES-256, su método distribución es a través de correo electrónico spam con archivos ejecutables maliciosos. Inicialmente afectó a computadoras utilizadas para videojuegos infectando así a archivos jugadas, perfiles y mapas guardados. Además, en las nuevas versiones ya han sido afectados documentos con extensiones .doc, .png, jpg entre otras.

Spora:

Spora es un malware tipo Ransomware que es capaz de encriptar archivos que están fuera de línea. Es una de las muestras que tienen una mayor tasa de propagación.

Spora se distribuye principalmente a través de correos electrónicos no deseados con documentos adjuntos de facturas o recibos para llamar la atención de la víctima y una vez descargado el archivo aparece una ventana de Word que indica que el documento se encuentra dañado, mientras tanto detrás de este proceso, un malware se encuentra

escaneando todos los archivos con extensión de uso diario como .doc, .jpeg, .zip entre otros.

Uiwix:

UIWIX es un malware tipo Ransomware que una vez ejecutado en la memoria del equipo, elimina el archivo ejecutable que realizó la infección reduciendo así su huella y, disminuyendo la posibilidad de detección.

Una de sus principales características es que se da de baja cuando detecta que está siendo ejecutado en una máquina virtual.

CTBLocker:

CTBLocker, como gran parte de los malware tipo Ransomware llega a sus víctimas por medio de correo electrónico no deseado con un adjunto de extensión .zip.

Cuando la víctima se encuentra infectada, se le mostrará una imagen con un mensaje en varios idiomas indicando que debe realizar el pago mediante Bitcoins.

2.2. Recopilación de muestras de archivos dentro del ranking.

Para la obtención de las muestras necesarias para el análisis dinámico y estático, se toma en cuenta el listado que se encontraban en el ranking obtenido.

La descarga de las muestras se las realiza mediante el siguiente proceso:

1. Se ingresa al sitio https://www.hybrid-analysis.com, en el que es necesario realizar un registro con una cuenta corporativa, por lo que se utilizó el correo institucional de la Escuela Politécnica Nacional, para corroborar que las muestras serán utilizadas con fines académicos. Como se indica en la Figura 2.3.

Figura 2.3. Ingreso al sitio Hybrid Analysis *[31]*.

2. Para obtener la opción de “realizar descarga de muestras”, es necesario validar con el personal de Hybrid Analysis cuál es el objetivo de la descarga, con lo que se habilita la opción de descarga, como indica la figura 2.4.

Figura 2.4. Verificación de validez de la cuenta *[31]*.

3. En el menú de búsqueda se escoge la opción “Report Search”, seguido por la opción de “Advanced Search”, como se observa en la Figura 2.5.

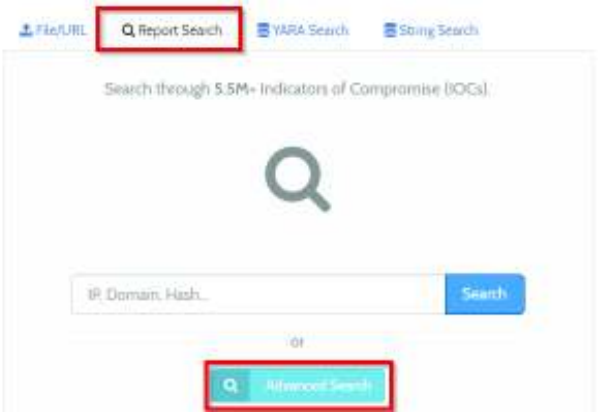

Figura 2.5. Proceso de búsqueda de muestras *[31]*.

4. En la opción “Hashtag”, se etiqueta el tipo de muestra que se desea encontrar y se selecciona el botón de “Search database”, como se demuestra en la Figura 2.6.

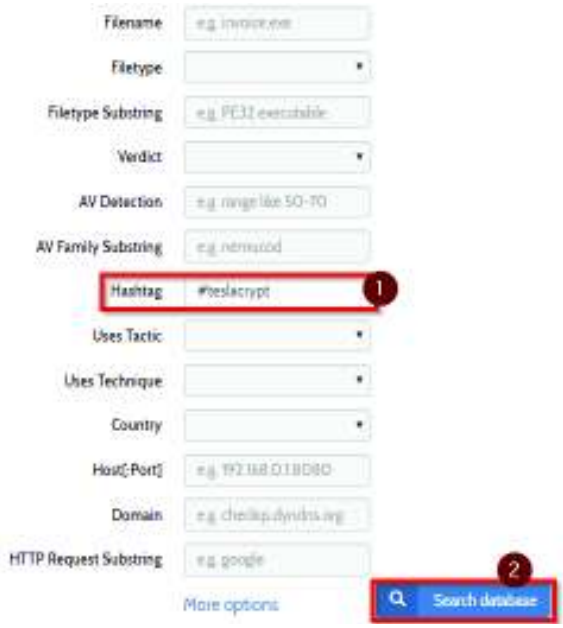

Figura 2.6. Filtrado de búsqueda de la muestra *[31]*.

5. El resultado de la búsqueda entrega información importante para la descarga de la muestra:

   (1) Fecha en la que fue detectada la muestra.

   (2) Hash de la muestra.

   (3) Etiquetas del tipo de muestra.

   (4) Indicador que existe una muestra disponible para la descarga.

Como se indica en la Figura 2.7.

Figura 2.7. Resultado de búsqueda de la muestra *[31]*.

WannaCry

La muestra se obtiene en el sitio Hybrid Analysis [31].

La muestra se la puede encontrar con el hash ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa y se realiza un análisis con VirusTotal [33], donde se puede observar que la amenaza ha sido detectada por varias soluciones de seguridad informática, como se puede observar en la Figura 2.8.

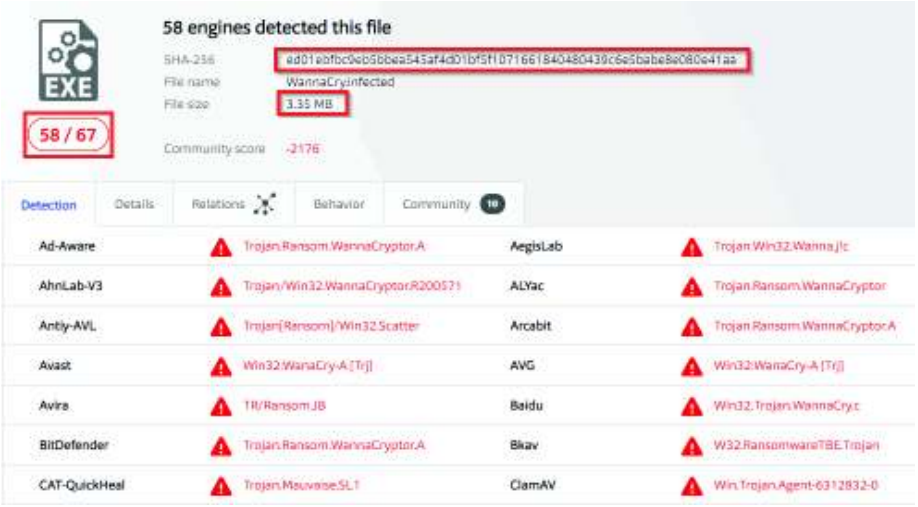

Figura 2.8. Detección de la muestra WannaCry *[33]*.

Spora:

La muestra se obtiene en el sitio Hybrid Analysis [31].

La muestra se la puede encontrar con el hash 2e6868e2bf0eefa5948f7a512d1eccbc705e97559f39c9e0f6e62378a8fdd548 y se realiza un análisis con VirusTotal [33], donde se puede observar que la amenaza ha sido detectada por varias soluciones de seguridad informática, como se puede observar en la Figura 2.9.

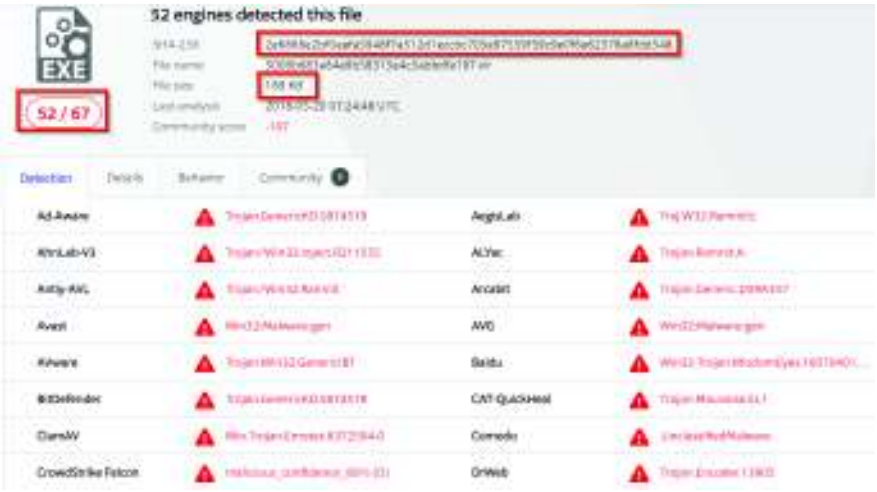

Figura 2.9. Detección de la muestra Spora *[33]*.

Locky:

La muestra se obtiene en el sitio Hybrid Analysis [31].

La muestra se la puede encontrar con el hash 8bf303dda84a1e0552f98370dd5dbfdf127d7ec9b5caab948874a897771ce142 y se realiza un análisis con VirusTotal [33], donde se puede observar que la amenaza ha sido detectada por varias soluciones de seguridad informática, como se puede observar en la Figura 2.10.

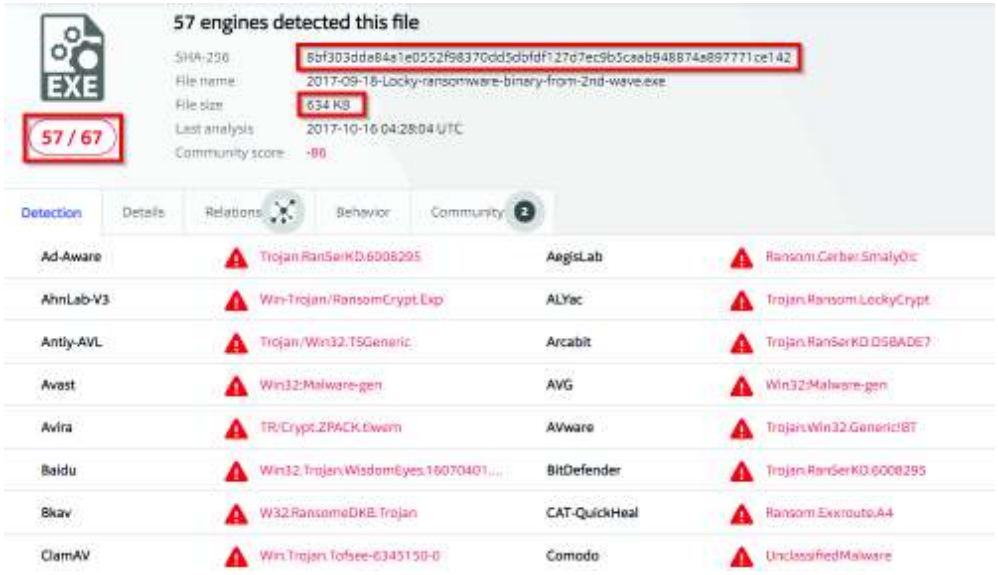

Figura 2.10. Detección de la muestra Locky *[33]*.

CTBLocker:

La muestra se obtiene en el sitio Hybrid Analysis [31].

La muestra se la puede encontrar con el hash 5445ec669432bdc6c283694bbe6309f60ef574c6d1e70b2f8df77514ef1638b0 y se realiza un análisis con VirusTotal [33], donde se puede observar que la amenaza ha sido detectada por varias soluciones de seguridad informática, como se puede observar en la Figura 2.11.

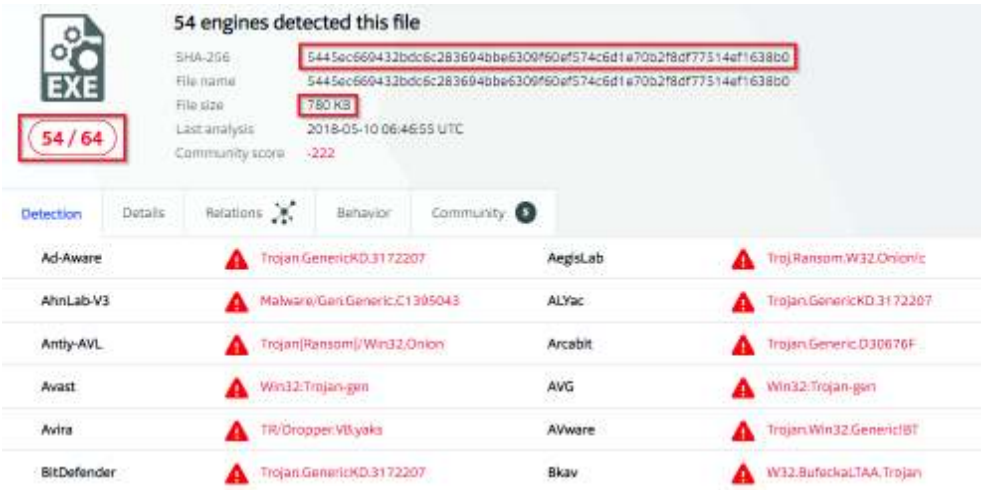

Figura 2.11. Detección de la muestra CTBLocker *[33]*.

2.3. Metodología a utilizar

Se procede a estudiar dos metodologías existentes y a tomar de ellas las propuestas y elementos que resulten para el propósito de la realización del presente trabajo.

En primer lugar, se analiza y resume el documento [34], para de esta manera plantear una metodología que se adapte a los objetivos fijados para este trabajo de titulación.

Estudio de Metodologías existentes para análisis de Malware

- **SANS**

El Instituto SANS es una organización encaminada al estudio y entrenamiento en ciberseguridad, seguridad de la información y administración de sistemas. En el documento

Malware Analysis: An introduction, proponen un método para el análisis de Malware el cuál ha sido bastante citado y aceptado. Dicho análisis propone procedimientos para llevar a cabo cada una de sus etapas sin entrar en pormenores en cuanto al uso de las herramientas propuestas [34].

En primer lugar se analizan los métodos para la obtención de muestras de Malware, mismas que pueden llegar a manos del analista de tres maneras.

La primera forma es la que ocurre en incidentes de infección de Malware en usuarios finales, es decir, cuando el usuario intencionalmente visita un servidor que ha sido comprometido con la presencia de dicho malware. Este tipo de encuentro con un software malintencionado supone la existencia de un incidente.

El segundo método estudiado es la adquisición de malware a través de Honeynets, es decir, el uso de servidores diseñados para atraer la atención de posibles atacantes para contener y estudiar los métodos utilizados por dichos Hackers para explotar vulnerabilidades de estos sistemas señuelo. Finalmente se propone la obtención de muestras de malware a través de uso de motores de búsqueda o sitios web especializados para el intercambio de código.

Se menciona al experto en ciberseguridad H.D. Moore conocido por el desarrollo de la plataforma de herramientas de seguridad Metasploit. Moore liberó en el año 2007 código para usar el motor de búsqueda Google para obtener muestras de malware.

- **Ambiente de análisis de malware:**

Previo a realizar el análisis de Malware es necesario preparar un ambiente adecuado para la realización controlada, segura y contenida de dicho proceso.

Para dicho procedimiento SANS propone un laboratorio con cuatro máquinas virtuales. Una máquina “Víctima” con sistema operativo Windows instalado, que es en la que el malware será ejecutado durante el análisis. Se propone un sistema operativo Windows XP [34].

Para el análisis de malware en un servidor Windows, se propone la creación de una máquina virtual con Windows Server 2003 conectado a la misma red virtual.

Una máquina virtual será la encargada de proveer servicios tales como DNS y DHCP. El documento propone que para tales propósitos se utilice un sistema operativo Linux; más específicamente, su distribución Mandriva.

Finalmente, los autores del documento añaden a su red de equipos virtuales un servidor Linux para que actúe como un equipo víctima Línux. Para dichos fines añaden una máquina virtual con sistema operativo Mandriva. Como se indica en la Figura 2.12.

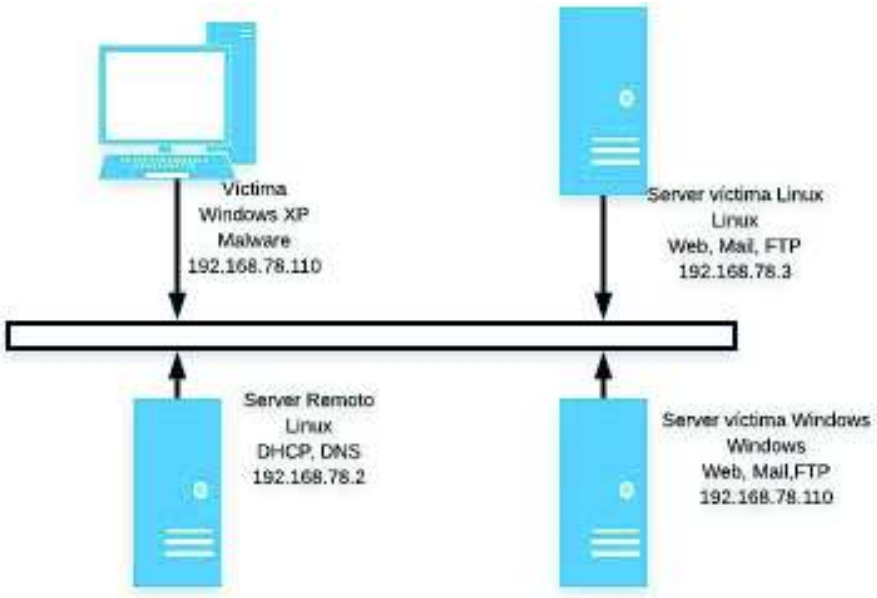

Figura 2.12. Red del ambiente virtual según la metodología SANS *[34]*.

Posterior a la instalación de los ambientes virtuales se procede a instalar las herramientas necesarias para la realización del análisis. Es indispensable tomar los hashes de las herramientas de análisis antimalware que se utilicen, para contrastarlas consigo mismas posteriormente para asegurarse que ningún malware ha modificado las herramientas.

Finalmente se realiza capturas del estado inicial de las máquinas virtuales, para restaurarlas cada vez que sea necesario.

- **Análisis Estático:**

Se propone de manera inicial realizar un escaneo de virus con distintos sistemas antimalware con el propósito de comprobar su detectabilidad.

Posteriormente se utiliza un editor hexadecimal para determinar la existencia de cadenas de caracteres, funciones citadas y direcciones web que puedan ser mencionadas. Este proceso permite conocer si el malware ha sido comprimido y especular sobre funcionalidad del malware.

Dentro de este análisis es necesario descomprimir el archivo ofuscado (de ser posible), capturar las cadenas de caracteres presentes en el archivo binario, tanto en ASCII como en UNICODE.

Este paso puede revelar información valiosa relacionada, por ejemplo los intentos del malware por comunicarse con servidores remotos. El presente proceso permite observar los registros y archivos creados, eliminados o modificados.

- **Análisis Dinámico:**

A continuación, se necesita ejecutar la muestra seleccionada para observarla en el laboratorio virtual con el propósito de verificar los efectos que esta ejecución produce directamente sobre el sistema.

Es necesario tener las precauciones previas a la ejecución de la muestra para se minimicen los posibles impactos del proceso.

Durante la ejecución de la muestra de Malware es importante observar los procesos iniciados por el Malware, así también las conexiones de tráfico que el malware intenta iniciar.

Para ambos propósitos es necesario permitir al malware ejecutarse por un tiempo determinado. En este caso se sugiere un tiempo de 15 minutos.

Es necesario tomar en cuenta la información desprendida por las herramientas utilizadas para el análisis, y recopilar la información que las mismas muestran para su análisis y comprensión.

- **MARE:**

Malware Analysis Reverse Engineering (Mare) [35], es un método de Análisis de Malware propuesto por Cory Q. Nguyen y James Goldman con el objetivo de crear un proceso formal

y estructurado a través del cual se pueda obtener reportes de hallazgos más completos y estructurados.

El principal aporte de la metodología MARE es un sistema estructurado de diagrama para el proceso de análisis de malware, que da una pauta clara de pasos a seguir.

El enfoque propuesto en la metodología MARE presenta una ruptura que no considera de manera completa el análisis estático y dinámico de las muestras de malware. Desde el punto de vista de los autores, se debe evitar el uso de herramientas si no se tiene en claro el proceso que debe seguirse para llevar el análisis de malware de una manera orgánica.

Además, la metodología MARE propone una lógica estructurada de cuándo y cómo usar las herramientas de análisis del malware.

MARE considera para su proceso una Línea de Tiempo de Defensa contra el Malware. La Metodología de Análisis de Malware e Ingeniería Inversa forma parte de este proceso. Este proceso considera seis “procesos”:

- Detección
- Aislamiento y extracción
- Análisis de Comportamiento
- Análisis del código e Ingeniería Inversa
- Reconocimiento de patrón
- Reparación

De estos seis procesos, los cuatro primeros comprenden la Metodología MARE. Estos cuatro procesos entregan a su salida el producto de una serie de pasos realizados durante su ejecución, estos son recogidos y propuestos por MARE para la consecución de resultados repetibles y aplicables. Como se indica en la Figura 2.13.

Figura 2.13. Línea de tiempo del proceso MARE *[35]*.

Cada uno de los procesos de la metodología MARE consta de diversos pasos, los cuales permiten la obtención de información útil para realizar el proceso siguiente. Es importante aclarar que la metodología de análisis de Malware propuesta por MARE no es necesariamente consecutiva, ya que los resultados de un proceso pueden realimentar la repetición de uno de los procesos anteriores como explica la Figura 2.14 [35].

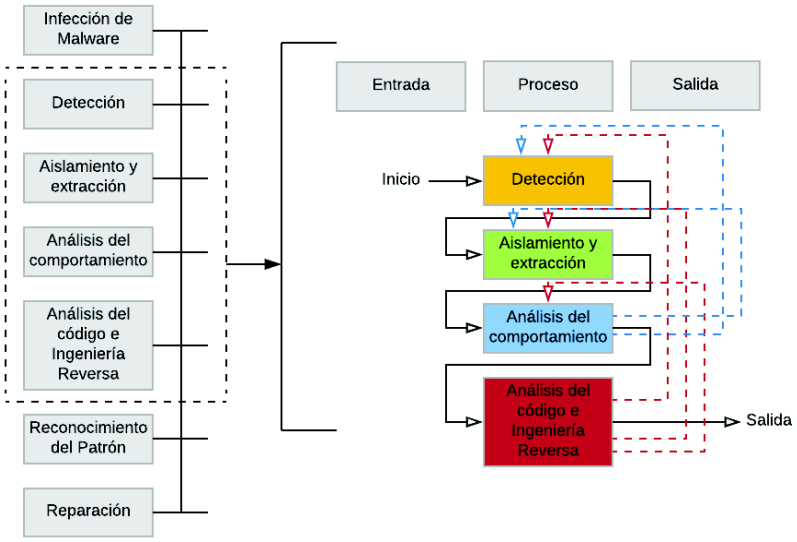

Figura 2.14. Metodología MARE *[35]*

A continuación, se realiza una descripción breve de cada uno de los procesos que forman parte de la línea del tiempo de la Línea de Tiempo de Defensa contra el Malware.

- **Detección:**

La metodología MARE, y la Línea de Tiempo de Defensa de Malware consideran como primer proceso la detección.

Durante esta fase se recomienda usar distintas herramientas antimalware para comprobar su detectabilidad. En caso de que la muestra de malware no fuese detectada por los distintos motores antimalware puede ser un indicador de que el malware en cuestión es muy reciente, o que su tecnología sofisticada le permite evadir la detección.

El documento presenta un esquema para la fase de detección de Malware, que muestra a la infección y/o obtención de la muestra como elemento de entrada. El proceso de detección propone tres pasos dentro de sí.

El uso de herramientas locales, es decir herramientas antivirus instaladas dentro de un computador (se recomienda que no sea una sola herramienta antivirus, ya que una muestra puede pasar inadvertida ante un antivirus y ser detectado por otro). Se propone también el uso de herramientas online de escaneo de Malware, las cuales analizan una muestra remitida y producen un reporte.

Finalmente, el documento sugiere incluir dentro del presente proceso la obtención de hashes de la muestra; ya que a través de esta información se puede obtener información sobre dicha muestra en análisis previos que hayan sido publicados [35].

A la salida del presente proceso puede obtenerse información sobre el tipo de malware con el que se está lidiando o los paquetes antivirus idóneos para tratar dicho malware, así también se puede estimar la amenaza que dicho Malware representa para el equipo o conjunto de equipos en cuestión.

- **Aislamiento y extracción**

La presente fase tiene por objetivo retirar la muestra del equipo infectado de manera que pueda ser transferido al laboratorio de análisis reduciendo el riesgo de infección de otros equipos.
Para esto se propone que la muestra sea almacenada en un archivo comprimido y asegurado con contraseña.

A la entrada del presente proceso se tiene el archivo sospechoso, el cuál es localizado durante la fase de detección. El Aislamiento del malware se realiza cuando el analista de Malware obtiene la ubicación del archivo sospechoso y su ubicación en el disco duro o espacio de memoria.

Durante este proceso se recomienda tener cuidado especial si se sospecha que el malware en cuestión es un Rootkit, ya que este tipo de Malware es difícil de detectar, y puede explotar vulnerabilidades del sistema operativo para realizar efectos adicionales como descargar otros programas maliciosos [35].

El objetivo primario al finalizar la fase de Aislamiento y detección es solamente obtener la muestra de malware a analizarse durante los procesos posteriores y dejar las bases para

que el analista formule hipótesis sobre el tipo de Malware con el que se está lidiando, información que será de gran utilidad en los procesos posteriores.

- **Análisis del comportamiento**

La correcta realización de los pasos que comprenden el análisis de comportamiento de malware permiten al analista conocer los archivos que el malware crea, modifica o elimina y los registros del sistema operativo que son afectados durante la ejecución de dicho malware. En esta fase es necesario determinar el propósito del malware y comprender su funcionamiento.

Se propone como primer paso el uso de un sistema automatizado de análisis dinámico de Malware (Sandboxes). Dichos servicios en línea presentan al analista un reporte previo de las configuraciones, registros y archivos que pudieran ser modificados.

Es importante considerar a este paso únicamente una guía para el proceso de análisis local que se haga en laboratorio ya que el malware puede presentar comportamientos adicionales no ejecutados durante el análisis automatizado (algunos malware son capaces de detectar ambientes Sandbox) [35].

Posteriormente se debe realizar una ejecución del Malware en el laboratorio de análisis, para esto se propone el uso de herramientas de monitoreo de procesos y captura de registros.

El método MARE propone también durante la esta fase realizar el monitoreo y captura de la actividad de red que ocurre durante la ejecución. Este paso proporciona al analista un panorama sobre el tipo de información que está siendo enviada hacia servidores remotos en caso de que el malware intente obtener información sobre el sistema.

A la salida del proceso de análisis de Comportamiento se puede obtener información sobre el origen del malware y las intenciones de las personas u organizaciones involucradas con su creación y distribución. Como se indica en la Figura 2.15.

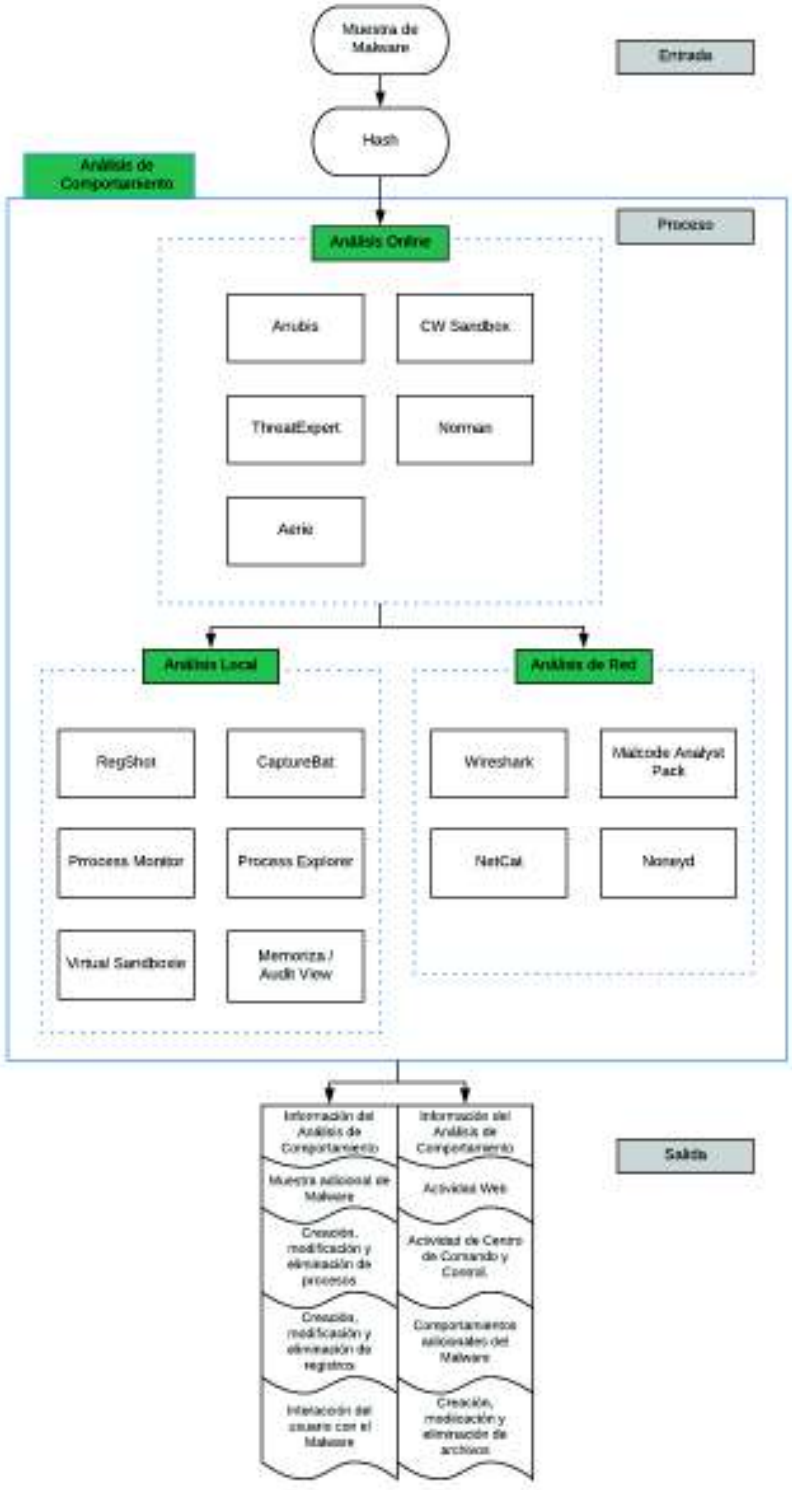

Figura 2.15. Proceso de Análisis de comportamiento MARE *[35]*.

Se obtiene como salida del proceso de análisis de comportamiento información de las actividades realizadas por el Malware durante su ejecución en Laboratorio. Este proceso, tiene utilidad directa sobre las pautas que revela, y brinda información útil para el posterior proceso de análisis de código.

- **Análisis de código e Ingeniería inversa:**

A la entrada del proceso de análisis de código se cuenta con la muestra de malware obtenida durante la fase de aislamiento y extracción, además de la utilidad que brinda el proceso de análisis de comportamiento en las pautas de funcionamiento del Malware.

El documento propone en inicio la búsqueda de cadenas de texto en el archivo de la muestra. Este paso se realiza mediante la utilización de herramientas diseñadas para este fin como Strings. Este paso permite al analista conocer los detalles con respecto a la ejecución del código, las funciones llamadas por el mismo e información de direcciones con las que el malware puede intentar contactarse.

Es imperativo determinar en el proceso de análisis de código si el código del malware ha sido comprimido y/o ofuscado. Es posible notar dicha ofuscación durante el paso anterior o mediante el uso de herramientas capaces de identificar la ofuscación y el tipo de algoritmo utilizado para la compresión, esto será de gran utilidad para desempaquetar el malware y continuar con su análisis.

En caso de no poder realizarse un desempaquetamiento del Malware, es necesario realizar un proceso de Debugging para observar paso a paso las funciones llamadas por el Malware y los espacios de memoria y archivos implicados durante su ejecución.

Solo después de obtener una muestra no ofuscada, el analista puede generar de la muestra de Malware información como las credenciales con las que dicho malware intenta contactarse con servidores remotos que pueden ser Centros de Comando y Control del Malware [35].

A continuación, se presenta el diagrama para el proceso de Análisis de Código e Ingeniería Inversa propuesto en la metodología MARE. Como se indica en la Figura 2.16.

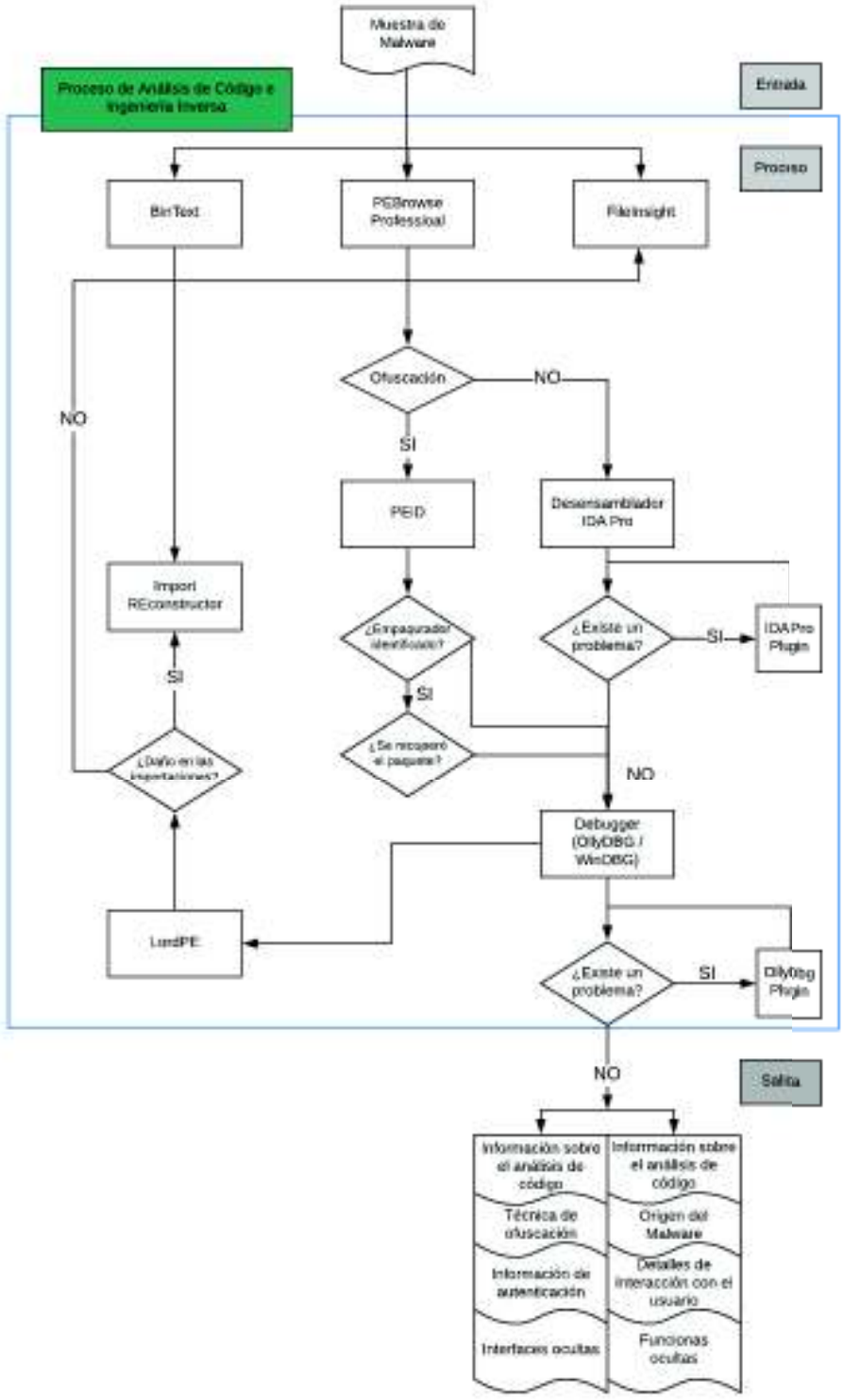

Figura 2.16. Proceso de análisis de código e Ingeniería Inversa *[35]*.

- **Reconocimiento de Patrón:**

Una vez realizado el análisis de malware dentro de la metodología propuesta, la fase de reconocimiento de patrones enfatiza en utilizar la información obtenida durante cada uno de los procesos anteriores, esto incluye el tipo de malware, el método de ofuscación, la forma en que el malware se propaga, etc.

Esta información se utiliza para crear métodos de prevención y selección de herramientas ante posibles futuros incidentes o ataques.

- **Reparación:**

En esta fase se desarrolla una herramienta o método para eliminar la infección y sus efectos de las máquinas afectadas por el malware.

Esta fase incluye tanto soluciones locales para las máquinas infectadas, como soluciones a nivel de sistemas de prevención y detección de intrusiones para bloquear la circulación del tipo de tráfico generado por el malware en la red.

Preparación del laboratorio Virtual con la Metodología propuesta

Previo a la realización del trabajo de análisis de Malware, se procede a la creación del laboratorio virtual que se utiliza para dichos fines. Durante la presente sección se exponen los criterios utilizados en la creación de tal laboratorio virtual.

Para la creación del laboratorio virtual se seleccionó la herramienta VMWare Workstation Pro de la compañía VMWare debido a sus funcionalidades como VMWare Tools, que transfiere archivos entre las máquinas huesped y anfitrión de una manera sencilla.

Otra característica principal de VM Workstation que será de gran importancia es la posibilidad de realizar capturas de estado de los equipos, lo cual permitirá retornar a un estado no infectado en cuestión de minutos después de haber ejecutado un malware en el equipo de análisis.

Para la instalación de la máquina virtual que hará las veces de víctima se ha seleccionado el sistema operativo Windows 7 en su versión Ultimate, debido a su amplia popularidad,

estabilidad y existencia de soporte. Se aplica la configuración recomendada por defecto en la herramienta virtual.

La máquina virtual queda instalada con 1 GB de memoria RAM y un disco duro virtual de 60 GB de asignamiento dinámico.

A continuación, se instala la máquina virtual dedicada a brindar servicios al equipo víctima. Para el equipo de servicios se utilizará un sistema operativo Kali Linux versión Kali-Roling publicada en el año 2016. Este equipo brindará los servicios HTTP, HTTPS, SMTP y hará las veces de Gateway a ojos de la máquina víctima.

El equipo de servicios queda instalado con una memoria RAM de 2GB y un disco duro virtual de 80 GB.

Es importante entender que la herramienta de virtualización VMWare Workstation Pro realiza una asignación dinámica de espacio de almacenamiento; es decir que el espacio que los equipos virtuales usan en el disco duro físico no se reservará en su totalidad sino que será asignado dinámicamente a medida que sea necesario.

La herramienta de Virtualización VMWare Workstation Pro les permite a las máquinas virtuales ejecutar dentro de la plataforma varios esquemas de conexión de red. Entre ellas una opción Bridge que conecta la máquina directamente a los dispositivos de red excluyendo a la máquina anfitriona de su uso.

Por otro lado, existe una opción de Network Address Translation en la cual, la máquina huésped se conecta a internet a través del equipo anfitrión que se encarga de gestionar su tráfico.

Además, se tiene el caso de redes virtuales llamadas VMNet capaces de conectar distintos equipos virtuales entre ellos. Para la realización del presente trabajo se usará la opción VMNet.

La opción de conectividad VMNet permite a los equipos del laboratorio virtual utilizar una red interna que conectará únicamente dichos equipos. Como se indica en la Figura 2.17.

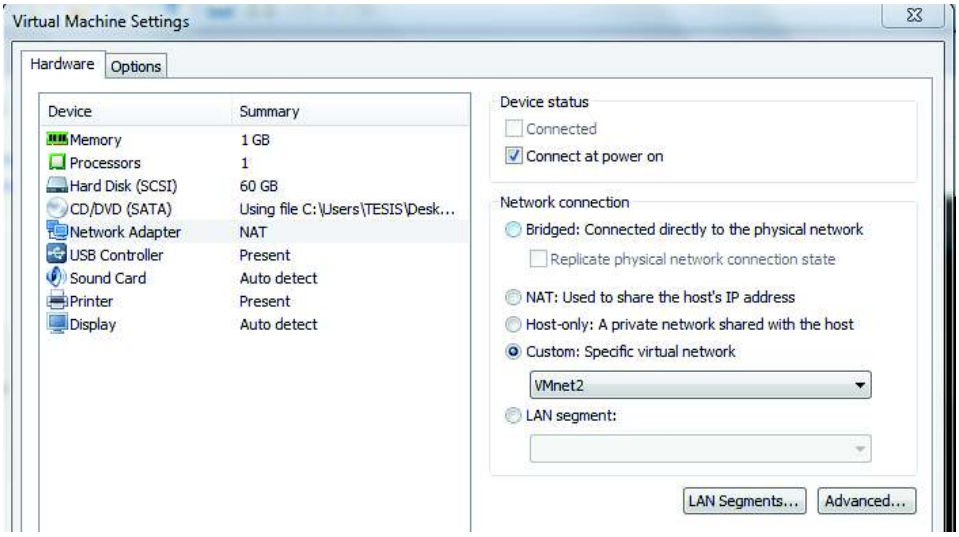

Figura 2.17. Configuración de la máquina virtual.

Las máquinas virtuales que forman parte del laboratorio de análisis no tendrán la capacidad de comunicarse por red con el computador físico ni de conectividad a internet. Contener a los equipos virtules dentro de una red virtual elimina la posibilidad que el equipo anfitrión como otros equipos (físicos o virtuales) conectados a su misma red, se vean afectados por el malware existente en el laboratorio de análisis y su posible difusión por actividad de red.

A continuación, se esquematiza la topología de red que es utilizada durante el desarrollo del presente trabajo. Como se indica en la Figura 2.18.

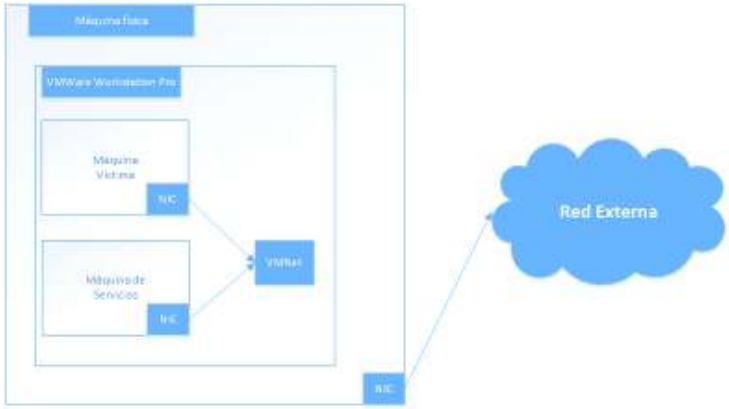

Figura 2.18. Esquema de la topología de red propuesto.

Adicional a la configuración de las tarjetas de red virtuales es necesaria la configuración de direcciones IP dentro de cada sistema operativo con una configuración estática.

A continuación, en la Figura 2.19 se presenta la configuración de direcciones a utilizarse durante el desarrollo del trabajo de análisis de malware, tanto para el equipo víctima como para el equipo de servicios.

Figura 2.19. Configuración de red del equipo Windows.

En la Figura 2.20 se indica la Configuración de red del Equipo.

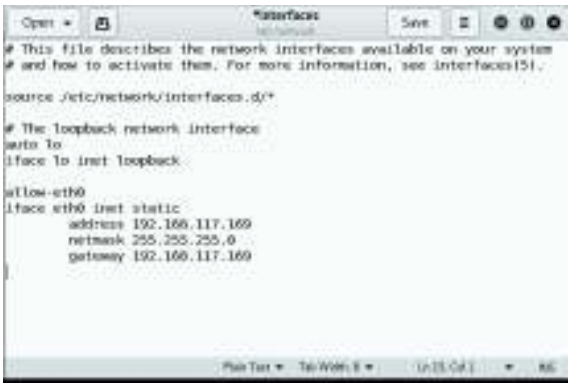

Figura 2.20. Configuración de red del equipo Kali Linux.

Se observa en el equipo víctima la dirección de gateway correspondiente a la dirección IP del equipo de servicios, y el servidor DNS configurado con la dirección de local host. Por su parte el equipo de servicios tiene su propia dirección configurada como gateway.

Proceso de Análisis Estático.

Para la realización del presente trabajo se toma en cuenta los pasos de las metodologías MARE y SANS descritas anteriormente para seleccionar tanto los pasos como las herramientas a utilizarse durante la realización del análisis estático básico.

Durante la presente sección se hace una explicación de los pasos considerados para el análisis estático básico realizado, las herramientas seleccionadas para utilizarse dentro de cada sección, y la utilidad de la información a obtenerse al realizar estos pasos.

- **Análisis con Antivirus**

El primer paso a tomarse en cuenta para este análisis es el uso de varios paquetes antimalware para confirmar la detectabilidad de las muestras.

Este paso es de gran utilidad en el caso de que la muestra que ha llegado a manos del analista haya sido recogida dentro de las bases de datos de paquetes de escaneo de virus comerciales disponibles. Durante este paso se recomienda utilizar varios paquetes antimalware, ya que alguna muestra podría no ser detectado por alguno de los paquetes antivirus. De esta manera, se incrementa la probabilidad de detección.

La principal utilidad del paso de Análisis con Antivirus es la identificación del malware con el que el analista deberá tratar. La correcta identificación del tipo de Malware bajo análisis permitirá al especialista estimar el posible impacto sobre el sistema, lo cual brinda indicios útiles hacia la remediación.

Es importante mencionar las limitaciones del escaneo de virus, de las cuales se mencionará dos de gran importancia.

Los ataques de día cero son irrupciones que se dan explotando vulnerabilidades del sistema no conocidas por los desarrolladores del sistema, este tipo de ataques no pueden ser contrarrestados de manera rápida al no existir mecanismos apropiados para neutralizar sus efectos.

Un ataque es llamado ataque de día cero desde que se conoce la vulnerabilidad, hasta que se ha desarrollado mecanismos para corregir la misma. De acuerdo al cuarto reporte trimestral de la compañía, un 30 % de los ataques reportados durante este periodo

correspondieron a ataques de día cero, claro ejemplo de los límites de los paquetes antivirus.

Adicional a esto se analiza el caso de las amenazas direccionadas, es decir, malware desarrollado de manera específica para atacar una cantidad reducida de objetivos en comparación con los malware de difusión masiva.

Para este análisis se utilizan 2 herramientas pagadas Sophos Home y Kasperky Internet Security las cuales fueron tomadas después de analizar su posicionamiento en el mercado según el cuadrante mágico de Gartner [30] y los resultados de funcionamiento según NSS Labs [29], adicional se utilizó una herramienta gratuita Avast Free Antivirus, las 3 herramientas serán descargadas de sus sitios oficiales.

- **Avast Free Antivirus:**

La herramienta Avast utilizada tiene las siguientes especificaciones:

- Versión: 18.6.2349
- Fabricane: AVAST Software
- Protección: Análisis de todos los archivos nuevos y añadidos en el equipo, alerta en caso de que exista un comportamiento sospechoso en el equipo, bloquea los ataques vía web.

La versión utilizada es gratuita y en caso de querer tener más funciones de protección se puede optar por la versión pagada.

- **Análisis con Kaspersky Internet Security**

La herramienta Kaspersky utilizada tiene las siguientes especificaciones:

- Versión: 19.0.0.1088
- Fabricane: Kaspersky Lab
- Protección: En tiempo real, de privacidad, control parental.

Esta herramienta puede ser utilizada de manera gratuita por 30 días, durante los cuales la base de datos se actualizan periódicamente, una vez vencido el tiempo de prueba, la base de datos no se actualizará y en caso de que existan nuevos ataques, no serán detectados.

- **Análisis con Sophos Home**

La herramienta Sophos utilizada tiene las siguientes especificaciones:

- Versión: 1.2.1
- Fabricane: Sophos Limited
- Protección: En tiempo real, a aplicaciones no deseadas, descargas por reputación, navegación Web por categorías.

La versión utilizada es gratuita y en caso de querer tener más funciones de protección se puede optar por la versión pagada.

- **Toma de Huellas de Archivo y propiedades estáticas.**

La toma de huellas (Hashing) corresponde a la aplicación de una función matemática a un archivo determinado.

Dicho algoritmo producirá una cadena de caracteres de la misma longitud independientemente de la longitud del archivo de entrada. Una característica deseable en una función Hash es que sea resistente a colisiones, es decir que sea de gran dificultad que dos entradas distintas produzcan la misma salida.
Además es deseable que dos entradas similares produzcan a la salida de la función Hash cadenas de caracteres bastante distintas.

Los dos algoritmos más utilizados en la toma de Hashes dentro del análisis de Malware son el algoritmo MD5 desarrollado en el año de 1991 por Ronald Rivest, y el algoritmo SHA-1 desarrollado por la Agencia Nacional de Seguridad de los Estados Unidos de América en el año de 1995.

Las funciones Hash pueden ser utilizadas para criptografía, verificación de integridad y chequeo de paridad [36]. Dentro del proceso de análisis de Malware se utilizan herramientas de cálculo de funciones hash para identificación de archivos potencialmente maliciosos (buscando información sobre los mismos a través de su hash), además se utilizan para detectar la modificación en archivos binarios benignos.

En el desarrollo del presente trabajo se utilizaron las herramientas detalladas a continuación para el cálculo y verificación de hashes.

- **WinMD5Free:**

La herramienta WinMD5Free es una herramienta gratuita y de libre distribución diseñada para trabajar en distintas versiones de sistemas operativos Windows. Se trata de una herramienta portable, que ocupa poco espacio en disco y que presenta una interfaz gráfica de usuario amigable, fácil de usar para propósitos de cálculo y comparación de las cadenas de caracteres producidas al correr el algoritmo MD5. Como se indica en la Figura 2.21.

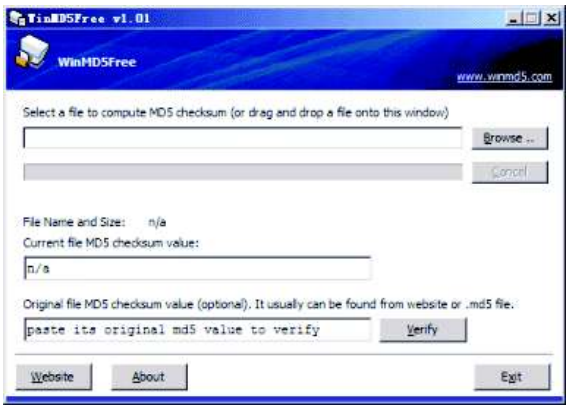

Figura 2.21. Interfaz de WinMD5Free.

- **7ZIP**

7zip es una herramienta de compresión de archivos gratuita, de código abierto y de libre distribución. Permite comprimir y descomprimir archivos de distintos formatos (7z, XZ, BZIP2, GZIP, TAR, ZIP, etc) y dentro de sus utilidades extras, permite el cálculo de códigos de redundancia cícilca (CRC) y hashes con los algoritmo SHA-1 y SHA-256, como se indica en la Figura 2.22.

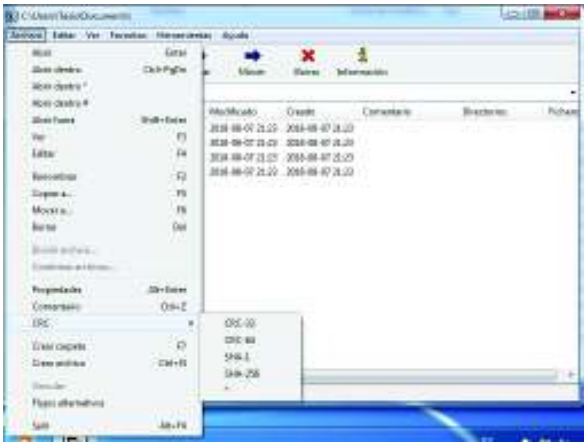

Figura 2.22. Interfaz de 7Zip.

Una ventaja adicional de utilizar el software de compresión 7Zip es que añade una función directa para el cálculo de hashes SHA en el menú de contexto de Windows, al hacer click derecho en cualquier archivo. Lo cual supone un atajo útil al momento de realizar la toma de hashes.

- **Búsqueda de cadenas de caracteres**

Dentro de un archivo binario, existen secuencias de bits que pueden traducirse a cadenas de caracteres Unicode o ASCII.

Estas cadenas de caracteres pueden representar mensajes impresos por el ejecutable, direcciones URL a las que el malware intentará conectarse o instrucciones que el malware ejecuta a través del API de Windows.

Un editor hexadecimal es una herramienta útil para una identificación inicial de la presencia de cadenas de caracteres, ya que muestra los posibles caracteres producidos por cada Byte.

Durante este paso es necesario que el analista dedique atención especial durante el análisis con editor hexadecimal, ya que este proceso permite visualizar cadenas de caracteres de la abstracción total del archivo examinado, pero no las extrae, es decir que los posibles caracteres se muestran en medio de gran cantidad de símbolos que en realidad no significan nada.

Sin embargo, es necesario utilizar el editor hexadecimal como primer paso, ya que hay secuencias de caracteres que podrían ser ignoradas por los extractores automáticos, los cuales por defecto toman en cuenta como strings únicamente secuencias de una cantidad determinada de caracteres (típicamente tres o cuatro).

Posteriormente a la búsqueda de cadenas de caracteres mediante editor hexadecimal es recomendable utilizar una herramienta adecuada para la extracción automática de strings. Estas herramientas son capaces de buscar cadenas de caracteres tanto en Unicode como en ASCII y las imprimen en pantalla, lo que posibilita el almacenaje para análisis posteriores y documentación de reportes sobre el análisis.

Durante la primera fase del análisis será de gran utilidad encontrar posibles encabezados como “MZ”, que significa que el archivo en cuestión está en formato Portable Executable, es decir un archivo ejecutable dentro de sistemas operativos Windows.

Este formato de archivo tiene un tipo de estructura necesaria para que el sistema operativo Windows pueda ejecutar el código contenido. En caso de que el archivo no se halle encriptado, podrán visualizarse también las cabeceras de las distintas secciones del programa.

Es posible dentro del análisis de caracteres, cómo ya se había mencionado, encontrar URLs o direcciones IP, a las cuales es presumible que el software maligno en cuestión intente conectarse.

En caso de existir es importante entender que existe un riesgo en intentar contactarse directamente hacia esas direcciones. Se recomienda en primera instancia utilizar un servicio online WHOIS, para consultar la procedencia y registro de dichas direcciones. Es importante señalar que estas podrían corresponder a un Centro de Comando y Control, el cuál esperaría obtener información del equipo infectado.

A continuación, se estudian las herramientas a utilizarse durante el presente paso de análisis estático.

- **FileInsight:**

FileInsight es un software desarrollado por McAffee Labs que sirve para analizar archivos e incluye un editor hexadecimal entre otras herramientas diseñadas para tener una comprensión del archivo a analizarse.

McAfee FileInsight funciona de manera independiente puesto que no forma parte de un paquete antivirus McAfee, es de distribución gratuita, código cerrado y protegida por derechos de autor. A continuación, se presenta una vista preeliminar de la interfaz gráfica de usuario del software en cuestión.

Como se observa en la Figura 2.23. FileInsight brinda grandes facilidades para el desarrollo del análisis, ya que muestra adicionalmente las secciones del archivo ejecutable PE, así como las librerías que se importan. Esto es de gran utilidad para comprender las funciones invocadas por el malware.

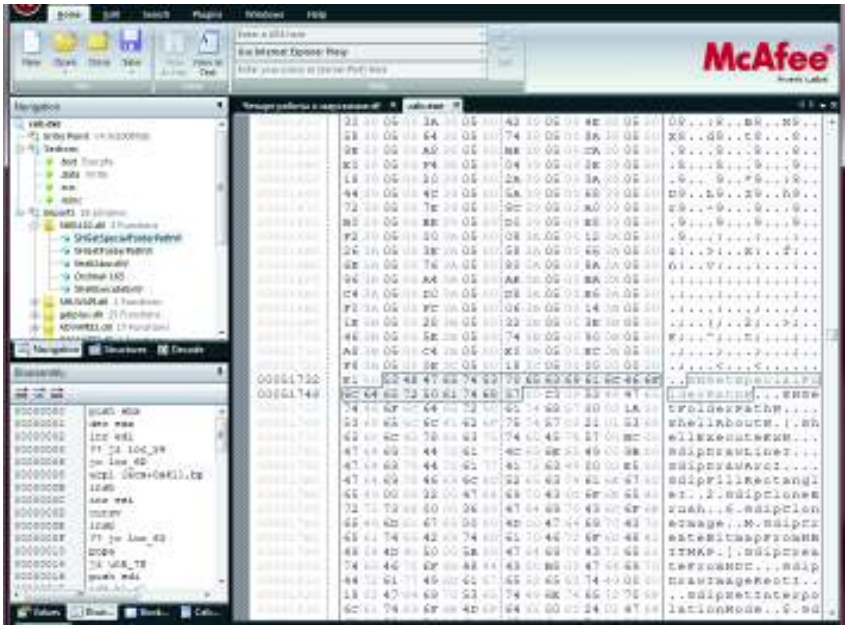

Figura 2.23. Interfaz de FileInsight.

File Insight contiene incluso un pequeño módulo de desensamblaje, a través del que se pueden observar algunas funciones a las que el archivo binario hace llamado.

- **Strings:**

Strings es una herramienta de tamaño pequeño, de libre descarga y que forma parte del paquete de utilidades Sysinternals, distribuido por Microsoft.

De acuerdo a su sitio web de distribución, el paquete administra, diagnostica y soluciona problemas de aplicaciones en el entorno de Windows.

Originalmente Sysinternals fue desarrollado por la compañía Winternals Software LP, la cual fue adquirida por Microsoft en el año 2006.

Strings es un software ejecutable por línea de comando en el entorno CMD de Windows. Se ejecuta por medio de una sintaxis simple y su función principal por defecto es mostrar en pantalla las cadenas de caracteres encontradas durante el análisis de un archivo cuya longitud sea igual o mayor a tres símbolos. La salida del comando Strings puede guardarse dentro de un archivo de texto gracias a la característica de recursividad de funciones del Command Prompt de Windows, como se indica en la Figura 2.24.

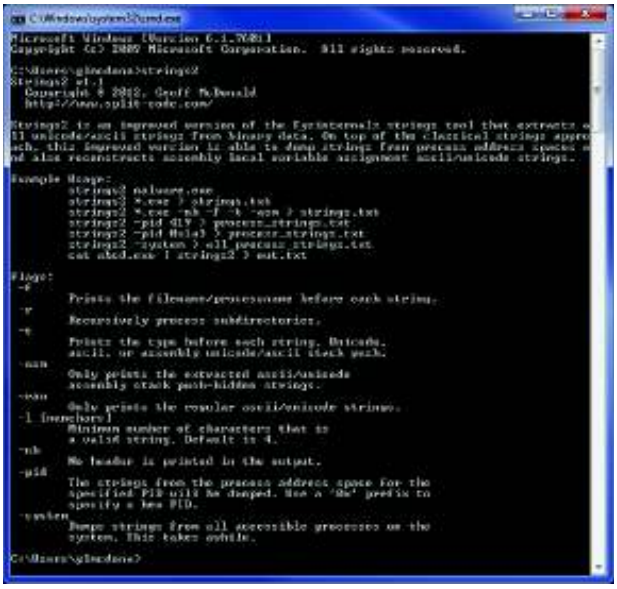

Figura 2.24. Salida del comando Strings.

Se puede observar en la Figura 2.25 la sintaxis utilizada por el programa Strings para su ejecución en Windows y la salida posterior a que ha sido aplicado.

Figura 2.25. Sintaxis del comando Strings.

- **Detección de empaquetamiento y ofuscación.**

Durante el proceso de búsqueda de cadenas de caracteres pueden obtenerse muy pocos resultados debido a que el contenido del archivo ejecutable se ha empaquetado para evitar la ingeniería inversa.

Adicionalmente, empaquetar el Malware reduce el tamaño del archivo haciendo más fácil su descarga, y cambia los hashes del archivo, lo que puede dificultar su detección. El empaquetamiento no es un proceso realizado únicamente por elaboradores de malware, también está presente en software benigno con la intención de ocultar el código con el que se ha desarrollado.

Existen varias opciones para empaquetar un archivo ejecutable las cuales pueden realizarse en escritorio o mediante aplicaciones online. Dentro de los empaquetadores, el más popular es UPX (Ultimate Packer for eXecutables), el cuál es fácilmente desempaquetable, sin embargo es un empaquetador de código abierto, lo que significa que existen variantes a partir de este programa. Esto dificulta los desempaquetamientos si el software se ha empaquetado utilizando una variable.

El empaquetamiento del software puede detectarse a menudo durante la etapa de búsqueda de strings, sin embargo no es sencillo detectar el tipo de empaquetamiento que se ha utilizado para ocultar el contenido del Malware.

Por esta razón es necesario el uso de herramientas que identifiquen el packer que se utilizó para ofuscar el Malware.

Una vez detectado el empaquetador utilizado por el Malware, el proceso de desempaquetamiento es variable dependiendo del empaquetador usado y no se cubrirá en la presente sección.

Este proceso, puede incluir distintas herramientas de software y su dificultad puede ser altamente significativa, por lo que un desempaquetamiento no siempre es viable.

A continuación, se realiza una descripción breve de PEiD, Software utilizado en el presente proyecto con el fin de detectar e identificar empaquetadores existentes en el malware.

- **PEiD**

PEiD es capaz de detectar al menos 470 distintos tipos de firmas en archivos PE, lo cual incluye los empaquetadores, encriptadores y compliladores más comunes.

PEiD contiene además plugins capaces de detectar algoritmos criptográficos, además detecta los puntos de entrada del Malware, e incluso desempaqueta el archivo analizado.

Al momento de realización del presente trabajo, el proyecto PEiD ha sido descontinuado, sin embargo, el software en cuestión continúa siendo altamente referido dentro de los trabajos de análisis de Malware estudiados.

A continuación, se presenta la interfaz gráfica de usuario presentada por el software PEiD para su uso en sistemas operativos Windows, como se indica en la Figura 2.26.

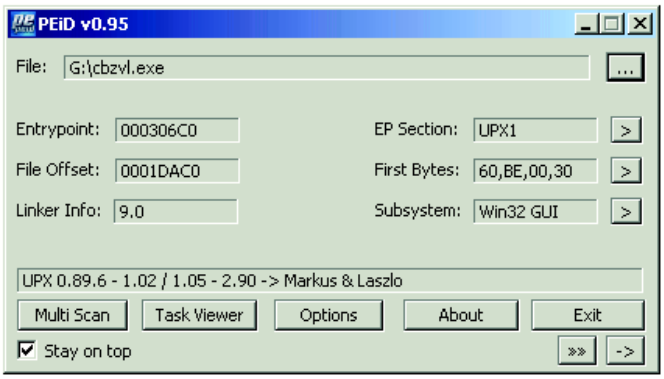

Figura 2.26. Interfaz gráfica de PEiD.

Durante la realización del presente trabajo, en caso de que el malware analizado se encuentre encriptado, se analizará el empaquetador que fue usado, y se intentará su desencripción.

En caso de lograr desencriptar el archivo binario, se procederá nuevamente con el análisis estático de la muestra desempaquetada en búsqueda de nuevos indicios sobre el funcionamiento del Malware.

Es importante tener en cuenta que el análisis de malware utilizando PEiD puede provocar la ejecución inadvertida del mismo, razón por la cuál es importante regresar la máquina virtual de análisis dentro del laboratorio virtual a su estado base después de que el análisis de empaquetamiento se realice.

- **Librerías y funciones invocadas.**

Un software ejecutado dentro del sistema operativo Windows interactúa con el mismo mediante la importación de Bibliotecas de Enlace Dinámico (DLL), e interacción directa con la API de Windows.

Dentro del proceso de análisis de Malware, la invocación de librerías y la ejecución de instrucciones será de particular interés del analista de malware, quien puede servirse de estas librerías para obtener indicios de acciones tomadas por el software malicioso para afectar al sistema, o interactuar con el mismo con otros propósitos tales como obtener información o acceder a servicios de red.
La utilidad y uso de las funciones importadas y utilizadas por el software malicioso pueden ser consultadas por el analista a través de la Red de Desarrollo de Microsoft (MSDN), librería online de Microsoft con gran cantidad de información sobre las principales funciones de la API de Microsoft, además de recursos didácticos de distintos fines de gran ayuda para comprender el desarrollo dentro del entorno de Microsoft Windows.

A continuación, se describe el software que será de principal ayuda dentro de la presente sección:

- **Dependency Walker:**

El software Dependency Walker es un software de distribución gratuito para Windows, que analiza librerías y funciones importadas. Su organización en un esquema de árbol de dependencias, vincula cada función utilizada con su librería correspondiente.

Esta aplicación además detecta problemas en la ejecución de un software, y busca las librerías y funciones de las que este software depende.

A pesar de que el Software Dependency Walker se encuentra descontinuado, es referenciado en la Red de Desarrollo de Microsoft como un método utilizado para analizar las librerías DLL de las que un archivo depende [37].

En la Figura 2.27 se presenta la interfaz gráfica del software Dependency Walker en el cual se puede observar la salida del análisis de un ejecutable, lo cual incluye un esquema de árbol de las librerías relacionadas con el programa, y las funciones utilizadas por el mismo.

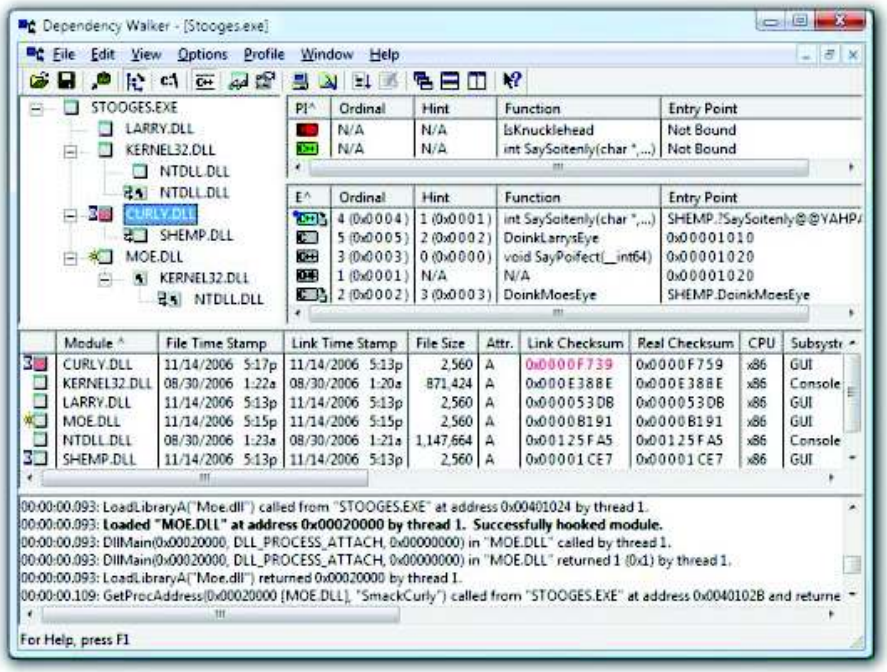

Figura 2.27. Interfaz gráfica de Dependency Walker.

Se puede observar en la interfaz de usuario, en la parte izquierda el despliegue del esquema de árbol correspondiente a los archivos DLL invocados por el código del archivo, mientras que en la parte derecha un listado de las funciones citadas.

- **Dependencies:**

Dependencies es un proyecto de código abierto publicado en Github y desarrollado por los usuarios Tobias Fenster y lucasg.

En su sitio oficial se describe Dependencies como una reescritura del software Dependency Walker cuyo objetivo es ayudar a los desarrolladores de Windows a identificar inconvenientes relacionados con la carga de la importación de librerías dinámicas en sus programas [38].

A pesar de que Dependencies no ofrece las funcionalidades completas de Dependency Walker, este software maneja de mejor manera la exploración de dependencias en el sistema operativo Windows 7.

En la Figura 2.28, se presenta la interfaz gráfica de usuario de Dependencies que cuenta con muchas características similares a la interfaz presentada por Dependency Walker.

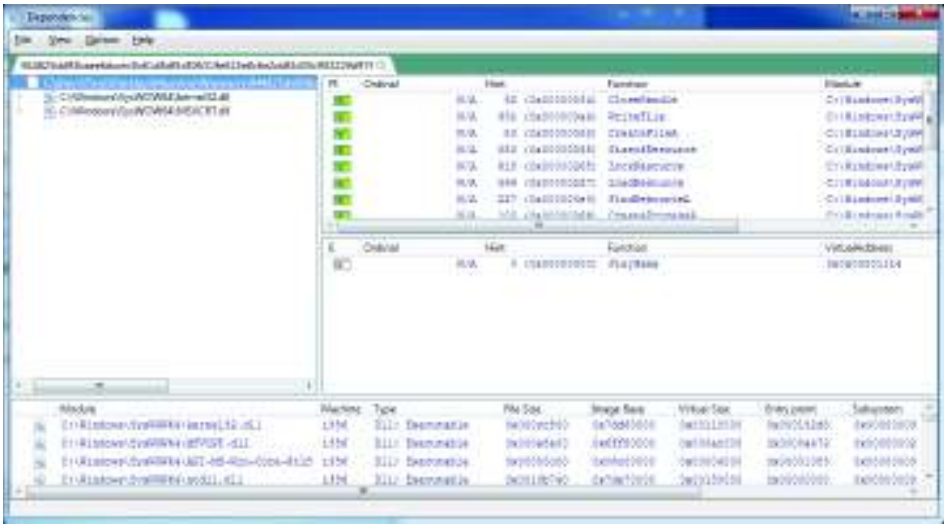

Figura 2.28. Interfaz gráfica de Dependencies.

Proceso de Análisis Dinámico.

Es importante tener en cuenta de antemano que durante el análisis dinámico el malware será ejecutado en la máquina víctima para observar los efectos causados por el mismo en el sistema operativo.

Parte de la información que se observa durante el proceso de análisis dinámico puede variar, razón por la cuál es importante repetir varias veces el mismo proceso con la finalidad de obtener la mayor cantidad de información posible.

Aún dado que el trabajo presente se desarrolla dentro de un laboratorio virtual, es necesario verificar periódicamente la configuración de red de las máquinas virtuales involucradas antes de ejecutar el malware, ya que, un descuido en verificar dicha configuración podría resultar en la infección de la máquina física huésped u otras máquinas conectadas a la misma red, y causar la pérdida de información o comprometimiento de la información almacenada en las mismas.

El proceso de análisis dinámico supone la capacidad de observar el malware de manera más directa, y analizar sus efectos en el sistema operativo. Esto contribuye a confirmar o descartar las sospechas levantadas durante el proceso de análisis estático.

Para el presente trabajo se adoptó el esquema propuesto por la metodología MARE que considera tres etapas dentro del desarrollo del análisis dinámico.

En primer lugar se considerará un análisis automatizado a partir de un reporte del malware mediante la utilización de un ambiente automatizado.

En una analogía con el análisis inicial con software antivirus durante el proceso de análisis estático, dicho análisis automatizado no puede considerarse suficiente dentro del análisis dinámico, debido a varios factores, que incluyen la variación del comportamiento de Malware en diferentes entornos, así como los requisitos que el sistema debe cumplir para la correcta ejecución del malware.

Posteriormente se realizará un análisis local de los efectos dinámicos del Malware. Dicha etapa recolectará distinta información a capturarse antes, durante y después de la ejecución de dicho malware en el ambiente de análisis.

Dentro de los datos se analizará las modificaciones realizadas dentro del editor de registros de Windows, los archivos creados, modificados o eliminados, los procesos existentes, entre otros.

Finalmente se realizará un análisis de la actividad de red generada por el Malware. Durante esta etapa es necesario utilizar la máquina virtual de servicios. La máquina de servicios con sistema operativo Kali Linux será la encargada de proveer servicios a la máquina víctima, con el objetivo de simular acceso a internet.

Para el análisis de tráfico en dicha etapa es necesaria la utilización de herramientas tanto en la máquina víctima como en la de servicios.

- **Análisis Automatizado:**

Para el análisis automatizado se utiliza herramientas Sandbox, para de esta manera utilizar un entorno controlado con el fin de conocer el comportamiento y procesos utilizados de las muestras a ser analizadas

  - **CWSandbox:**

CWSandbox también conocido como Threat Track, es una herramienta que ejecuta los archivos maliciosos, una de sus principales ventajas es que su sistema remoto proporciona al usuario la seguridad de evitar ejecutar el malware.

En su sitio web se pueden enviar muestras infectadas de hasta 16MB y cargarlas a su sistema. Cuando el archivo o muestra ya haya sido analizado se pueden observar los resultados ya sea en su página web o serán enviados al correo electrónico registrado

Es una herramienta diseñada para analizar el comportamiento de los procesos y cambios que existen en un equipo y determinar si la muestra es maliciosa.

CWSandbox entrega informes con gran cantidad de información la cual que puede ser entendida por usuarios sin muchos conocimientos sobre el tema.

Adicional se pude realizar el análisis de muchas muestras a la vez, de esta manera se optimizarán los tiempos durante el análisis.

Para la utilización de esta herramienta es necesario tener una cuenta en su sitio, para lo cual se solicitó a la marcar Threat Track un tiempo de prueba de su herramienta. Se se puede tener el servicio Sandbox tando en la nube como en un equipo físico, para este caso se utilizará el servicio en la nube.el servicio en la nube.

- **Análisis Local:**

Dentro del proceso de análisis dinámico, se llamará análisis local al que se realice dentro del equipo a infectarse con malware (equipo víctima) para la comprensión de sus efectos.

En esta sección se hace una lista las herramientas a utilizarse durante el proceso, y se menciona la información que espera obtenerse a partir de la utilización de cada herramienta.

- **Autoruns:**

Autoruns es una herramienta de pequeño tamaño, que forma parte del paquete de utilidades Sysinternals y disponible para descarga gratuita a través de los sitios de Microsoft.

De acuerdo a la información provista por Microsoft, Autruns es una herramienta utilizada para el monitoreo de tareas de inicio. Autoruns es capaz de mostrar los programas que se ejecutarán durante el arranque del sistema operativo, o el ingreso de un usuario y observar las propiedades del mismo, como se indica en la Figura 2.29.

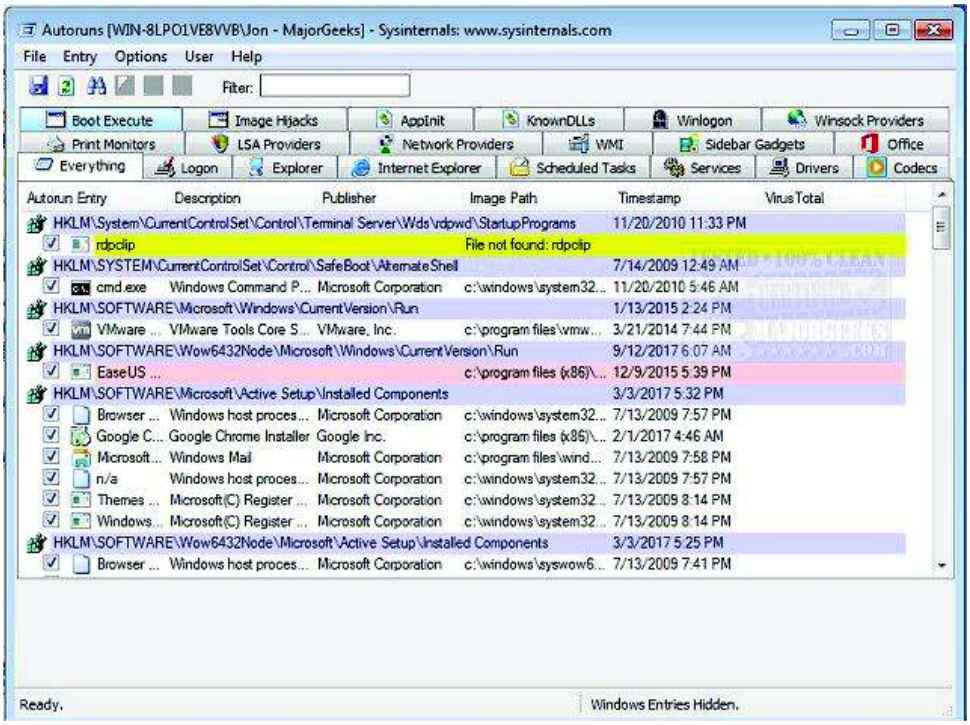

Figura 2.29. Ejecución de Autoruns.

Autoruns permite guardar los resultados de sus análisis y brinda la posibilidad de comparar un análisis realizado con uno guardado anteriormente, por lo que es posible realizar conocer de manera sencilla las tareas programadas para el inicio del sistema que han sido creadas posterior a la instalación de un software, ya sea este benigno o maligno.

Autoruns permite observar tareas programadas, Registros modificados, utilidades adicionales adheridas a exploradores de internet, DLLs, códecs para multimedia entre otras.

- **Regshot:**

Regshot es una herramienta de código abierto, distribuída bajo licencia LGPL.

Se trata de una herramienta bastante simple que permite realizar dos capturas del registro de Windows, y después compararlas con el propósito de detectar cambios realizados. La aplicación de esta herramienta para análisis de malware es muy sencilla:

- ✓ Se toma la captura inicial.
- ✓ Se ejecuta el malware.
- ✓ Se realiza la segunda captura.
- ✓ Finalmente se compara ambos registros haciendo click sobre el botón comparar.

Regshot genera a su salida un reporte de los cambios existentes en el registro, el cuál es almacenado en la dirección que se le indique. A continuación en la Figura 2.30 se presenta la interfaz gráfica de Regshot:

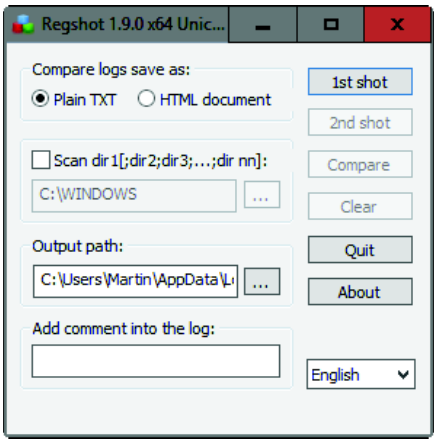

Figura 2.30. Interfaz gráfica de CaptureBat.

- **Process Explorer:**

Preocess Explorer es una herramienta portable, de libre descarga y que forma parte del paquete de utilidades Sysinternals, distribuido por Microsoft.

De acuerdo a su sitio web de distribución, el paquete tiene por objeto ser de utilidad para administrar, diagnosticar y solucionar problemas de aplicaciones en el entorno de Windows.

Process Explorer permite tener una vista de los procesos que se encuentran corriendo en el sistema. El programa muestra un esquema de árbol que permite ver a qué programas corresponden los procesos. Además permite obtener información similar a la obtenida en CaptureBAT.

Dentro del esquema de árbol se resaltan con distintos colores:

- ✓ Servicios.
- ✓ Procesos.
- ✓ Procesos nuevos.
- ✓ Procesos terminados.

De esta manera el explorador de procesos observa aquellos que han arrancado o terminado durante la ejecución del Malware.

Process Explorer presenta utilidades muy importantes a momento de analizar malware, como:

- ✓ Acceder a las propiedades de un DLL.
- ✓ Verificar la firma de seguridad de Microsoft para comprobar si una aplicación es la auténtica o se trata de una suplantación.

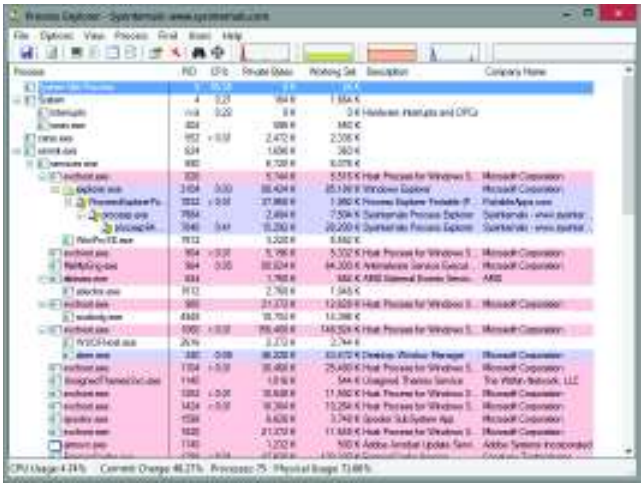

Figura 2.31. Interfaz gráfica de Process Explorer

La Figura 2.31 presenta la interfaz gráfica de Process Explorer y el esquema de árbol de servicios y procesos.

- **Process Monitor.**

Process Monitor es una herramienta de monitoreo avanzada disponible en el entorno de Windows que permite realizar un monitoreo tanto de archivos como de registros.

Process Monitor captura la información de las acciones llevadas a cabo por distintos procesos que se encuentren activos en la computadora durante el periodo monitoreado.

La información capturada por Process Monitor puede ser filtrada para obtener únicamente detalles sobre las llamadas realizadas por procesos que son de interés; dicha característica permite explorar con mayor facilidad la información importante para la realización del análisis.

Es crucial considerar que la recolección de información sobre procesos realizada por Process Monitor se realiza sobre la memoria RAM, es decir que si Process Monitor realiza una captura durante un tiempo muy prolongado podría llenar la memoria disponible causando fallos en el sistema operativo.

A continuación, en la Figura 2.32 se presenta la interfaz gráfica de usuario de Process Monitor.

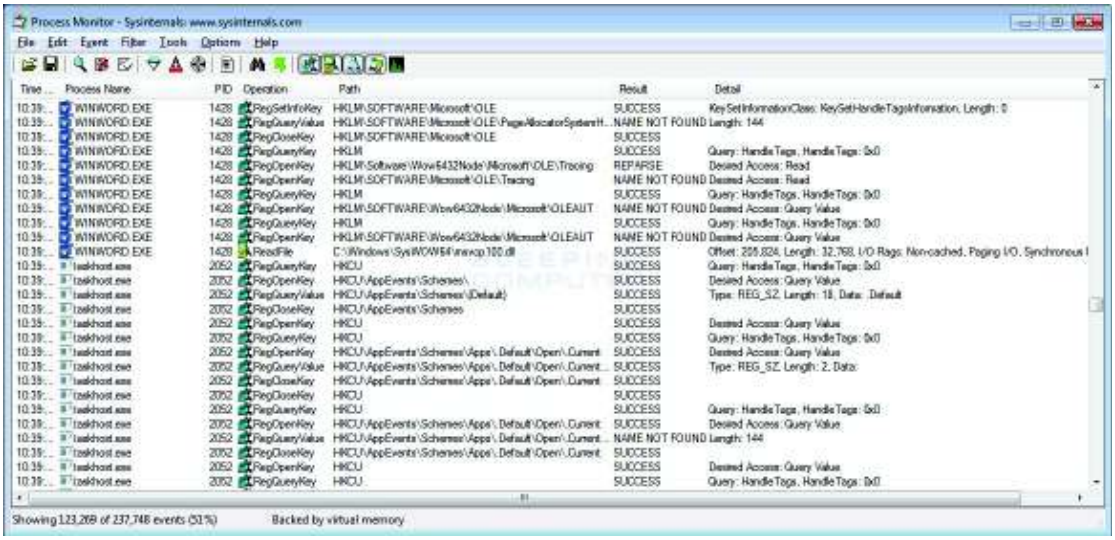

Figura 2.32. Interfaz gráfica de Process Monitor.

- **Análisis de Actividad de Red**

La presente etapa del análisis dinámico o de comportamiento permite observar la actividad de red de la máquina victima posterior a la infección.

Es importante mencionar que la presente etapa considera software corriendo en dos ambientes virtuales distintos. El tráfico generado por el malware será analizado desde la propia máquina víctima y desde la máquina virtual de servicios conectada directamente.

La máquina de servicios ayudará a monitorear la actividad de red de la máquina víctima y servirá para simular conectividad de red, para que el malware muestre funcionalidades que ocultaría ante ausencia de conectividad.

Para la realización de esta etapa es importante que la configuración de red de la máquina víctima tenga configurada a la máquina de servicios como puerta de enlace y la dirección de local host como servidor DNS.

Por otro lado, la máquina de servicios tendrá instalado un software capaz de simular varios servicios de internet de forma mucho más sencilla y eficiente que configurándolos individualmente.

Previo al análisis de actividad web de Malware será importante realizar las siguientes verificaciones:

- La existencia de tarjetas de red virtuales en cada equipo y su correcto funcionamiento.
- Que ambas máquinas virtuales se encuentren dentro de una misma red virtual.
- Que dicha red virtual se encuentra aislada de otras redes.
- Que ambas máquinas virtuales pueden contactarse entre sí.
- Que la máquina víctima pueda contactarse con los puertos necesarios de la máquina de servicios.
- Que no existen máquinas adicionales conectadas a la misma red virtual.

A continuación, se indican las herramientas necesarias para el análisis de actividad de red que forma parte del análisis dinámico de malware.

- **ApateDNS:**

ApateDNS es una utilidad desarrollada por la compañía FireEye, de distribución gratuita bajo registro y escrita por Steve Davis en el año 2011.

ApateDNS es una herramienta capaz de funcionar como un servidor DNS falso ubicado en la máquina local (Para el caso concerniente, se instalará en la máquina víctima).

El software escucha las peticiones realizadas a través del puerto 53 UDP, y responde a dichas solicitudes con una dirección IP especificada por el usuario a través de la interfaz gráfica.

Aparte DNS constituye un spoofer de DNS que funciona a nivel local y que captura las solicitudes internas de resolución de nombre de dominio en su interfaz para posteriormente mostrarlas.

La principal utilidad de Aparte DNS es observar nombres de dominios solicitados por el software malicioso y que pudieran corresponder a servidores de comando y control para dicho malware.

En la Figura 2.33 se presenta la interfaz gráfica de ApateDNS donde puede observarse los nombres de dominio capturados.

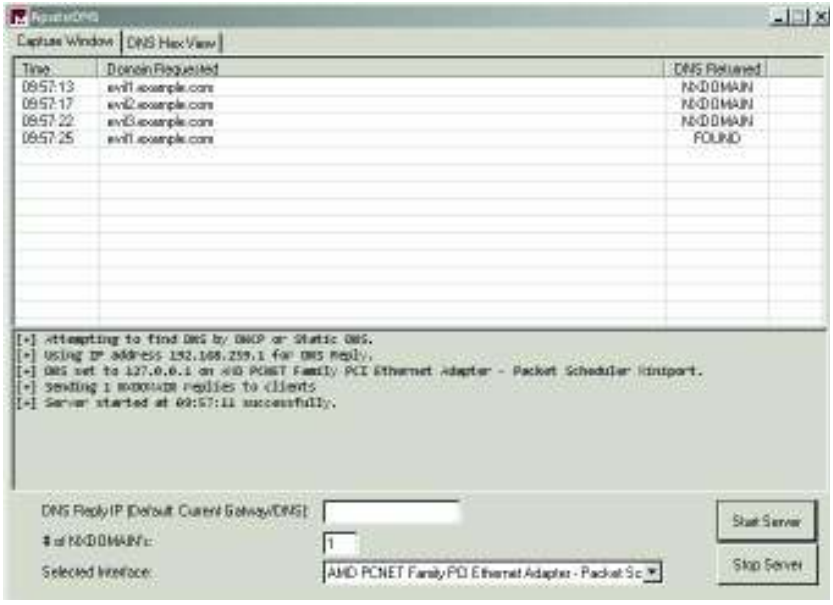

Figura 2.33. Interfaz gráfica de ApateDNS.

- **Tcpdump:**

Tcpdump es una herramienta desarrollada originalmente en el año 1988 por un conjunto de trabajadores del Lawrence Berkeley Laboratory.

Tcpdump es una herramienta muy completa utilizada para la observación de tráfico de red. Funciona mediante un intérprete de línea de comandos. Tcpdump muestra en pantalla el contenido de un paquete de red.

Para el caso práctico del presente estudio técnico Tcpdump se utilizará para capturar los paquetes enviados desde la máquina virtual víctima en sus intentos por establecer comunicaciones con el internet u otros computadores de la red.

Tcpdump se implementará en la máquina virtual de servicios para monitorear la actividad de red de la máquina víctima.

- **Wireshark:**

Wireshark es un analizador de protocolos utilizado principalmente para realizar una captura de los paquetes de computadores en una red para diagnosticar problemas existentes en la red.

Wireshark es una herramienta de código abierto, y de licencia GPL, es decir de libre distribución. Originalmente lanzado en el año de 1999, Wireshark se ha convertido en el sniffer de mayor uso en distintos ámbitos.

Dentro del contexto de análisis de Malware, Wireshark puede ser de gran utilidad para analizar la manera en que el malware intenta establecer sus comunicaciones de red.

Las ventajas de Wireshark son:

- Gran número de filtros con los cuales elegir los datos capturados a visualizarse, de manera que solo se muestren los que son de interés para el análisis.
- Indicar de manera conjunta los mensajes pertenecientes a una misma comunicación en ambos sentidos, lo que ahorra esfuerzo con respecto a buscar dicha información paquete por paquete.

En la Figura 2.34 se presenta una captura de pantalla de la interfaz principal de Wireshark, en la misma se puede observar la información capturada siendo filtrada a partir de dos parámetros: una dirección IP origen y un puerto TCP.

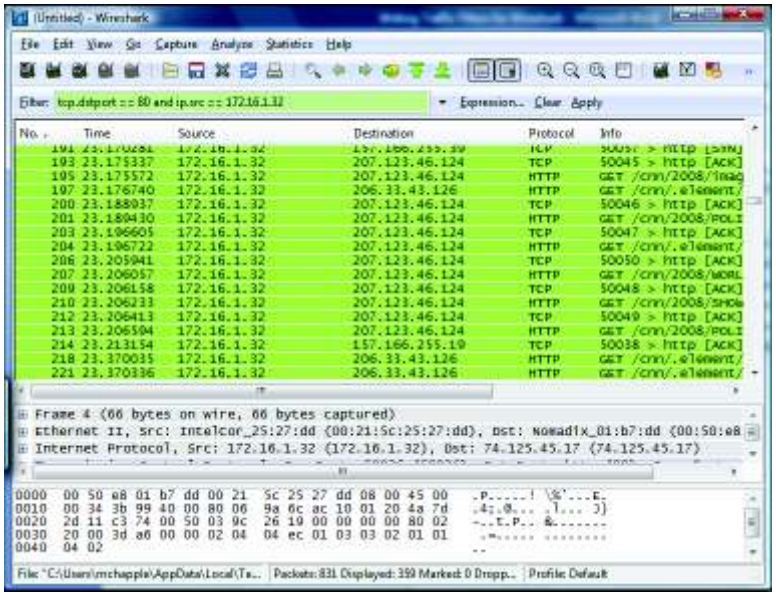

Figura 2.34. Interfaz gráfica de Wireshark.

Dentro del contexto de la realización del presente trabajo el analizador de protocolos Wireshark se instalará en la máquina de servicios, desde la cual se observarán las comunicaciones desarrolladas a través de distintos protocolos o puertos.

- **INetSim:**

INetSim es un paquete de programas utilizada para simular servicios de red. De acuerdo a la información provista por el sitio web de INetSim, esta herramienta fue desarrollada específicamente con el propósito de utilizarse en tareas de análisis de malware [39].

El uso de INetSim facilita la tarea de los analistas de malware al presentar un log conjunto y funciones de control centralizadas, lo cual facilita la configuración de los servicios individualmente. INetSim se trata de una suite de código abierto distribuido a través de licencia GNU GPL. A partir de este paquete Kali Linux se intentará establecer comunicaciones con la muestra de malware, las mismas que serán capturadas para el análisis de Malware por NetCat y WireShark.

3. RESULTADOS Y DISCUSIÓN

3.1. Proceso de Análisis Estático

Análisis de CTBLocker

- **Análisis con Antivirus**

Acorde con el procedimiento propuesto en el Capítulo 2, dentro de la máquina virtual víctima se realizaron pruebas de detección con tres herramientas antimalware locales para comprobar el nivel de detección de las muestras de malware por parte de dichas herramientas.

Adicionalmente a esto se mostrarán los resultados de búsqueda de archivos maliciosos del sitio online Hybrid Analysis para conocer si un archivo es potencialmente malicioso a partir de diferentes motores.

En esta sección se presentarán los resultados de detección en tiempo real de los motores de búsqueda de virus.

- **Avast**

En la Figura 3.1 se observa que Avast detecta el archivo infectado y lo mueve al Baúl de virus. La herramienta tiene almacenada a la muestra analizada con el nombre de Win32:Trojan-gen. La gravedad con la que se posiciona a la muestra es baja.

Figura 3.1. Detección de CTBLocker por Avast.

- **Kaspersky**

En la Figura 3.2 se puede observar que Kaspersky Internet Security realiza el siguiente proceso para detectar una amenaza:

1. Detección.
2. Aislamiento y puesta en cuarentena del archivo.
3. Eliminación del archivo infectado.

Adicional Kaspersky muestra el nombre con el que la muestra se encuentra en su base de datos, la cual es Trojan.Downloader.Win32.Onion.vxg.

Figura 3.2. Detección de CTBLocker por Kaspersky.

- **Sophos**

En la Figura 3.3, se evidencia que CTBLocker es detectado y eliminado por Sophos, en la administración web se observan las acciones y los detalles de la localización del archivo malicioso. El nombre definido por Sophos es Troj/Agent-ALFM.

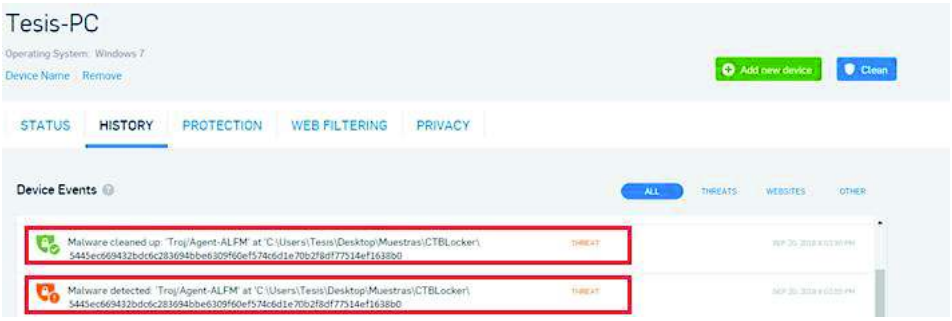

Figura 3.3. Detección de CTBLocker por Sophos.

- **Hydrid Analysis**

En la Figura 3.4, se observa como Hybrid Analysis, realiza la validación con 3 distintos motores de antivirus, para determinar que la muestra se encuentra infectada.

El nombre con el que la muestra se reconoce es Factuur_00887362833441.pdf.exe. En este caso se realiza la validación con CrowdStrike, Metadefender y VirusTotal.

Figura 3.4. Detección de CTBLocker por Hybrid Analysis.

- **Toma de Huellas de Archivo y propiedades estáticas**
  - **Toma de Huellas**

Se extrajeron las huellas de archivo correspondientes a los algoritmos MD5, SHA1 y SHA 256 para la muestra del malware CTBLocker, los cuales se presentan a continuación en la Tabla 3.1.

Tabla 3.1. Algoritmos de malware CTBLocker.

ALGORITMO	VALOR
MD5	77fac4194a04d2bbd9b4503044f4250c
SHA1	303EB16BB294B22F41FEF00D8EEE6ADAB6D81775
SHA256	5445ec669432bdc6c283694bbe6309f60ef574c6d1e70b2f8df77514 ef1638b0

- **Propiedades Estáticas**

A continuación en la Figura 3.5. se expone la información extraída a partir de la cabecera PE del archivo CTBLocker:

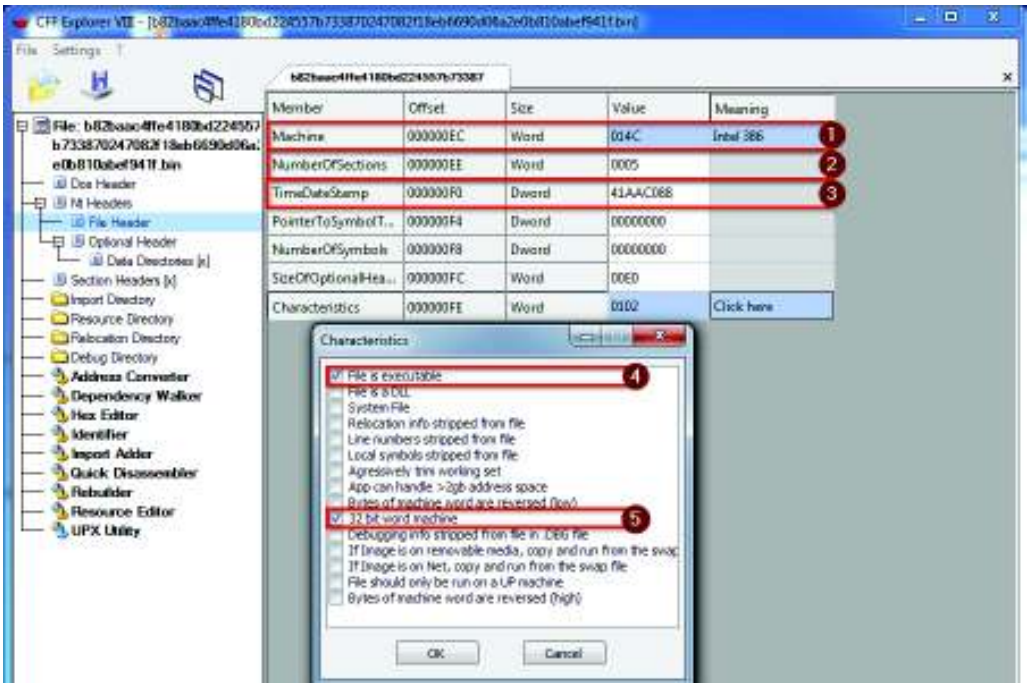

Figura 3.5. Resultado de la cabecera de CTBLocker.

En la Figura 3.6. se indica la traducción del equivalente de la fecha de compilación de CTBLocker.

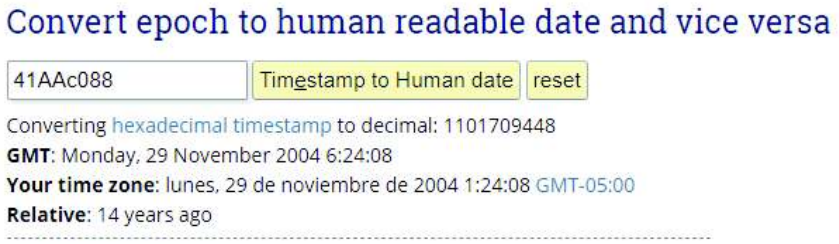

Figura 3.6. Traducción del equivalente de la fecha de compilación de CTBLocker.

De la cabecera PE del archivo se extrae que:

1. El archivo fue diseñado para una arquitectura Inter 386, es decir que el código de ensamblaje utilizado es de 32 bits.
2. Se tienen cinco secciones dentro del archivo.

3. Se consultó el equivalente TimeDateStamp en el sitio web Epoch Converter [40], como se observa en la Figura 3.6 obteniéndose que el archivo fue compilado el 29 de Noviembre del 2004, a las 1h24, hora de Ecuador.
4. El archivo es un ejecutable.
5. Utiliza palabras de 32 bits.

Como se indica en la Figura 3.7, de la cabecera opcional se extrae qué:

1. Se trata de un ejecutable portable de 32 bits.
2. El código empieza a ejecutarse en algún punto de la sección ".text".
3. Las secciones deben ubicarse en espacios de disco múltiplos de 200.
4. El programa puede relocalizarse si la sección solicitada para cargarse en memoria estuviera ocupada.
5. Existen secciones del código que pueden marcarse para no ejecución.
6. El programa puede conectarse con algún tipo de aplicación remota

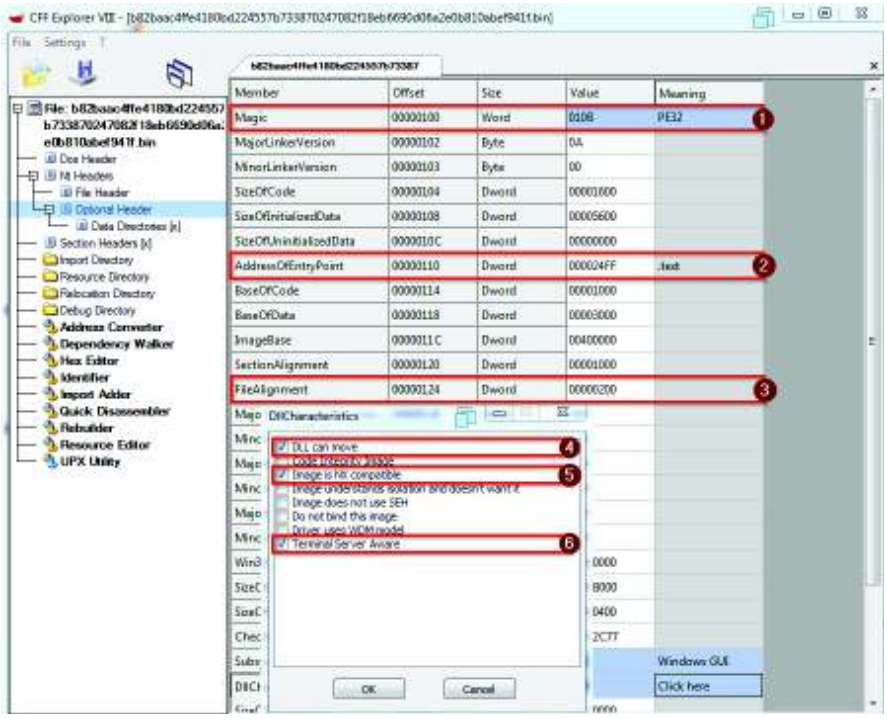

Figura 3.7. Traducción del equivalente de la fecha de compilación de CTBLocker.

En la Figura 3.8. se indica la sección de información de directorios, de la que se obtiene:

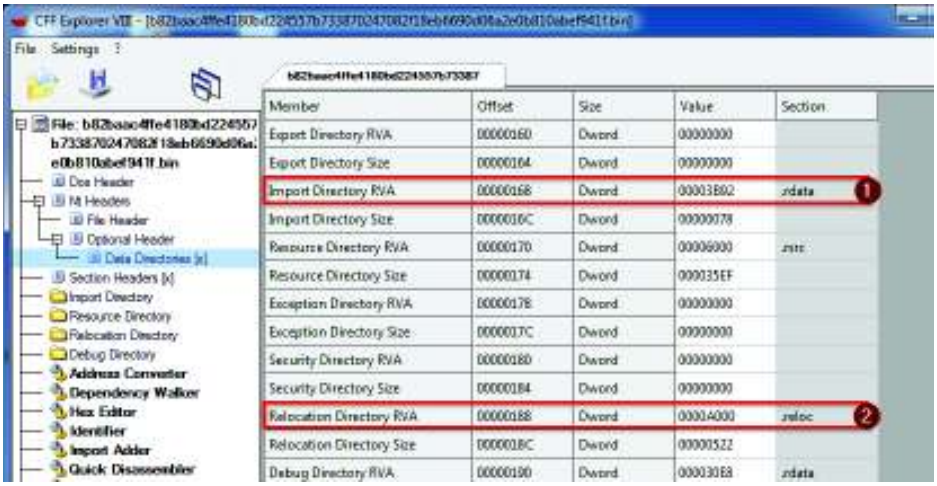

Figura 3.8. Información de cabecera de CTBLocker.

1. El directorio de importaciones se encuentra en la sección “.rdata”.
2. El directorio de relocalización se encuentra ubicado en la sección “.reloc”.

En la Figura 3.9, se indica que de los encabezados de sección obtenidos:

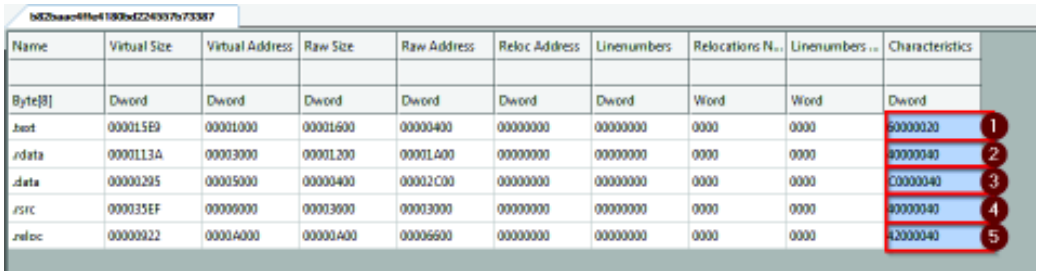

Figura 3.9. Encabezados de CTBLocker.

1. La sección “.text” contiene código ejecutable, y tiene permisos de lectura y ejecución.
2. La sección “.rdata” contiene variables que han sido inicializadas, y tiene permiso de solo lectura.
3. La sección “.data” contiene información que ha sido inicializada y tiene permisos de lectura y escritura.
4. La sección “.rsrc” tiene permisos de solo lectura, y contiene información o variables que han sido inicializadas

La sección ".reloc" tiene permisos de solo lectura, y contiene información o variables que han sido inicializadas, además esta sección puede ser descartada de memoria una vez que el código esté corriendo en caso de existir problemas con el manejo de memoria.

- **Búsqueda de cadenas de caracteres**

A continuación se realiza un reporte sobre los Strings encontrados dentro del archivo binario de la muestra de malware CTB Locker.

Para el análisis de cadenas de caracteres se utilizaron en conjunto dos herramientas:

- El editor hexadecimal File Insight.
- El programa Strings, parte del paquete de programas Sysinternals.

Strings apoyó con la extracción de las cadenas de caracteres presentes en toda la muestra binaria.

Dichas muestras binarias fueron exportadas a un archivo de texto. Posteriormente, de las secuencias de caracteres extraídas se seleccionaron las que tenían un significado válido como: Nombres de funciones y librerías correspondientes al Windows API, instrucciones correspondientes a lenguajes de programación, nombres de archivos, URLs y direcciones IP, entre otras.

Finalmente se utilizó FileInsight para clasificar las secciones del archivo EP dentro del que se ubicaban las cadenas de caracteres para estimar su posible uso dentro del programa.

A continuación se realiza un reporte correspondiente al resultado de análisis de cadenas de caracteres encontradas.

Se encontró la cadena de caracteres "klospad.pdb". La extensión de este archivo ".pdb" puede ser usado en el proceso de depuración de un programa y en las interfaces de desarrollo de Microsoft Visual Studio.

Dentro de la muestra CTB Locker, se encontraron cadenas de caracteres correspondientes a funciones y librerías dentro de la sección ".rdata" del archivo binario. Dicha sección es la misma en la que se deben ubicar las funciones importadas de acuerdo con la información obtenida a partir de la cabecera PE.

Dentro de la sección ".rdata" de la muestra de CTBLocker se encontraron menciones de cinco librerías de enlace dinámico y 48 funciones correspondientes a las mismas.

A continuación, se indican las librerías a las que se hace mención en la muestra:

- KERNEL32.dll
- msimg32.dll
- WTSAPI32.dll
- SHLWAPI.dll
- nddeapi.dll

Dentro de la sección ".data" se encontraron menciones de la librería "kernel32.DLL" y de la función "VirtualAllocEx", es importante mencionar esto, ya que están en un lugar distinto que el resto de importaciones.

Un análisis más detallado de las funciones y librerías nombradas se realiza posteriormente en la sección Funciones y librerías.

Adicionalmente a las ya mencionadas, no se hallaron cadenas de caracteres que pudieran ser significantes durante este análisis.

- **Detección de empaquetamiento y ofuscación**

No hay sospechas sobre empaquetamiento dentro del presente archivo, ya que se encontró la suficiente cantidad de strings referentes a librerías y funciones.

Para realizar este proceso, se carga la muestra en cuestión en el programa PEiD, dedicado a detectar packers, y se obtiene la siguiente información. Como se indica en la Figura 3.10.

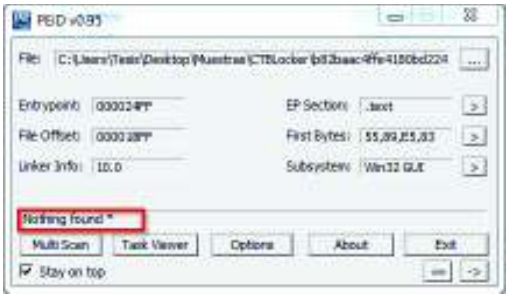

Figura 3.10. Resultado del análisis en PEiD de CTBLocker.

La Figura 3.11 indica la sección de PEiD referente a información adicional que señala estos datos:

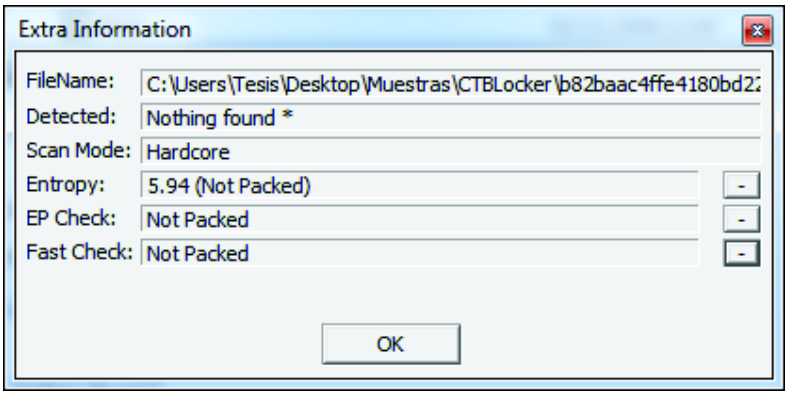

Figura 3.11. Información adicional de CTBLocker en PEiD.

Como primer punto se señala que PEiD no identificó un empaquetador, por otro lado, a pesar de la búsqueda exhaustiva que incluyó el chequeo de Ejecutable Portable, como el chequeo rápido reflejan que no existe empaquetamiento. Adicionalmente, PEiD realizó un cálculo de entropía que demuestra la inexistencia de un empaquetamiento.

Finalmente, como se indica en la Figura 3.12. se utiliza el Plugin KriptoANALizer (KANAL) para buscar firmas relacionadas con encriptación.

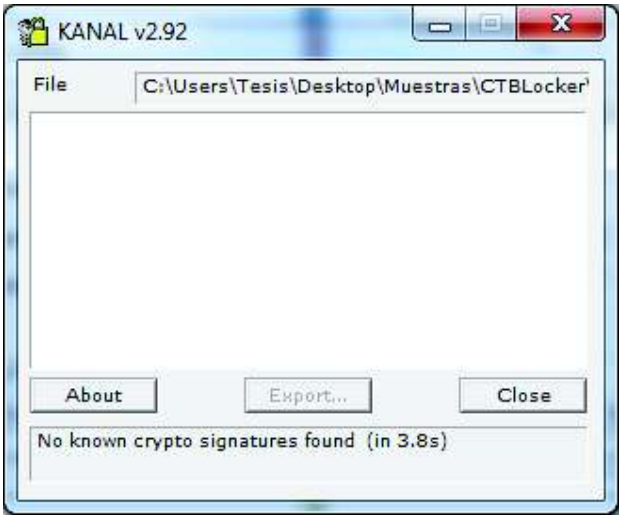

Figura 3.12. Resultado de KANAL de CTBLocker.

Tal como demuestra la Figura 3.12, no se encontró evidencia de empaquetamiento.

Después del análisis realizado se concluye por ende que la muestra en cuestión no se encuentra empaquetada.

- **Librerías y Funciones Invocadas**

Para conocer las librerías como las funciones a las que la muestra invoca antes de ser ejecutada, se utilizaron las herramientas Dependecy Walker y Dependencies.

- **Dependency Walker**

En la Figura 3.13, se observa la siguiente información de la muestra CTBLocker analizada:

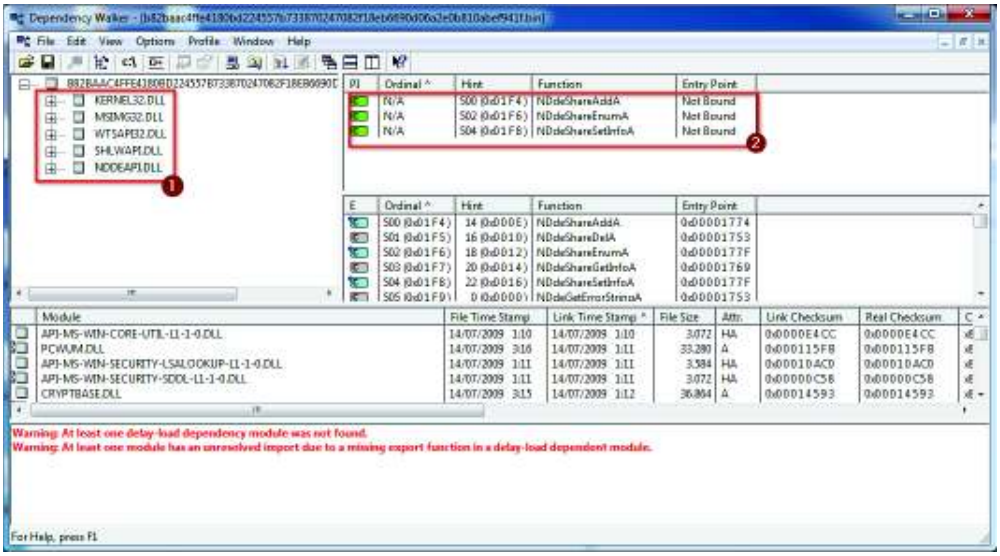

Figura 3.13. Librerías de CTBLocker detectadas en Dependency Walker.

1. Las librerías que está importando la muestra son: KERNEL32.DLL, MSIMG32.DLL, WTSAPI32.DLL, SHLWAPI.DLL y NDDEAPI.DLL.
2. Al ingresar a cada una de las librerías invocadas se observa un listado de las funciones que pueden ser importadas.

- **Dependencies**

A continuación en la Figura 3.14. se muestra la siguiente información:

1. Las librerías y la localización que la muestra invoca.
2. Las funciones que pueden ser importadas por la muestra.
3. El tipo de archivo, en este caso es DLL y de tipo ejecutable.

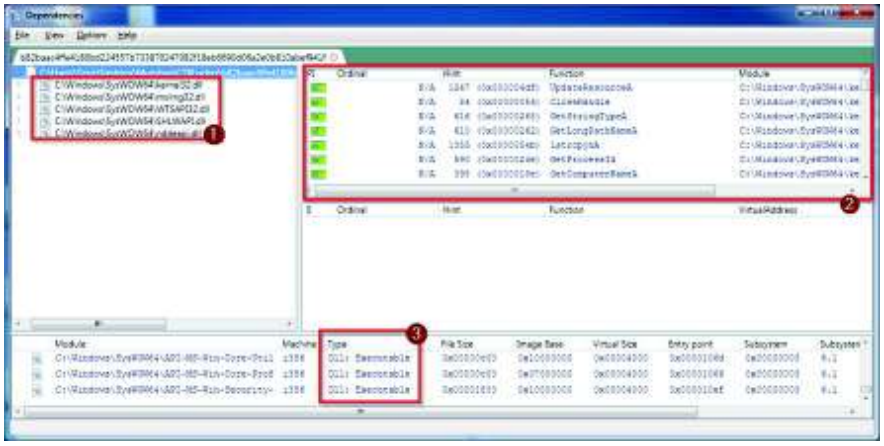

Figura 3.14. Librerías y funciones de CTBLocker detectadas en Dependencies.

Muchas veces las funciones convocadas no son utilizadas, solo cumplen con el fin tener la librería completa; cabe mencionar que, esta es dinámica y puede llamar más funciones.

Entre las funciones obtenidas, las siguientes pueden afectar el normal funcionamiento del equipo.

- **GetProcAddress**: esta función se encuentra en la librería Kernel32.dll. Además de las funciones importadas en la cabecera del archivo, esta función puede importar funciones de otras DLL para llamar a cualquier función no declarada.

- **LoadLibraryA**: es peligrosa puesto que puede invocar a una librería completa que pudo o no ser llamada al inicio del proceso. Esta función se encuentra en la librería Kernel32.dll.

- **GetCurrentProcess**: Esta función tiene acceso a todos los procesos del equipo y puede alterar cualquiera de ellos. Esta función se encuentra en la librería Kernel32.dll.

En el Anexo I se pueden observar todas las librerías y funciones convocadas.

Análisis de Locky

- **Análisis con Antivirus**
  - **Avast:**

En la Figura 3.15 se observa que la herramienta Avast detecta y controla la amenaza con el nombre de Win32:Malware-gen y cumplió la función de Escudo del sistema de archivos, que es la base de datos de virus de la herramienta.

Adicional el archivo infectado fue enviado a un baúl de virus, para que el usuario final pueda decidir si eliminar o tratar de desinfectar el archivo. La gravedad posicionada por la herramienta para Locky es baja. Toda esta información es presentada desde la aplicación instalada en el equipo.

Figura 3.15. Detección de Locky por Avast.

  - **Kaspersky:**

En la Figura 3.16 se observa que Kaspersky Internet Security detecta, aísla y elimina el malware con nombre HEUR:Trojan.Win32Generic.

En los eventos de la herramienta se detallan los 3 trabajos realizados por la misma, así como indica la ruta en la cual se encontraba la amenaza.

Al eliminar el archivo comprometido, evita que el usuario por desconocimiento libere la infección. El informe minucioso sobre los procesos realizados por Kaspersky para bloquear la infección son detallados desde la misma aplicación del antimalware.

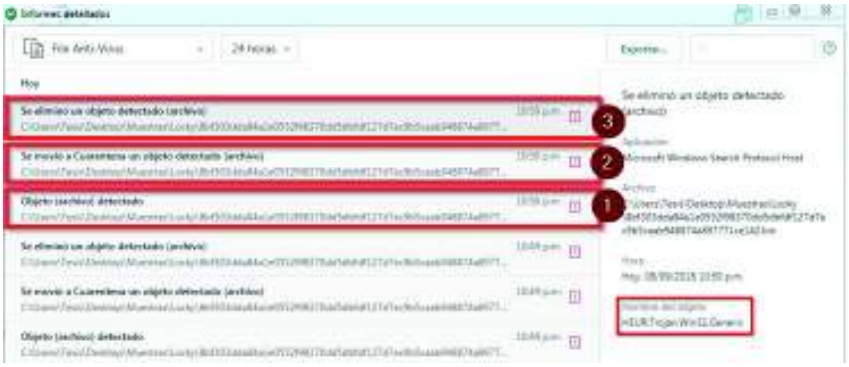

Figura 3.16. Detección de Locky por Kaspersky.

- **Sophos**

En la Figura 3.17 se observa que el malware es detectado y eliminado por Sophos, esta información se puede obtener desde el sitio web https://home.sophos.com/ donde se requiere registrarse e iniciar sesión.

En el informe se muestran todos los detalles de la detección y limpieza del equipo, así como la localización del archivo y la hora de infección. El nombre definido por Sophos es Elenoocka-E.

Figura 3.17. Detección de Locky por Sophos.

- **Hydrid Analysis**

En la Figura 3.18 se observa que el malware es detectado por Hybrid Analysis, mediante la validación de antivirus CrowdStrike, Metadefender y VirusTotal.

Figura 3.18. Detección de Locky por Hybrid Analysis.

- **Toma de Huellas de Archivo y propiedades estáticas**
  - **Toma de Huellas**

Se extrajeron las huellas de archivo correspondientes a los algoritmos MD5, SHA1 y SHA 256 para la muestra del malware Locky, los cuales se presentan a continuación en la Tabla 3.2.

Tabla 3.2. Algoritmos de malware Locky.

ALGORITMO	VALOR
MD5	C6475A9B90DCCEA03D93DEDF00EAC5EE
SHA1	B7AFBE3C25FA4A147B32FA37B71C95FF089489E9
SHA256	8BF303DDA84A1E0552F98370DD5DBFDF127D7EC9B5CAAB948874A897771CE142

- **Propiedades Estáticas**

A continuación en la Figura 3.19, se expone la información extraída de las cabeceras PE del archivo LOCKY

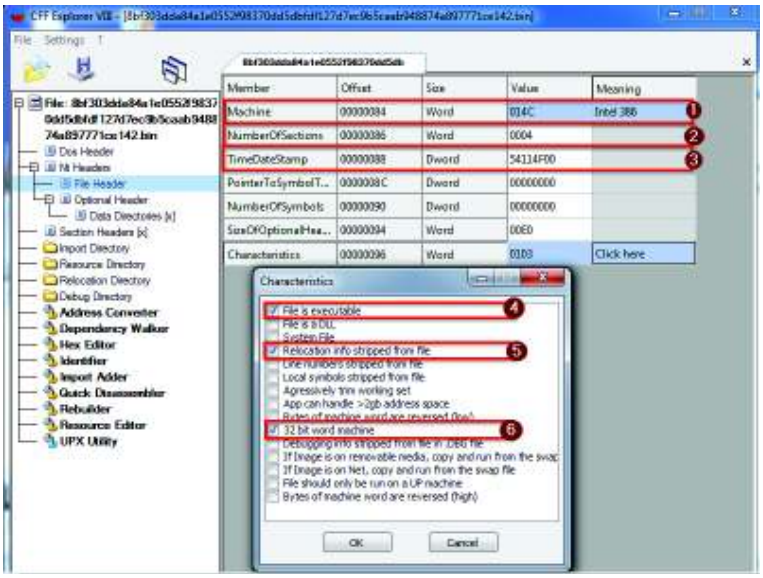

Figura 3.19. Resultado de la cabecera de Locky.

En la Figura 3.20 se indica la traducción del equivalente de la fecha de compilación de Locky.

Figura 3.20. Traducción del equivalente de la fecha de compilación de Locky.

De las cabeceras PE de archivo se puede extraer que:

1. El archivo fue diseñado para una arquitectura Intel 386, es decir que el código de ensamblaje utilizado es de 32 bits.
2. Se tiene cuatro secciones dentro del archivo.
3. Se consultó el equivalente TimeDateStamp en el sitio web Epoch Converter [40], obteniéndose que el archivo fue compilado el 11 de Septiembre del 2014, a las 02h28 hora de Ecuador.
4. El archivo es un ejecutable.

5. El archivo puede ejecutarse únicamente en su dirección preferida de memoria, en caso de que la misma se encuentre ocupada, el archivo no puede ejecutarse.
6. Utiliza palabras de 32bits.

De la cabecera opcional se extrae que:

1. Se trata de un ejecutable portable de 32 Bits.
2. El código debe empezar a ejecutarse dentro de la sección ".text".
3. Las secciones deben ubicarse en espacios de disco múltiplos de 200 Bytes.
4. El programa puede conectarse con algún tipo de aplicación remota.

Esto se indica en la Figura 3.21.

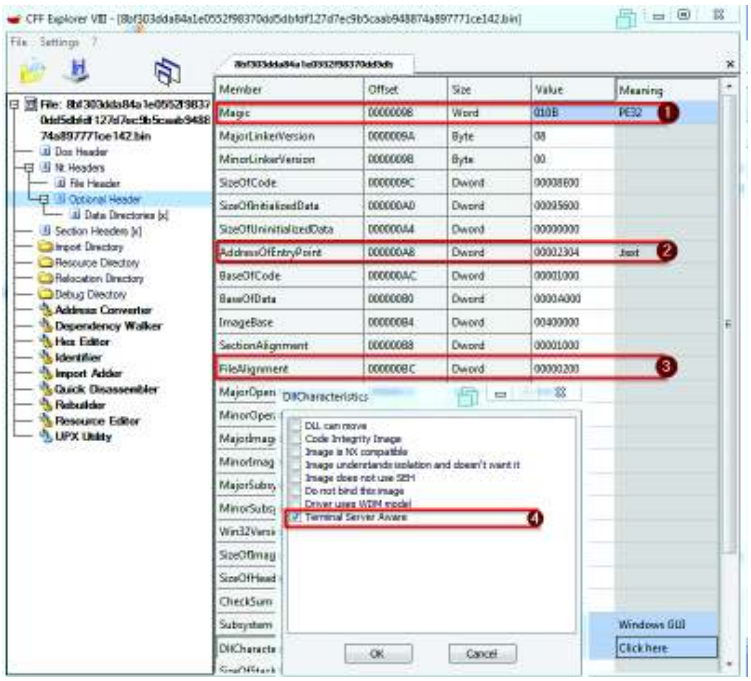

Figura 3.21. Traducción del equivalente de la fecha de compilación de Locky.

De la sección "Información de Directorios" como se puede observar en la Figura 3.22 se obtiene que:

1. El directorio de importaciones se encuentra en la sección ".rdata".
2. El directorio de relocalización simula ubicarse en la sección ".rdata", lo cual se contradice con la información encontrada previamente, ya que este software no admite relocalización.

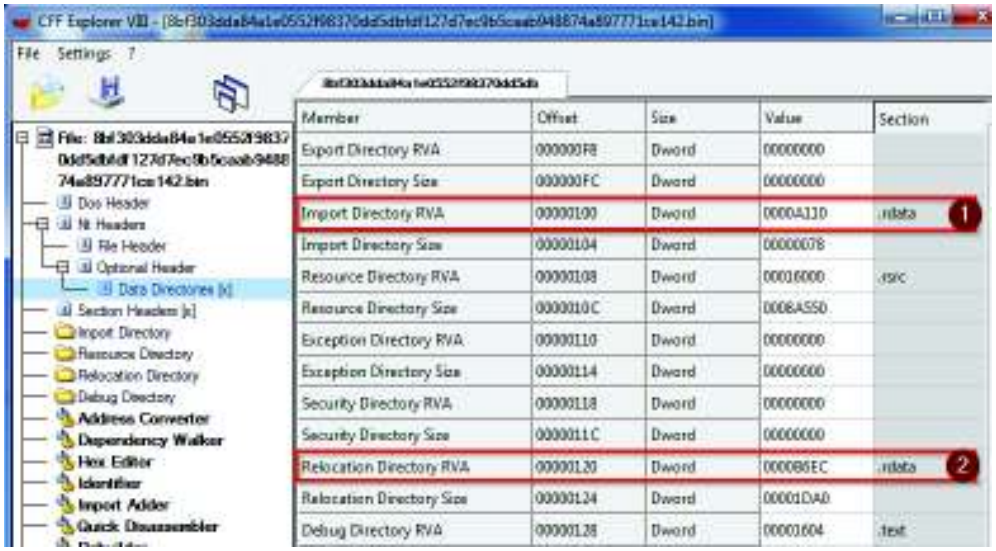

Figura 3.22. Información de cabecera de Locky.

De los encabezados de sección de la Figura 3.23. se tiene que:

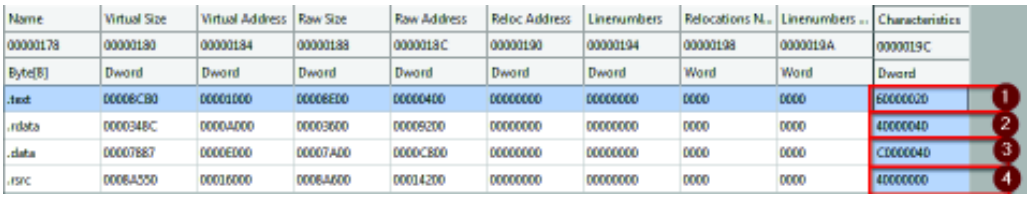

Figura 3.23. Encabezados de Locky.

1. La sección ".text" contiene código ejecutable, y tiene permisos de lectura y ejecución.
2. La sección ".rdata" contiene variables que han sido inicializadas, y tiene permiso de solo lectura.
3. La sección ".data" contiene información que ha sido inicializada, y tiene permisos de lectura y escritura.
4. La sección ".rsrc" tiene permisos de solo lectura.

- **Búsqueda de cadenas de caracteres**

A continuación se realizará un reporte sobre los Strings encontrados dentro del archivo binario de la muestra de malware Locky.

Para el análisis de cadenas de caracteres se utilizaron en conjunto dos herramientas: el editar hexadecimal File Insight y el programa Strings, parte del paquete de programas Sysinternals.

Gracias a Strings se extrajeron las cadenas de caracteres presentes en toda la muestra binaria. Estas muestras binarias fueron exportadas a un archivo de texto.

Posteriormente, de las secuencias de caracteres extraídas se seleccionaron las que tenían un significado válido como: Nombres de funciones y librerías correspondientes al Windows API, instrucciones correspondientes a lenguajes de programación, nombres de archivos, URLs y direcciones IP, entre otras. Finalmente se utilizó FileInsight para clasificar las secciones de archivo EP dentro de las cuales estaban ubicadas las cadenas de caracteres para de esta manera estimar su posible uso dentro del programa.

A continuación se realizó un reporte sobre el resultado del análisis de las cadenas de caracteres encontradas.

Dentro de la muestra Locky, se encontraron cadenas de caracteres correspondientes a funciones y librerías dentro de la sección ".rdata" del archivo binario. Dicha sección es la misma en la que se deben ubicar las funciones importadas de acuerdo con la información obtenida a partir de la cabecera PE.

Dentro de la sección ".rdata" de la muestra de CTBLocker se encontraron menciones de cinco librerías de enlace dinámico y 48 funciones correspondientes a las mismas.

A continuación, se indican las librerías y sus funciones correspondientes a las que se hace mención en la muestra:

- mprapi.dll
- kernel32.dll
- advapi32.dll
- shell32.dll
- user32.dll

Dentro de la sección ".data" se encontraron las siguientes cadenas de caracteres:

- exrsvr32.dll.et7sjs7
- eaab__o_es_Memory
- hbkke__2_dll
- gkatu___lloc
- tvgcwakykxhqlfn

En primer lugar, se encuentra un supuesto archivo de librería dinámica llamado "exrsvr32.dll", que no coincide con nombres de librerías conocidas. Su nombre se aproxima a Regsvr32 que sirve para manejo de registros.

Se puede observar que varias de las cadenas de caracteres presentes a continuación pueden corresponder a funciones ofuscadas, no obstante, se puede presumir que al iniciarse el programa podría utilizar estas funciones para realizar manejo de memoria.
Un análisis más detallado de las funciones y librerías nombradas se realiza posteriormente en la sección Funciones y librerías.

Adicionalmente a las ya mencionadas, no se hallaron cadenas de caracteres que pudieran ser significantes en lo correspondiente al análisis.

- **Detección de empaquetamiento y ofuscación**

No se tiene sospechas sobre empaquetamiento dentro del presente archivo, debido a que se encontró buena cantidad de strings referentes a librerías y funciones.

Se carga la muestra en cuestión en el programa PEiD, dedicado a detectar packers, y muestra la siguiente información, como se indica en la Figura 3.24.

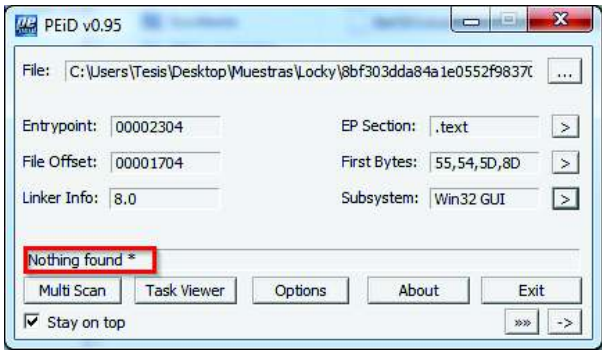

Figura 3.24. Resultado del análisis en PEiD de Locky.

En la Figura 3.25 se muestra que la sección de PEiD referente a información adicional muestra la siguiente información.

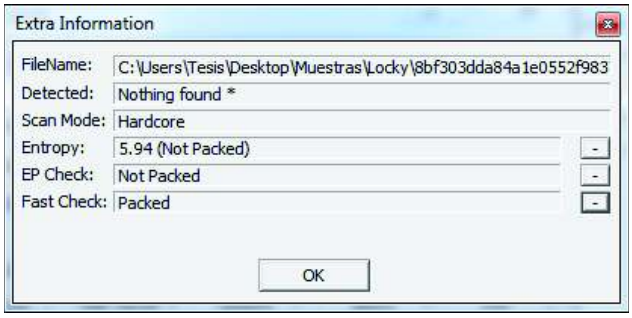

Figura 3.25. Información adicional de Locky en PEiD.

A pesar de que el análisis rápido muestre empaquetamiento, se concluye que la muestra no se encuentra empaquetada ya que no se identificó un empaquetador, aun realizando una búsqueda exhaustiva. Adicionalmente, PEiD realizó un cálculo de entropía que estima inexistente un empaquetamiento.

Finalmente como se indica en la Figura 3.26, se utiliza el Plugin KriptoANALizer (KANAL) para buscar firmas relacionadas con encriptación.

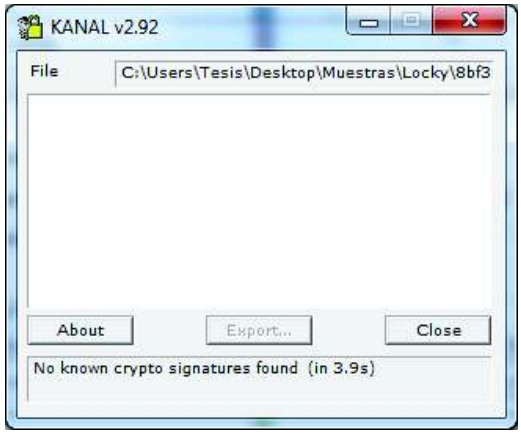

Figura 3.26. Resultado de KANAL de Locky.

Tal como se muestra en la captura de pantalla expuesta, no se encontró evidencia de empaquetamiento.

Se concluye que la muestra en cuestión no se encuentra empaquetada.

- **Librerías y Funciones Invocadas**

Para conocer las librerías como las funciones a las que la muestra invoca antes de ser ejecutada, se utilizaron las herramientas Dependecy Walker y Dependecies.

- **Dependency Walker**

En la Figura 3.27, se observa la siguiente información de la muestra Locky analizada:

- Las librerías que están importando la muestra, MPRAPI.DLL, KERNEL32.DLL, ADVAPI32.DLL, SHELL32.DLL y USER32.DLL.
- Al ingresar a cada una de las librerías convocadas se puede ver un listado de las funciones que pueden ser importadas.

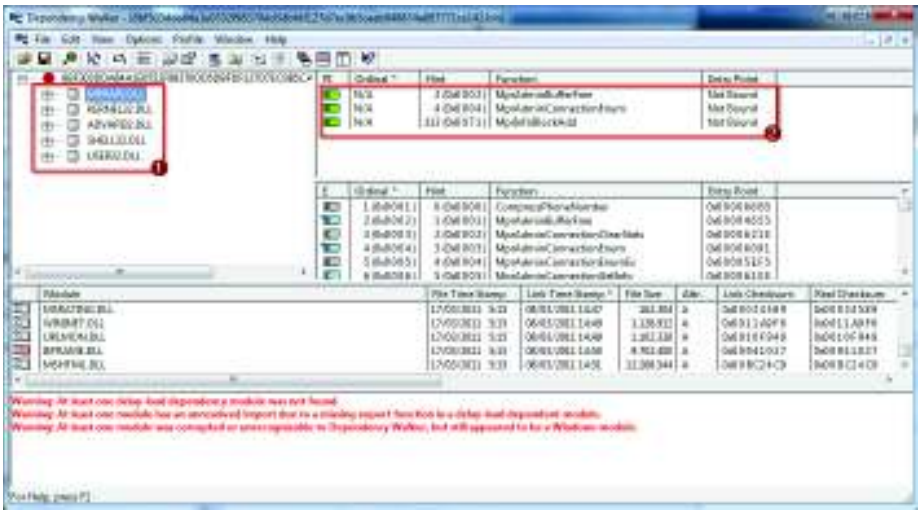

Figura 3.27. Librerías de Locky detectadas en Dependency Walker.

- **Dependencies**

A continuación, en la Figura 3.28 se muestra la siguiente información:

1. Las librerías y la localización que refleja coinciden con los que Locky invoca.
2. Las funciones que pueden ser importadas por la muestra.
3. Las acciones que pueden realizar en caso de ejecutarse.

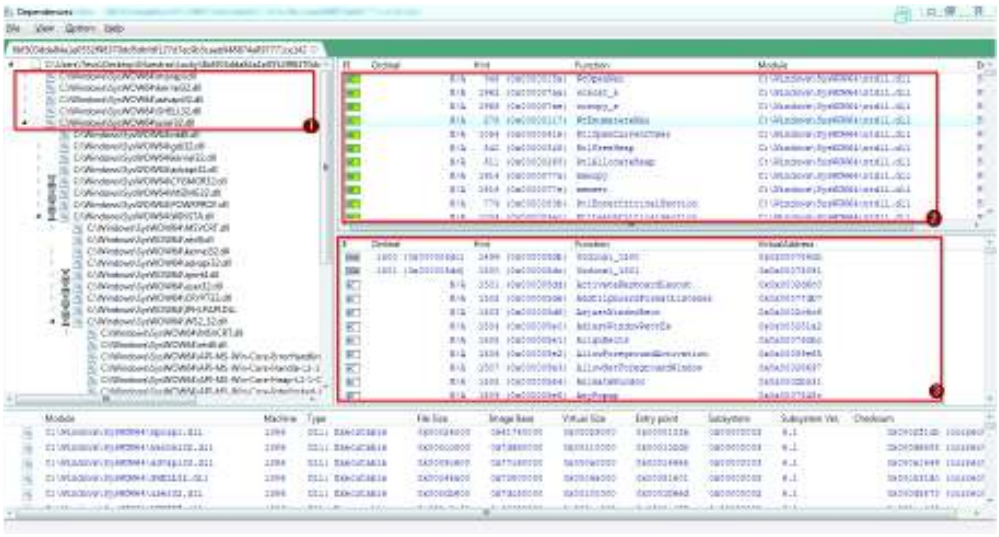

Figura 3.28. Librerías y funciones de Locky detectadas en Dependencies.

Las DLL son librerías dinámicas, lo que quiere decir que al momento de invocar una librería, no se invoca a todas las funciones que esta contenga; sino que, llama solo a ciertas de ellas, que en muchos de los casos resultan ser inofensivas; no obstante, solo basta que la librería ya se encuentre convocada para que esta pueda llamar a todas sus funciones al momento de ejecutar una muestra.

Entre las funciones obtenidas las siguientes pueden realizar cambios no deseados en el equipo:

- **CreateFileMappingA:** Se encuentra en la librería kernel32.dll. Esta función puede crear un identificador para mapear archivos que se cargan en la memoria creando accesos directos a través de las direcciones de memoria.

- **GetProcAddress:** Además de las funciones importadas en la cabecera del archivo, esta función puede importar funciones de otras DLL permitiendo que se invoque a cualquier función no declarada. Esta función se encuentra en la librería kernel32.dll.

- **ControlService:** Esta función se encuentra en la librería advapi32.dll. Se utiliza para iniciar, detener, modificar o enviar una señal a un servicio en ejecución. El malware pude utilizar esta función para manejar su propio servicio.

- **CreateDesktopW:** Esta función se encuentra en la librería user32.dll. Puede ser utilizada por el malware para crear una ventana de escritorio y ser manipulada para mostrar algún mensaje al usuario.

En el Anexo I, se pueden observar todas las librerías y funciones convocadas.

Análisis de Spora

- **Análisis con Antivirus**
  - **Avast**

En la Figura 3.29, se muestra como la amenaza fue detectada por la herramienta Avast y nombrada como Win32:Evo-gen.

La infección fue detectada por el escudo del sistema de archivos, que funciona como base de virus. Adicional muestra la dirección donde se encuentra la ubicación del archivo comprometido y la muestra enviada al Baúl de virus de Avast. La gravedad en la que la herramienta pone a Spora es media.

Figura 3.29. Detección de Spora por Avast.

- **Kaspersky**

En la Figura 3.30 se detalla como Kaspersky Internet Security, realiza los pasos de detección, aislamiento y eliminación del archivo comprometido, así mismo indica el nombre detectado por la herramienta como Trojan.Win32.Ramnit.bpr. La detección fue realizada por el Antivirus de archivos

Figura 3.30. Detección de Spora por Kaspersky.

- **Sophos**

En la Figura 3.31 se observa que el malware es detectado y eliminado por Sophos, la información se puede observar en el portal Web equipo al igual que en la localización del archivo y la hora de infección.

El nombre de esta muestra en la base de datos de Sophos es Mal/Generic-S.

Figura 3.31. Detección de Spora por Sophos.

- **Hydrid Analysis**

En la Figura 3.32 se observa como la muestra de Spora si se encuentra en la base de datos de Hybrid Analysis, y es detectada como maliciosa.

Figura 3.32. Detección de Spora por Hybrid Analysis.

- **Toma de Huellas de Archivo y propiedades estáticas**
  - **Toma de Huellas**

Se extrajeron las huellas de archivo correspondientes a los algoritmos MD5, SHA1 y SHA 256 para la muestra del malware Spora, los cuales se presentan a continuación en la Tabla 3.3.

Tabla 3.3. Algoritmos de malware Spora.

ALGORITMO	VALOR
MD5	5006B683A84A9B58315E4C5ABBDFA197
SHA1	701960BFCF4F3C5672E89ECD76C4E871DD13E690
SHA256	2E6868E2BF0EEFA5948F7A512D1ECCBC705E97559F39C 9E0F6E62378A8FDD548

- **Propiedades Estáticas**

A continuación en la Figura 3.33 se expone la información extraída de las cabeceras PE del archivo SPORA.

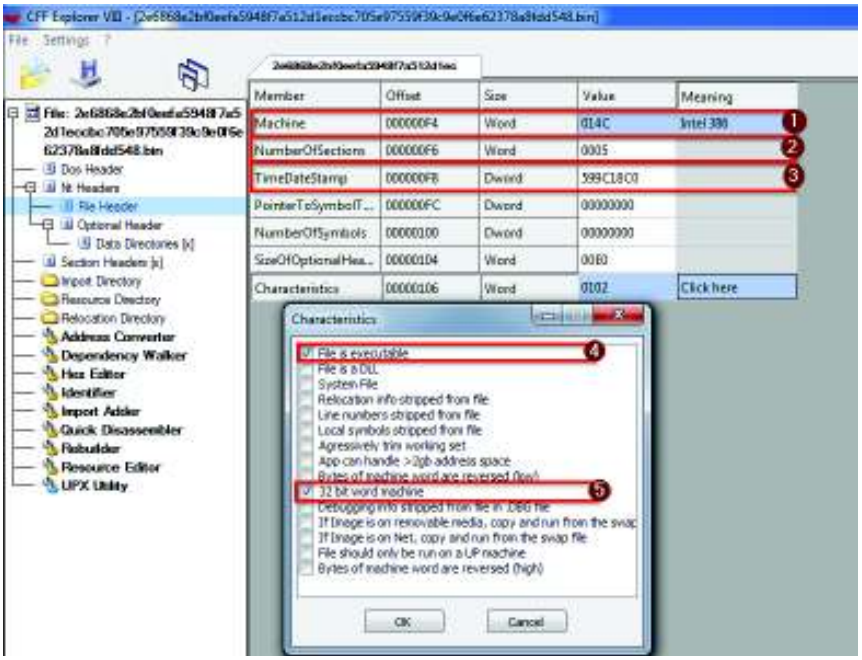

Figura 3.33. Resultado de la cabecera de Spora.

En la Figura 3.34 se observa la Traducción del equivalente de la fecha de compilación de Spora.

Figura 3.34. Traducción del equivalente de la fecha de compilación de Spora.

De las cabeceras PE de archivo se puede extraer que:

1. El archivo fue diseñado para una arquitectura Intel 386, es decir que el código de ensamblaje utilizado es de 32 bits.

2. Se tiene cinco secciones dentro del archivo.
3. Se consultó el equivalente TimeDateStamp en el sitio web Epoch Converter [40], obteniéndose que el archivo fue compilado el 22 de Agosto del 2017, a las 06h42 hora de Ecuador.
4. El archivo es un ejecutable.
5. Utiliza palabras de 32bits.

De la Figura 3.35 de la cabecera opcional se extrae que:

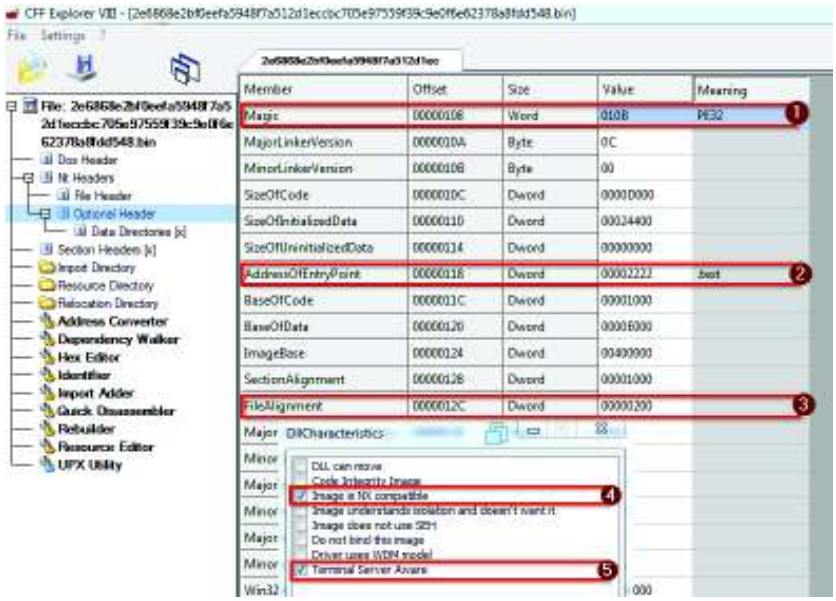

Figura 3.35. Traducción del equivalente de la fecha de compilación de Spora.

1. Se trata de un ejecutable portable de 32 Bits.
2. El código debe empezar a ejecutarse en alguna parte de la sección ".text"
3. Las secciones deben ubicarse en espacios de disco múltiplos de 200.
4. Existen partes del código que pueden ser marcadas para no ejecución.
5. El programa puede conectarse con algún tipo de aplicación remota.

De la sección información de Directorios, que se refleja en la Figura 3.35 se obtiene que:

1. El directorio de importaciones se encuentra en la sección ".rdata".
2. Existe una sección de relocalización (".reloc") especial para el caso en que el programa no pueda cargarse en la sección preferida de memoria, como se muestra en la Figura 3.36.

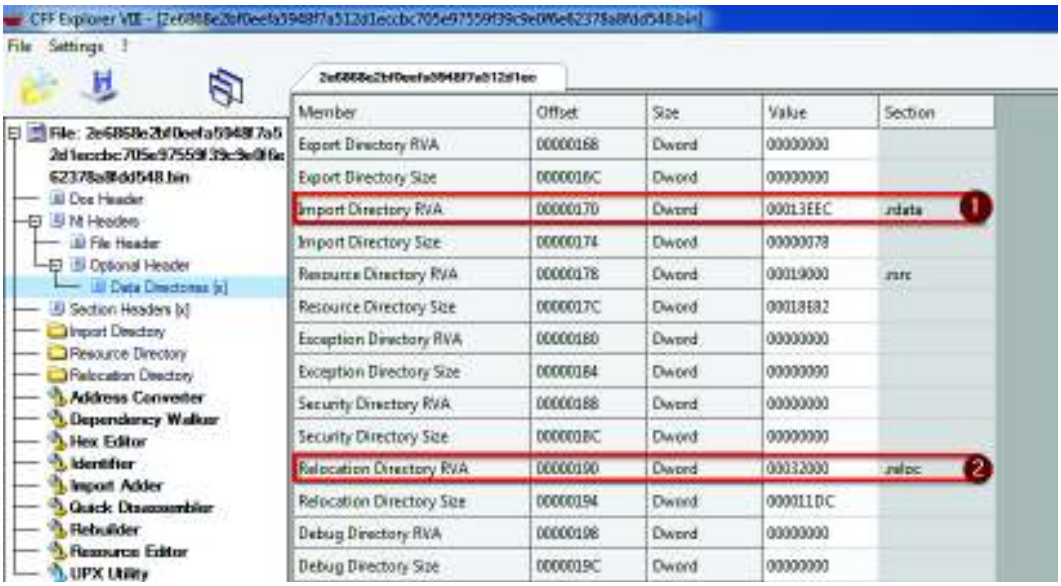

Figura 3.36. Información de cabecera de Spora.

En la Figura 3.37 de los encabezados de sección se tiene que:

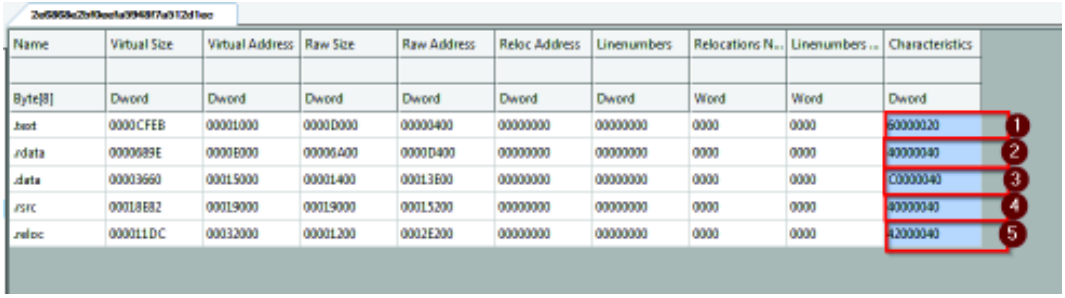

Figura 3.37. Encabezados de Spora.

1. La sección “.text” contiene código ejecutable, y tiene permisos de lectura y ejecución.
2. La sección “.rdata” contiene variables que han sido inicializadas, y tienen permisos de solo lectura.
3. La sección “.data” contiene información que ha sido inicializada, y tiene permisos de lectura y escritura.
4. La sección “.rsrc” contiene información y variables que han sido inicializadas y tiene permisos de solo lectura.
5. La sección “.reloc” contiene información y variables que han sido inicializadas, tiene permisos de solo lectura y puede descartarse una vez se ha iniciado el programa.

- **Búsqueda de cadenas de caracteres**

A continuación se realizará un reporte sobre los Strings encontrados dentro del archivo binario de la muestra de malware Spora.

Para el análisis de cadenas de caracteres se utilizaron en conjunto dos herramientas: el editar hexadecimal File Insight y el programa Strings, parte del paquete de programas Sysinternals.

Gracias a Strings se extrajeron las cadenas de caracteres presentes en toda la muestra binaria. Dichas muestras binarias fueron exportadas a un archivo de texto. Posteriormente, de las secuencias de caracteres extraídas se seleccionaron las que tenían un significado válido como: Nombres de funciones y librerías correspondientes al Windows API, instrucciones correspondientes a lenguajes de programación, nombres de archivos, URLs y direcciones IP, entre otras.

Finalmente se utilizó FileInsight para clasificar las secciones de archivo EP dentro de las que estaban ubicadas las cadenas de caracteres para estimar su posible uso dentro del programa.

A continuación se realiza un reporte correspondiente al resultado de análisis de cadenas de caracteres encontradas.

En primer lugar, se encuentran cadenas de caracteres que parecen corresponder a mensajes de error o mensajes imprimibles dentro de un programa de interacción humana. Dichos mensajes aparentemente están relacionados con operaciones de comunicaciones. Una búsqueda de dichos mensajes revela que están relacionados con errores de sistema correspondientes al lenguaje de programación C++.

Posteriormente y dentro de la misma sección se encuentran cadenas de caracteres relacionadas con funciones matemáticas, de las cuales se encontró pudieran corresponder también a C, o a C++ dentro de una librería estática para funciones matemáticas.

En la Tabla 3.4 se indican las funciones matemáticas utilizadas en Spora.

Tabla 3.4. Funciones matemáticas de Spora.

exp	Cosh	atan2	Ceil	_cabs	_y1
pow	Tanh	Sqrt	Floor	_hypot	_yn
log	Asin	Sin	Fabs	fmod	_logb
log10	acos	Cos	modf	frexp	_nextafter
sinh	atan	Tan	ldexp	_y0	

Se encuentran también una serie de mensajes impresos de manera intercalada con códigos. Una búsqueda minuciosa reveló que dichos mensajes corresponden a errores en tiempo de ejecución existentes en C, como mensajes de poco espacio o de error inesperado.

Finalmente una serie de cadenas de caracteres correspondiente a tipos de errores confirma la hipótesis de que estos mensajes corresponden a C++, tal como se puede leer en el mensaje “Microsoft Visual C++ Runtime Library”:

DOMAIN error
SING error
TLOSS error
runtime error
Runtime Error!
Program:
<program name unknown>
...
Microsoft Visual C++ Runtime Library

La presencia de todas las cadenas de caracteres expuestas previamente correspondientes al lenguaje de programación C++, y la última cadena de caracteres en que se menciona la librería para tiempo de ejecución Microsoft Visual C++, son un indicador de dependencia de instalación de dicha librería para la correcta ejecución del software malicioso.

Dentro de la muestra Locky, se encontraron cadenas de caracteres correspondientes a funciones y librerías dentro de la sección “.rdata” del archivo binario. Dicha sección es la misma en la que se deben ubicar las funciones importadas de acuerdo con la información obtenida a partir de la cabecera PE.

Dentro de la sección ".rdata" de la muestra de CTBLocker se encontraron menciones de seis librerías de enlace dinámico y 101 funciones correspondientes a las mismas. Se destaca que la librería mscoree.dll y su correspondiente función CorExitProcess no aparecieron en anteriores análisis de importaciones.

A continuación, se enlistan las librerías y sus funciones correspondientes a las que se hace mención en la muestra:

- mscoree.dll
- kernel32.dll
- USER32.DLL
- GDI32.dll
- ADVAPI32.dll
- SHELL32.dll

Dentro de la muestra Spora, se halló que todas las funciones y librerías mencionadas previamente se encontraban dentro de la sección ".rdata" del archivo binario. Esto coincide con la información obtenida en la cabecera PE correspondiente a la dirección del directorio de importaciones.

Sin embargo, es destacable que las librerías "USER32.DLL" y "kernel32.dll" se encontraron invocadas en dos ocasiones y en espacios no consecutivos de memoria.

Un análisis más detallado de las funciones y librerías nombradas se realiza posteriormente en la sección Funciones y librerías.

Posteriormente se encuentran cadenas de caracteres correspondientes con códigos de idioma definidos por la Internet Engineering Task Force. Estas pueden referirse a configuraciones regionales y de idiomas presentes en la configuración de Windows, incluso posibilitan que la detección de dicha configuración sea necesaria para el malware pueda elegir los mensajes a indicarse a través de una interfaz de usuario.

A continuación, en la tabla 3.5 se indica una muestra de los 328 idiomas encontrados.

Tabla 3.5. Ejemplos de idiomas de Spora.

CADENAS	IDIOMA
en-us	Inglés - Estados Unidos
es-ar	Español - Argentina
ca-ES	Catalán - España
ar-lb	Árabe - Lebanon
It-It	Italiano - Italia
pt-PT	Portugués - Portugal

Posteriormente se encuentran más cadenas de caracteres correspondientes al lenguaje de programación C++ relacionados principalmente con manejo de vectores.

Adicional se muestran cadenas correspondientes con manejo de fecha, hora, días de la semana, que utiliza un formato de fecha correspondiente al idioma inglés.

- **Detección de empaquetamiento y ofuscación**

No se tiene sospechas sobre empaquetamiento dentro del presente archivo, ya que se encontró buena cantidad de strings referentes a librerías y funciones.

Se carga la muestra en cuestión en el programa PEiD, dedicado a detectar packers, y se obtiene la siguiente información, que se indica en la Figura 3.38.

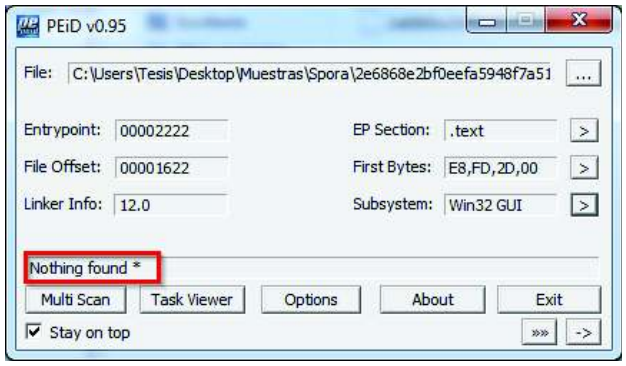

Figura 3.38. Resultado del análisis en PEiD de Spora.

La sección de PEiD referente a información adicional muestra la siguiente información, que se muestra en la Figura 3.39.

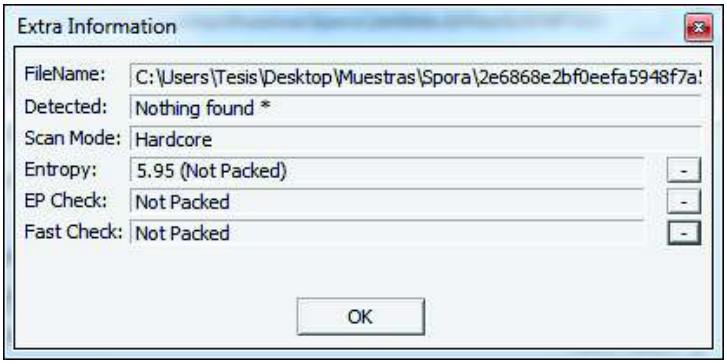

Figura 3.39. Información adicional de Spora en PEiD.

Como primera instancia se observa que PEiD no identificó un empaquetador, por otro lado, a pesar de la búsqueda exhaustiva, tanto el chequeo de Ejecutable Portable, como el chequeo rápido muestran que no existe empaquetamiento.

Adicionalmente, PEiD realizó un cálculo de entropía que estima inexistente un empaquetamiento.
Finalmente, se utiliza el Plugin KriptoANALizer (KANAL) para buscar firmas relacionadas con encriptación, como se indica en la Figura 3.40.

Figura 3.40. Resultado de KANAL de Spora.

Tal como se muestra en la captura de pantalla expuesta, no se encontró evidencia de empaquetamiento.

Se concluye por ende que la muestra en cuestión no se encuentra empaquetada.

- **Librerías y Funciones Invocadas**

  - **Dependency Walker**

En la Figura 3.41, se pueden observar la siguiente información de la muestra Spora analizada:

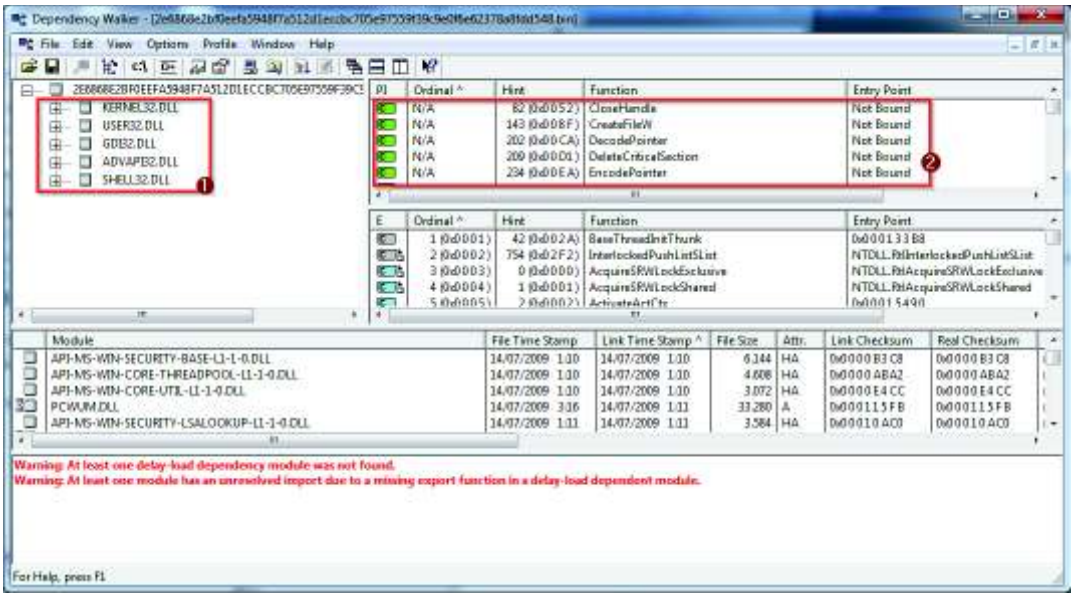

Figura 3.41. Librerías de Spora detectadas en Dependency Walker.

1. Las librerías que está importando la muestra, KERNEL32.DLL, USER32.DLL, GDI32.DLL, ADVAPI32.DLL y SHELL32.DLL.
2. Al ingresar a cada una de las librerías convocadas se puede ver un listado de las funciones que pueden ser importadas.

  - **Dependencies**

A continuación en la Figura 3.42. se muestra la siguiente información:

1. Las librerías y la localización de las mismas coinciden con los que está llamando la muestra Spora.
2. Las funciones que pueden ser importadas por la muestra.
3. Las acciones que pueden realizar en caso de ejecutarse.

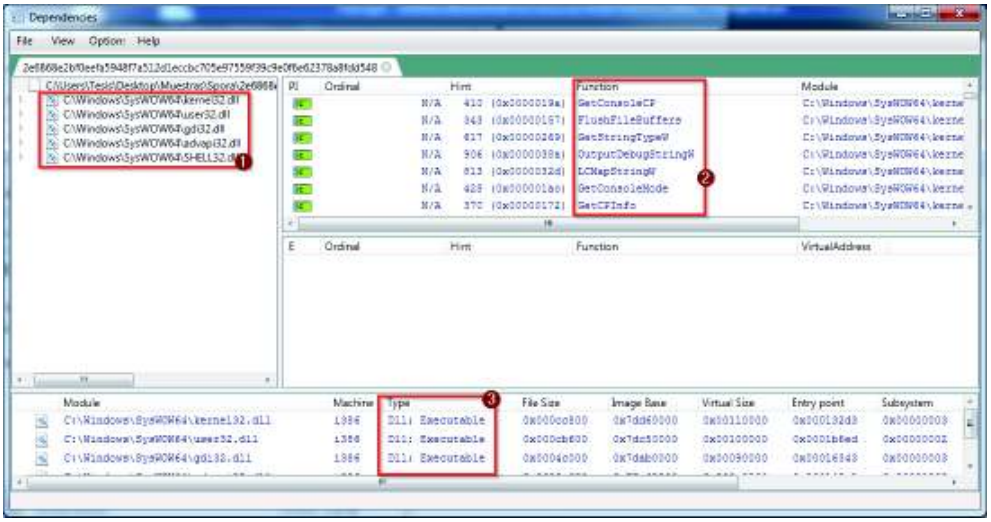

Figura 3.42. Librerías y funciones de Spora detectadas en Dependencies.

Entre las funciones obtenidas las siguientes pueden realizar cambios no deseados en el equipo:

- **CreateFileW**: Esta función se encuentra en la librería kernel32.dll. Esta función permitiría al malware crear archivos o abrir uno ya existente.

- **ClearEventLogA**: Se encuentra en la librería kernel32.dll. Permitiría al malware borrar todas las entradas de los registros de eventos específicos del equipo.

- **MapVirtualKeyA**: Esta función concedería al malware tener el registro de teclas. Esta función se encuentra en la librería user32.dll.

En el Anexo I se pueden observar todas las librerías y funciones convocadas.

Análisis de WannaCry

- **Análisis con Antivirus**
  - **Avast**

En la Figura 3.43 se observa que Avast detecta el archivo infectado y lo mueve al Baúl de virus. La herramienta tiene almacenada a la muestra analizada con el nombre de Win32:WannaCry-A[Trj]. La gravedad con la que tiene posicionada a esta muestra es baja.

Figura 3.43. Detección de WannCry por Avast.

- **Kaspersky**

En la Figura 3.44 se detallan las acciones de detección, aislamiento y eliminación del elemento infectado por parte de Kaspersky Internet Security, que nombra a la muestra como Trojan-Ransom.Win32.Wanna.zbu.

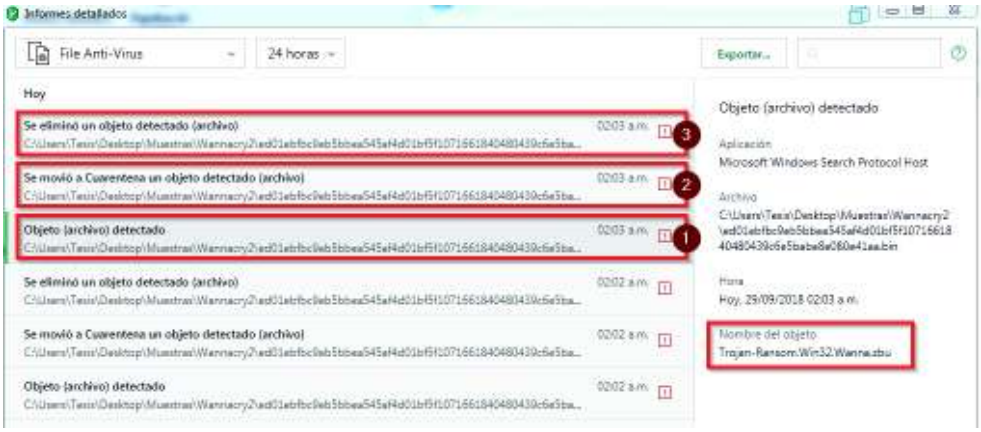

Figura 3.44. Detección de WannaCry por Kaspersky.

- **Sophos**

En la Figura 3.45, se puede evidenciar que Wannacy es detectado y eliminado por Sophos, en la administración de la herramienta se muestran todos los detalles de la detección y limpieza del equipo, así como la localización del archivo y la hora de infección. El nombre definido por Sophos es Troj/Ransom-EMG.

Figura 3.45. Detección de WannaCry por Sophos.

- **Hydrid Analysis**

En la Figura 3.46, se muestra que la muestra de WannaCry se encuentra en la base de Hybrid Analysis, validando con CrowsStrike.

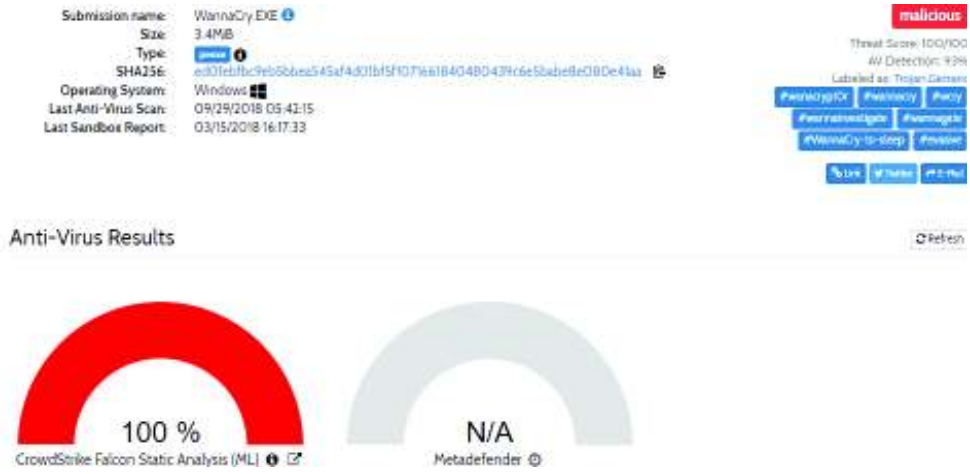

Figura 3.46. Detección de Wannacy por Hybrid Analysis.

- **Toma de Huellas de Archivo y propiedades estáticas**
  - **Toma de Huellas**

Se extrajeron las huellas de archivo correspondientes a los algoritmos MD5, SHA1 y SHA 256 para la muestra del malware WannaCry, los cuales se presentan a continuación en la Tabla 3.6.

Tabla 3.6. Algoritmos de malware WannaCry.

ALGORITMO	VALOR
MD5	FFB6BFEC7180F6B9188F351012361F53
SHA1	73D18F9191D42719B3382C0FC9830F1908CFEE39
SHA256	944825DD93BAEEFABCEC5B61D8DF5C806319E615E8C6E2DD F3D35C903229D97F

  - **Propiedades Estáticas**

A continuación se expone la información extraída de las cabeceras PE del archivo Wannacry.

Como se muestra en la Figura 3.47, de las cabeceras PE de archivo se puede extraer que:

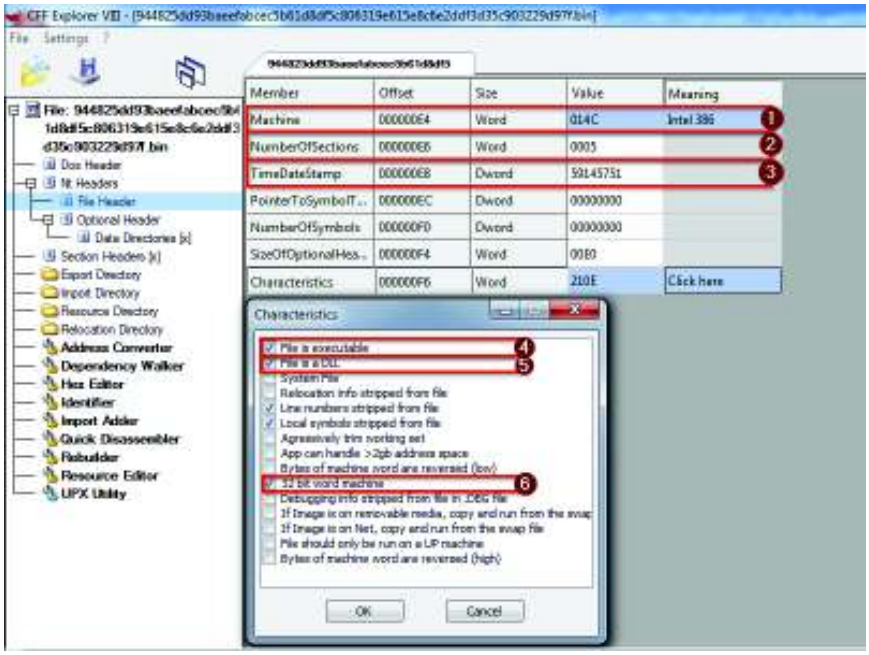

Figura 3.47. Resultado de la cabecera de WannaCry.

1. El archivo fue diseñado para una arquitectura Intel 386, es decir que el código de ensamblaje utilizado es de 32 bits.
2. Se tiene cinco secciones dentro del archivo.
3. Se consultó el equivalente TimeDateStamp en el sitio web Epoch Converter [40], obteniéndose que el archivo fue compilado el 4 de Agosto del 2014, a las 23h27 hora de Ecuador.
4. El archivo es un ejecutable.
5. Utiliza palabras de 32bits.

Esto se aprecia en la Figura 3.48:

Figura 3.48. Traducción del equivalente de la fecha de compilación de Wannacry.

Como se indica en la Figura 3.49, de la cabecera opcional se extrae que:

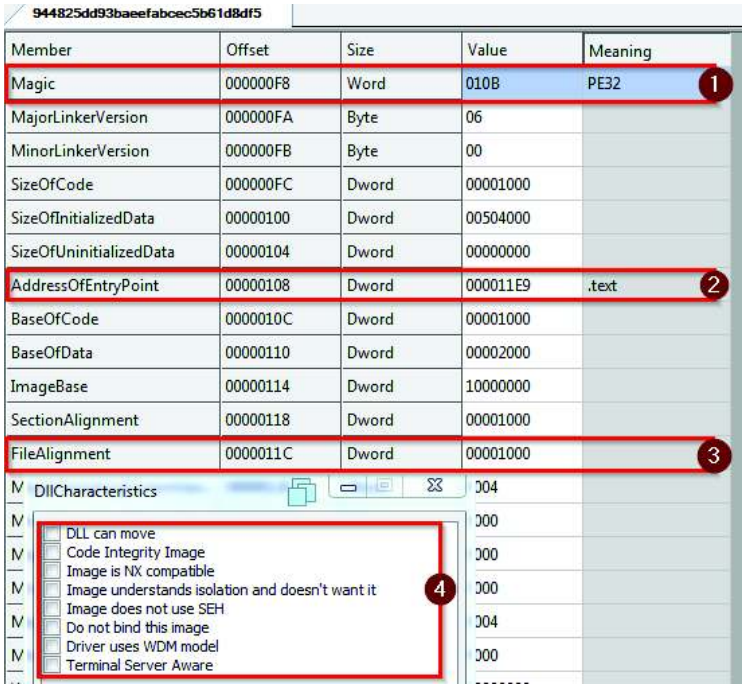

Figura 3.49. Traducción del equivalente de la fecha de compilación de Wannacry.

1. Se trata de un ejecutable portable de 32 Bits.
2. El código debe empezar a ejecutarse dentro de la sección “.text”.
3. Las secciones deben ubicarse en espacios de disco múltiplos de 1000 Bytes.
4. Es destacable que ninguna de las características DLL se encuentre marcada, ya que es poco común.

Como se indica en la Figura 3.50, de la sección información de Directorios se obtiene que:

1. Los directorios de importaciones y exportaciones se encuentran en la sección “.rdata”.
2. Existe una sección de relocalización (“.reloc”) especial para el caso en que el programa no pueda cargarse en la sección preferida de memoria, sin embargo en la cabecera de características DLL el archivo no admite ser cargado en otro espacio de memoria que no sea el preferido.

Figura 3.50. Información de cabecera de WannaCry.

En la Figura 3.51. se observa que de los encabezados de sección se tiene que:

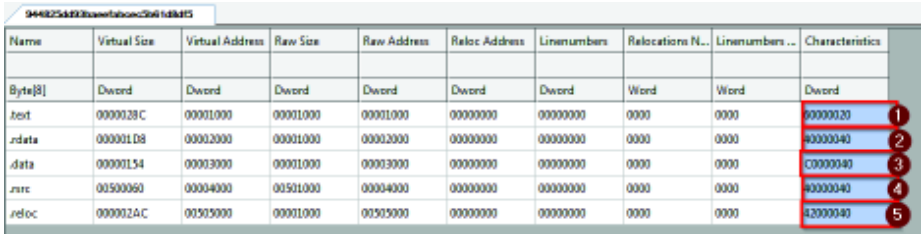

Figura 3.51. Encabezados de WannaCry.

1. La sección “.text” contiene código ejecutable, y tiene permisos de lectura y ejecución.
2. La sección “.rdata” contiene variables que han sido inicializadas, y tiene permiso de solo lectura.
3. La sección “.data” contiene información que ha sido inicializada, y tiene permisos de lectura y escritura.
4. La sección “.rsrc” contiene información y variables que han sido inicializadas y tiene permisos de solo lectura.
5. La sección “.reloc” contiene información y variables que han sido inicializadas, tiene permisos de solo lectura y puede descartarse una vez que se ha iniciado el programa.

- **Búsqueda de cadenas de caracteres**

A continuación se realiza un reporte sobre los Strings encontrados dentro del archivo binario de la muestra de malware Wannacry.

Para el análisis de cadenas de caracteres se utilizaron dos herramientas:

- El Editor hexadecimal File Insight
- El programa Strings, parte del paquete de programas Sysinternals.

Gracias a Strings se extrajeron las cadenas de caracteres presentes en la muestra binaria. Dicha muestra fue exportada a un archivo de texto. Posteriormente, de las secuencias de caracteres extraídas se seleccionaron las que tenían un significado válido para el estudio técnico realizado, como: nombres de funciones y librerías correspondientes al Windows API, instrucciones correspondientes a lenguajes de programación, nombres de archivos, URLs, direcciones IP, entre otras.

Finalmente se utilizó FileInsight para clasificar las secciones de archivo EP dentro de las cuales se ubicaban las cadenas de caracteres para estimar su posible uso dentro del programa.

A continuación se realiza un reporte correspondiente al resultado de análisis de cadenas de caracteres encontradas.

En primer lugar, se encuentran cadenas de caracteres relacionados con derechos de autor aparentemente de software, la primera (“inflate”) corresponde a una función existente dentro de la librería de compresión “zlib” desarrollada por Jean-loup Gailly y Mark Adler.

Específicamente la función “inflate” sirve para descomprimir información. La siguiente cadena de caracteres hace referencia a la función unzip, existente dentro del proyecto Minizip y desarrollada por Gilles Vollant, para la descompresión de archivos.

Dentro de la descripción del archivo en github [40] se hace mención del trabajo conjunto de esta función con otra llamada crypt y desarrollada por Terry Thorsen. Esta información resulta de gran utilidad si el código estuviese ofuscado.

A continuación, se enlistan las cadenas de caracteres mencionadas previamente.

- inflate 1.1.3 Copyright 1995-1998 Mark Adler
- -unzip 0.15 Copyright 1998 Gilles Vollant

Dentro de la muestra Wannacry, se encontraron cadenas de caracteres correspondientes a funciones y librerías dentro de la sección “.rdata” del archivo binario. Dicha sección es la misma en la que se ubican las funciones importadas de acuerdo con la información obtenida a partir de la cabecera PE.

Dentro de la sección “.rdata” de la muestra de Wannacry se encontraron menciones de siete librerías de enlace dinámico y 114 funciones correspondientes a las mismas. A continuación, se enlistan las librerías a las que se hace mención en la muestra:

Se encontraron las siguientes librerías y sus funciones correspondientes:

- Kernell32.dll
- USER32.DLL
- ADVAPI32.DLL
- MSVCRT.DLL
- SHELL32.dll
- OLEAUT32.dll
- WS2_32.dll

Por otro lado, dentro de la sección ".data" se hallaron menciones adicionales de funciones y librerías. Se encuentran algunas funciones que ya habían sido invocadas en la sección ".rdata", sin embargo se encuentran funciones adicionales capaces de manipular archivos.

A continuación, se hace mención de algunas de dichas cadenas de caracteres:

- CloseHandle
- DeleteFileW
- MoveFileExW
- MoveFileW
- ReadFile
- WriteFile
- CreateFileW
- kernel32.dll

Un análisis más detallado de las funciones y librerías nombradas se realiza posteriormente en la sección Funciones y librerías.

A continuación, dentro de la sección ".data" se encuentran las cadenas de caracteres "c.wnry", "WanaCrypt0r" y "Software\", las cuales indican archivos y carpetas. La primera cadena haría mención a un archivo con extensión "wnry" que podría ser una extensión para archivos de uso de este malware. Se sospecha que la cadena de caracteres "Software\" indica una carpeta a crearse durante la ejecución del software.

Posteriormente, pueden encontrarse cadenas de caracteres coincidentes con extensiones de tipos de archivos, unos comunes y otros especializados. Se presume que este listado de extensiones coincide con los archivos que el malware encripta.

En este conjunto de extensiones se encuentra correspondencia con una amplia variedad de tipos de archivos, que van desde imágenes y ofimática, a aplicaciones más específicas como archivos de diseños de Autocad, o archivos relacionados con diseño 3D siendo un total de 143 extensiones encontradas.

Se analiza a continuación la mención de cadenas relacionadas con manejo criptográfico de archivos. La presencia de la cadena de caracteres "RSA2" indica que el malware podría utilizar el protocolo de encriptación RSA versión 2.

El resto de las cadenas de caracteres mencionadas hacen referencia al uso de funciones de la librería estática wincrypt.h, que provee funciones criptográficas.

A continuación, se indican las cadenas de caracteres previamente mencionadas.

- RSA2
- Microsoft Enhanced RSA y AES Cryptographic Provider
- CryptGenKey
- CryptDecrypt
- CryptEncrypt
- CryptDestroyKey
- CryptImportKey
- CryptAcquireContextA

Dentro de la misma sección, se encuentran cadenas de caracteres relacionadas con archivos y carpetas. Al parecer corresponden a cadenas de caracteres en lenguajes de programación tal como C en donde %s puede cambiarse por una variable tipo cadena de caracteres y %d por una variable de tipo entero.

Por su parte “cmd.exe /c "%s" se utiliza para correr un programa dentro del símbolo del sistema de Windows. Se pueden encontrar también un programa y un comando cuyos nombres indicarían su utilidad en cuanto a programación de tareas, y finalmente “t.wnry” que pudiera ser un archivo creado por el virus.

Finalmente, dentro de la sección “.data” se encuentran strings relacionadas con errores, y se halla también menciones de lo que podrían ser funciones destinadas a crear excepciones, además de posibles mensajes de error, al realizar búsquedas relacionadas con varias de estas cadenas de caracteres los principales resultados están relacionados con java, lo que indica dentro de la muestra existe un programa desarrollado para este entorno.

Dentro de la sección “.rsrc” se hallaron varias cadenas de caracteres que estarían relacionadas con archivos contenidos dentro de la muestra con extensiones del tipo “wnry”. Muchos de ellos tienen nombres de idiomas, y simulan estar contenidos dentro de una carpeta dedicada a almacenar mensajes a mostrarse dentro de un entorno.

Se encuentran finalmente cadenas de caracteres que pudieran estar relacionadas con obtener el nombre del equipo, manejo de particiones, la versión del sistema operativo, además de lo que sería código XML del cual una parte menciona versiones de sistema operativo en las que puede ejecutarse algún programa.

Es importante mencionar la posibilidad de que la sección ".rsrc" contenga varios programas insertados dentro de la muestra de malware.

- **Detección de empaquetamiento y ofuscación**

No se tiene sospechas sobre empaquetamiento dentro del presente archivo, ya que se encontró buena cantidad de strings referentes a librerías y funciones.

Se carga la muestra en cuestión en el programa PEiD, dedicado a detectar packers, y se obtiene la siguiente información que se indica en la Figura 3.52.

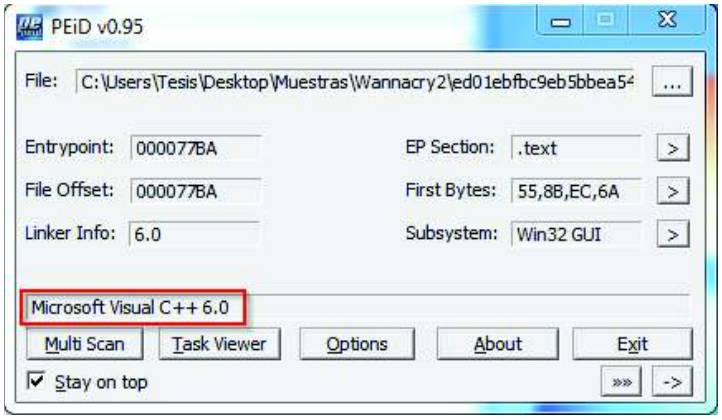

Figura 3.52. Resultado del análisis en PEiD de WannaCry.

Como primera instancia se observa que PEiD identificó la presencia de un empaquetador bajo el nombre Microsoft Visual C++ 6.0 . PEiD realizó una búsqueda en modo normal y arrojó que no se encontraron evidencias de empaquetamiento dentro de las cabeceras EP, mientras que la opción de chequeo rápido advierte de la presencia de un empaquetamiento.

La sección de PEiD referente a información adicional muestra la siguiente información, como se muestra en la Figura 3.53.

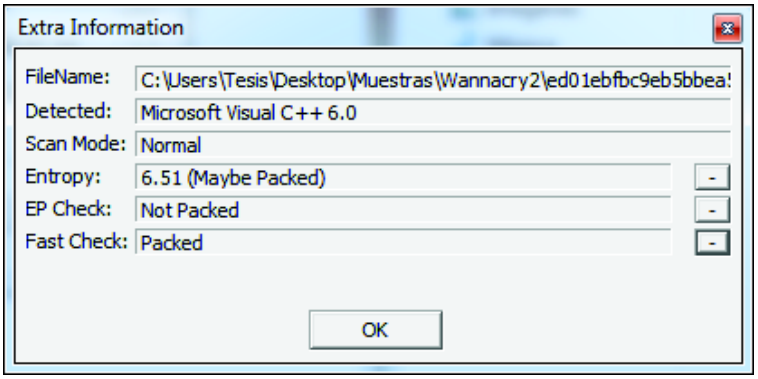

Figura 3.53. Información adicional de WannaCry en PEiD.

Adicionalmente, PEiD realizó un cálculo de entropía, mediante la cual estima como probable la existencia de un empaquetamiento. Finalmente, se utiliza el Plugin KriptoANALizer (KANAL) para buscar firmas relacionadas con encriptación, como se indica en la Figura 3.54

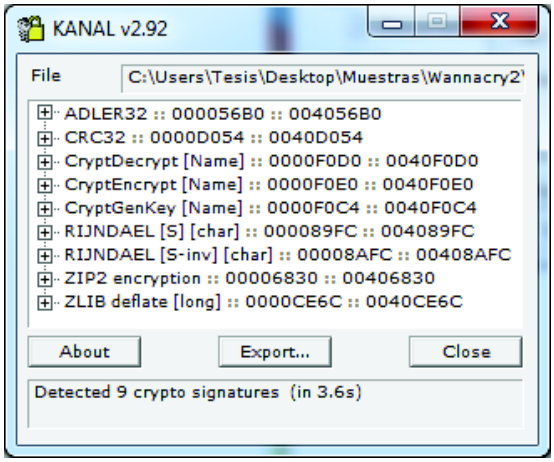

Figura 3.54. Resultado de KANAL de WannaCry.

Tal como se muestra en la Figura 3.54, se encontraron 9 firmas de empaquetadores dentro del archivo de la muestra. A continuación en la Tabla 3.7. se enlistan los detalles mostrados por KANAL referentes a los hallazgos referidos para cada empaquetador.

Tabla 3.7. Empaquetadores encontrados en WannaCry.

EMPAQUETADOR	DESCRIPCIÓN
ADLER32	Compresión ZLIB
CRC32	Transformación de bytes
CRYPTDECRYPT	Función de Microsoft que se encarga de descifrar datos encriptados
CRYPTENCRYPT	Función de Microsoft que se encarga de encriptar datos
CRYPTGENKEY	Función de Microsoft que genera un clave aleatoria
RIJNDAEL [S]	Algorithm de encripción AES
RIJNDAEL [S-inv]	Algorithm de encripción AES donde la tabla de encripción es inversa a RIJNDAEL [S]
ZIP2 encryption	Sistema de encripción utilizado por ZIP 2.0
ZLIB deflate	Algoritmo de compresión: valores basados en códigos literales, utilizados para construir los árboles

Se concluye por ende que la muestra en cuestión se encuentra empaquetada. A continuación se procederá a desempaquetar la muestra de malware para obtener información adicional sobre la misma dentro del análisis estático.

Con el propósito de obtener una muestra desempaquetada se utiliza el plugin "PEiD Generic Unpacker" que forma parte de la herramienta PEiD. Se muestra a continuación en la Figura 3.55 el menú del software PEiD para acceder al plugin mencionado

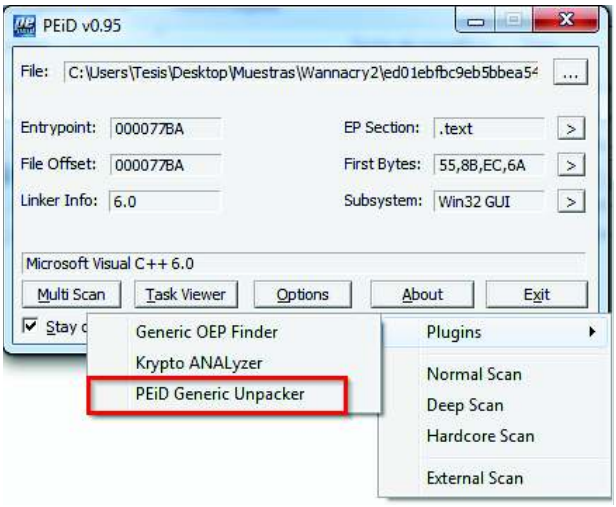

Figura 3.55. Plugin de desempaquetamiento para WannaCry.

Sin embargo, al dar click en el botón “OEP Detected” (Detectar el punto de entrad original) esto desencadena en la ejecución e instalación del Malware Wannacry, por lo que este método de desempaquetamiento se descarta.

Se muestra a continuación en la Figura 3.56 la captura del malware generando archivos en la carpeta en la cual se encontraba la muestra.

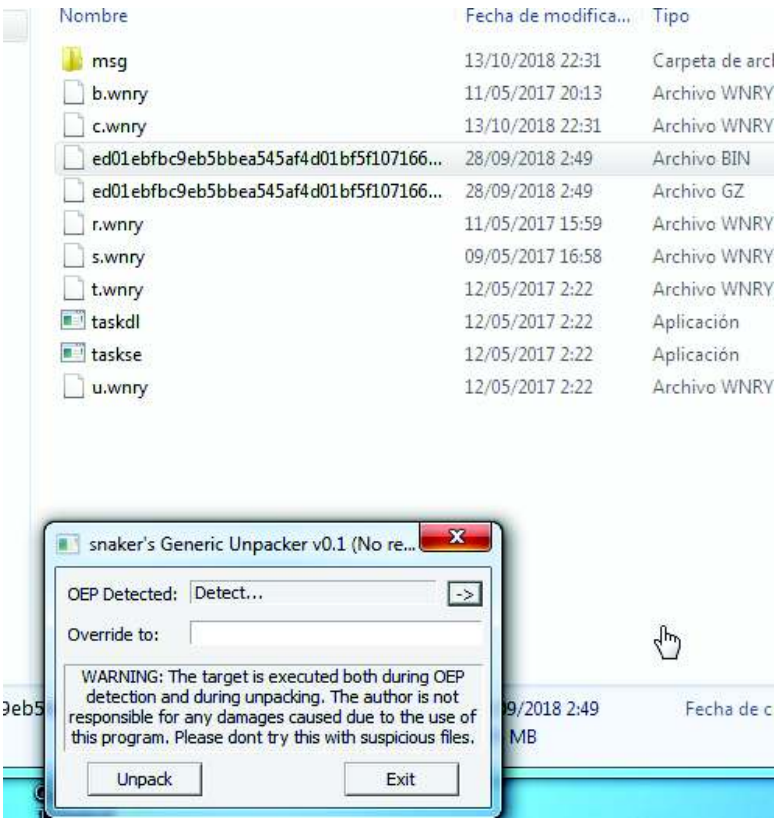

Figura 3.56. Archivos generados por WannaCry.

Debido a los problemas causados al intentar desempaquetar el malware, y la cantidad de información obtenida dentro de otros procesos de análisis, se decide continuar el análisis del comportamiento del malware mediante análisis dinámico en miras de conocer más sobre el mismo.

- **Librerías y Funciones Invocadas**

Para conocer tanto las librerías como también las funciones que invoca la muestra antes de ser ejecutada, se utilizaron las herramientas Dependecy Walker y Dependecies.

- **Dependency Walker**

En la Figura 3.57, se observa la información de la muestra WannaCry analizada:

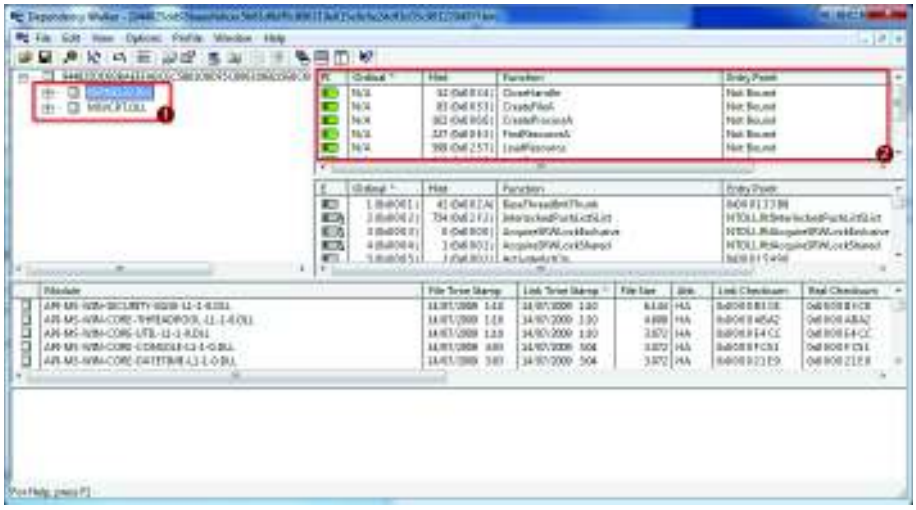

Figura 3.57. Librerías de WannaCry detectadas en Dependency Walker.

1. Las librerías que está importando la muestra son: KERNEL32.DLL, USER32.DLL, ADVAPI32.DLL y MSVCRT.DLL.
2. Al ingresar a cada una de las librerías convocadas se observa un listado de las funciones que pueden ser importadas.

- **Dependencies**

A continuación en la Figura 3.58, se muestra la siguiente información:

Figura 3.58. Librerías y funciones de WannaCry detectadas en Dependencies.

1. Las librerías y la localización coinciden con las que está llamando la muestra WannaCry.
2. Las funciones que pueden ser importadas por la muestra.
3. Las acciones que pueden realizar en caso de ejecutarse.

Entre las funciones más interesantes se pueden observar a:

- **CreateProcessA**: Esta función se encuentra en la librería kernel32.dll. Con esta función el malware podría crear nuevos procesos en el equipo infectado.

- **CreateFileA**: Se encuentra en la librería kernel32.dll. Permitiría al malware crear o modificar archivos.

- **VirtualAlloc:** Con esta función se puede asignar memoria en un proceso remoto. WannaCry podría utilizarlo para un proceso de inyección. Esta función se encuentra en la librería kernel32.dll.

- **CreateServiceA:** Permite al malware crear y eliminar servicios. Se encuentra en la librería kernel32.dll

En el Anexo I, se pueden observar todas las librerías y funciones convocadas.

3.2. Análisis Dinámico

Análisis de CtbLocker

- **Análisis Automatizado**

Al subir las muestras al portal de Threat Analyzer, se obtiene la siguiente información:

1. Entrega los hashes SHA256, SSDEP, MD5 y SHA1.
2. El número de cambios de archivos es 4673.
3. Está categorizado como un malware de tipo ransomware.
4. Se puede observar que entre los 4671 cambios realizados, se detectan amenazas inyectadas, determinando que es un archivo malicioso.
5. Se detectan 15 riesgos conocidos, 1 riesgo alto y 6 riesgos variados.

Estos resultados se aprencian en la Figura 3.59.

Figura 3.59. Análisis de CTBLocker en Threat Analyzer

En la Figura 3.60 se indica que al subir la muestra en el sitio de Hybrid Analysis se obtuvieron los siguientes resultados:

1. Un cambio notable para el usuario final es el cambio de la imagen de escritorio.
2. Lee datos como nombre del equipo, fecha de instalación del sistema operativo, cambios en el caché del navegador.
3. Realiza la petición a 4 dominios y 215 host.

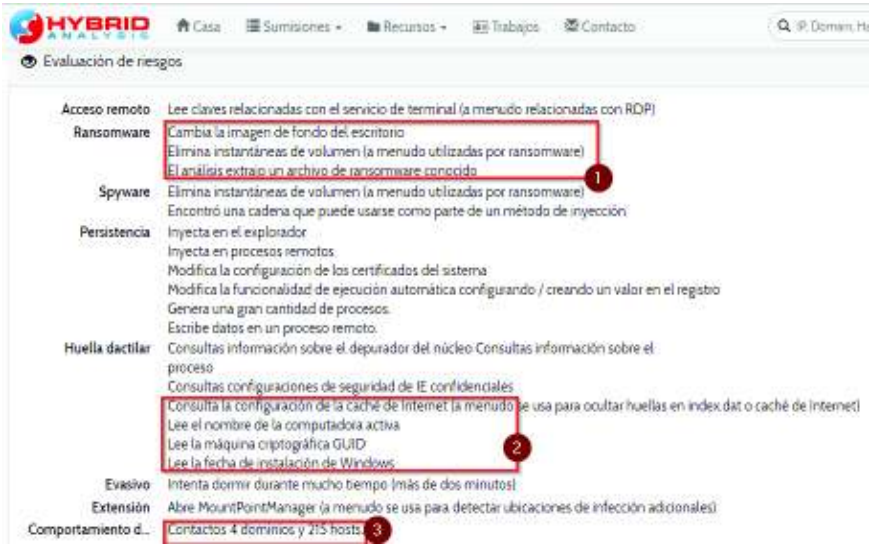

Figura 3.60. Análisis de CTBLocker en Hybrid Analysis

En la Tabla 3.8, se muestran las peticiones de DNS realizadas:

Tabla 3.8. Peticiones DNS que realiza CTBLocker..

zsn5qtrgfpu4tmpg.onion.lt
zsn5qtrgfpu4tmpg.onion.gq
www.spamhaus.org
ip.telize.com

En la Figura 3.61, se muestra un mapa de los 215 host contactados por el malware, como se puede observar la mayoría de peticiones son hacia países de Europa, tomando en cuenta que Francia es el país con más ataques, como se observa en la Tabla 3.9.

Figura 3.61. Países hacia donde realiza peticiones CTBLocker.

Tabla 3.9. Países hacia donde realiza peticiones CTBLocker.

PAIS	**PETICIONES**
Francia	47
Alemania	46
Reino Unido	23
Estados Unidos	21
Países Bajos	17
Otros	61

- **Análisis Local**

  - **AutoRuns**

Para la obtención de resultados en Autoruns se realizó y guardó un primer análisis del estado de la máquina virtual víctima del laboratorio de análisis, después se ejecutó la muestra de virus durante siete minutos, posteriormente se realizó un nuevo análisis con AutoRuns del estado de la máquina para comparar los resultados con el análisis previamente guardado.

Dicho procedimiento fue repetido tres veces con la finalidad de notar si los nombres de los procesos, registros o archivos creados por el malware cambian de nombre dependiendo de distintas circunstancias con la finalidad de ocultarse.

Dentro de los resultados, para las tres ejecuciones realizadas notó que se creó una tarea programada bajo el nombre “gdotogl” la cuál ejecuta un programa localizado en la carpeta de archivos temporales locales “c:\users\tesis\appdata\local\temp\” sin embargo el nombre del programa varió entre las ejecuciones del virus, llamándose “wybixad.exe” durante dos de las ejecuciones y “wybixadgmwlyjf” en una de ellas.

En la Figura 3.62, se observa la pantalla del análisis comparativo realizado por Autoruns.

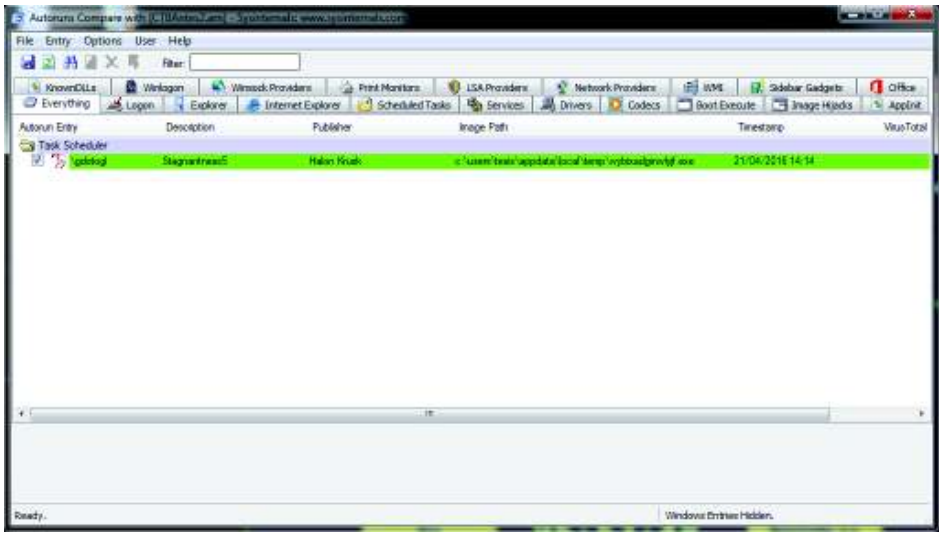

Figura 3.62. Comparación realizada por Autoruns para Locky.

Como se indica en la Figura 3.63 Autoruns permite acceder de manera directa a la pantalla de propiedades del archivo en cuestión para obtener la siguiente información sobre el mismo.

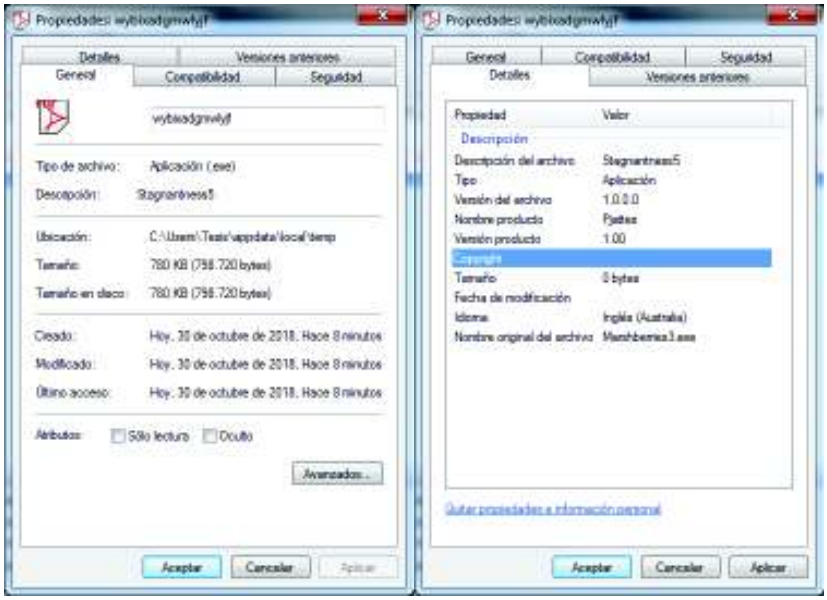

Figura 3.63. Propiedades del archivo ejecutado por CTBLocker.

Durante las distintas ejecuciones del malware se pudo observar que dentro de la pantalla de propiedades el nombre original del archivo es “Marshberries3.exe” y la descripción “Stagnantness5”, información que se mantuvo consistente durante las distintas ejecuciones realizadas.

Se procedió a inspeccionar la carpeta de ubicación del programa en cuestión y se tomó las huellas MD5, SHA1 y SHA 256 del archivo, obteniendo los resultados expresados en la Tabla 3.10:

Tabla 3.10. Huellas de CTBLocker.

MD5	77fac4194a04d2bbd9b4503044f4250c
SHA1	303EB16BB294B22F41FEF00D8EEE6ADAB6D81775
SHA256	5445EC669432BDC6C283694BBE6309F60EF574C6D1E70B2F8DF77514EF1638B0

Mismos que coinciden con la muestra original, lo que permite conocer que el programa se copió en dicha ubicación.

Se accedió al programador de tareas de Windows para observar la tarea creada por el Malware, la información se observa en la Figura 3.64.

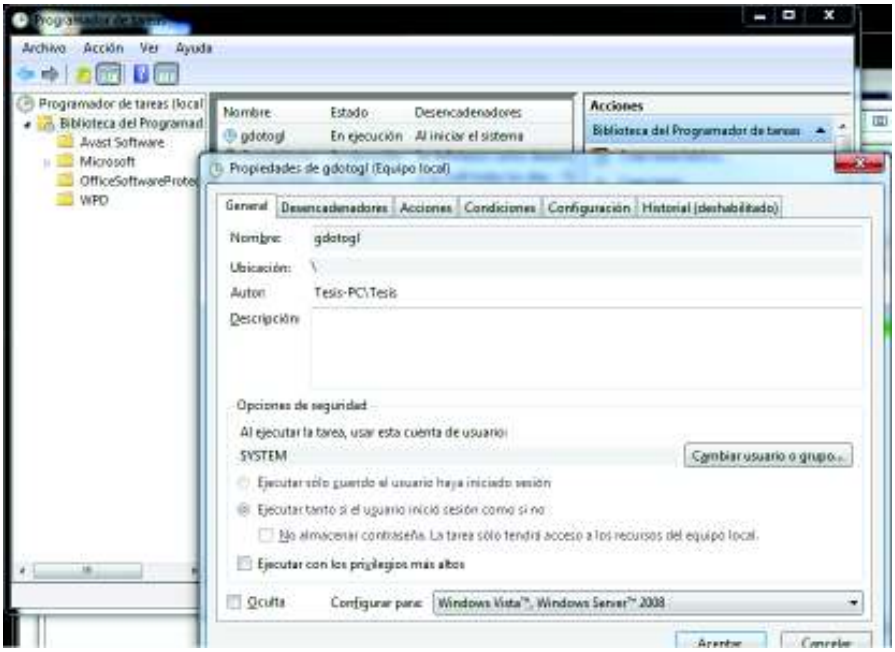

Figura 3.64. Propiedades del archivo por CTBLocker en el Programador de tareas.

De esto se puede aprender que la tarea creada para la ejecución del malware hará que el mismo se ejecute cada vez que se inicie el sistema, sin necesitar que el usuario inicie sesión. Dicha tarea está programada para evitar varias ejecuciones simultáneas de la misma.

Así mismo la tarea mantendrá el programa en ejecución durante un máximo de 3 días y se desencadenará únicamente si el equipo está funcionando conectado a la corriente alterna.

- **Regshot**

Al analizar los cambios realizados por la muestra CTBLocker con de RegShot, se observaron varios cambios significativos.

La muestra de CTBLocker realiza cambios en el registro HKU\Control Panel\Desktop\Wallpaper, que implica cambios en el fondo de pantalla del equipo.

Hace cambios en algunos de los registros contenidos en la capeta de KU\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\, en donde se realizan cambios de tamaño, ícono, vista o posición de las capetas.

Se observan que existen registros y archivos modificados que implica que se realizaron cambios en las configuraciones, caché, historial y contenido temporal del navegador Explorer, estas modificaciones se las pueden observar en:

- HKU\Software\Microsoft\Windows\CurrentVersion\Ext
- HKU\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\
- C:\Users\Tesis\AppData\Local\Microsoft\Windows\TemporaryInternetFiles\ Content.IE5\ index.dat
- HKU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012018090720180908

Se observan cambios a nivel de las tareas programadas en HKLM\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\ Schedule\ TaskCache \ Logon\

Una vez infectado el equipo se evidencia un total de 700 cambios realizados, tanto en archivos como claves.

- **Process Explorer**

Al realizar la infección con CTBLocker se monitoreó el equipo con la herramienta Process Explorer, evidenciando los siguientes cambios en el consumo de recursos:

- A nivel de CPU el equipo tuvo varios picos de 100% de uso, causando lentitud en su normal rendimiento.
- El número de Bytes de entrada y salida mostró picos durante la ejecución del malware, esto indica que existió un número inusual de procesos ejecutándose.
- El disco tuvo picos cercanos al 100% a causa de los procesos en segundo plano que CTBLocker ejecuta.

Esto se expresa en la Figura 3.65.

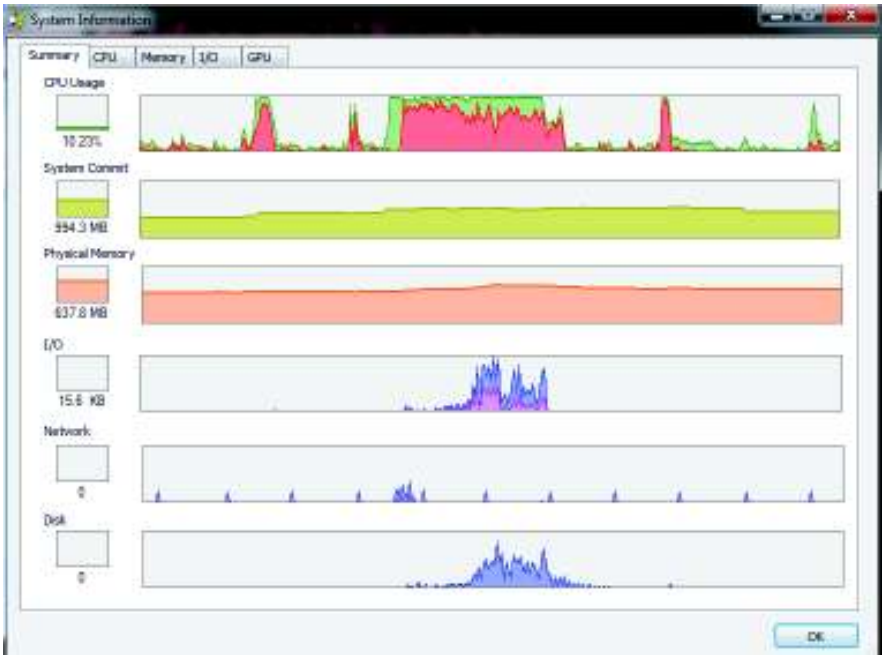

Figura 3.65. Consumo de recursos durante la infección de CTBLocker.

En la Figura 3.66, se muestran los procesos que se ejecutaron al momento de la infección indicando cuanto CPU ocupa cada de ellos, con una breve descripción del proceso.

Adicional en la parte inferior se evidencian las librerías convocadas por el malware CTBLocker.

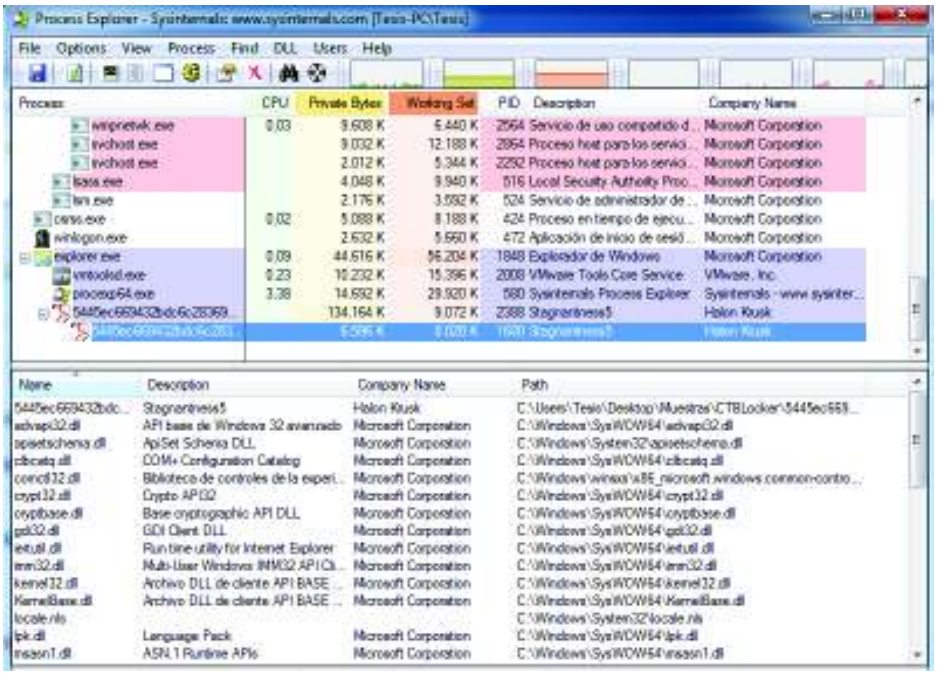

Figura 3.66. Procesos ejecutados durante la infección de CTBLocker.

Analizando el proceso levantado por el malware durante la infección se verificó que el ejecutable no tiene una firma certificada por Microsoft, indicando que es un archivo malicioso, como se puede observar en la Figura 3.67.

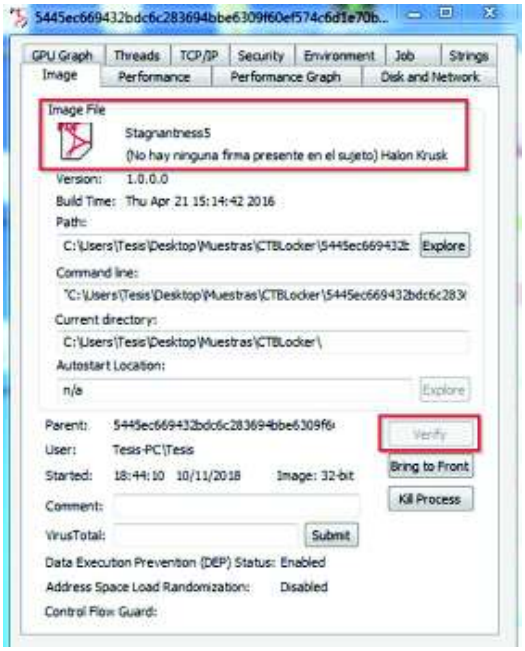

Figura 3.67. Verificación de firma de CTBLocker.

- **Process Monitor.**

Durante los experimentos de análisis dinámico se ejecutó Process Monitor para la captura de eventos del sistema y se permitió al Malware ejecutarse durante 5 minutos, antes de detener la captura con ProcMon.

De esta forma se obtuvo un número alto (en el orden de las centenas de miles) que resulta excesivo para el análisis. Fue necesario filtrar la información capturada de manera que fuera útil para el análisis en proceso.

Process Monitor permite exportar la información capturada en formato ".CSV" (Comma Separated Values), gracias al cual se puede visualizar y filtrar de manera más rápida los resultados dentro de una hoja de cálculo. Se procede entonces a observar inicialmente los resultados relacionados con el archivo original de la muestra. A continuación se resumen los resultados más importantes obtenidos de esta manera.

Primero se procede a explorar la información capturada referente al archivo original de la muestra, para observar los cambios realizados en el sistema y sus procesos asociados. Se encuentra en primer lugar en su lista de propiedades dos módulos:

- En la Figura 3.68 se observa el Árbol de procesos para “wybixad.exe”

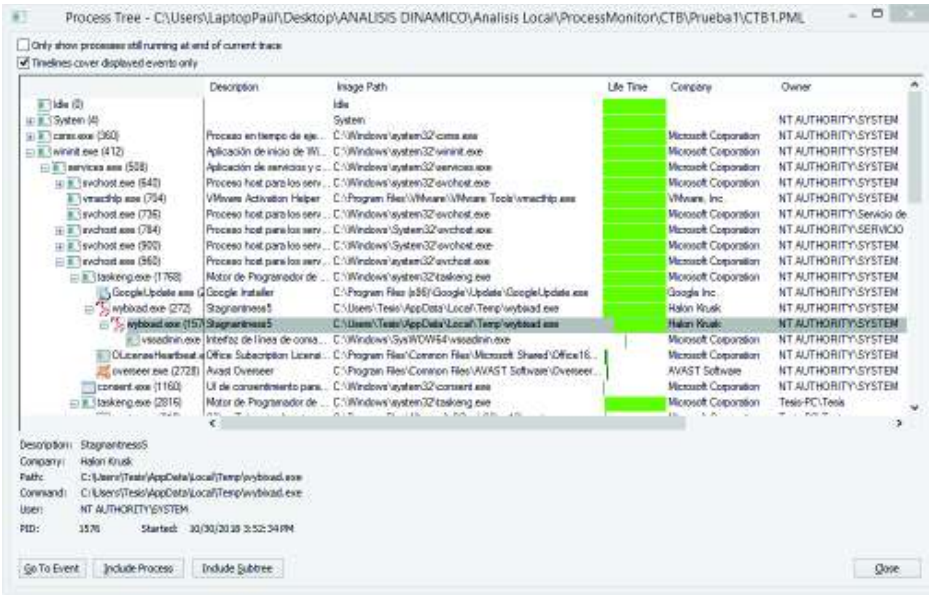

Figura 3.68. Árbol de procesos de “wybixad.exe”

- En la Figura 3.69 se indica el Árbol de procesos para la muestra original

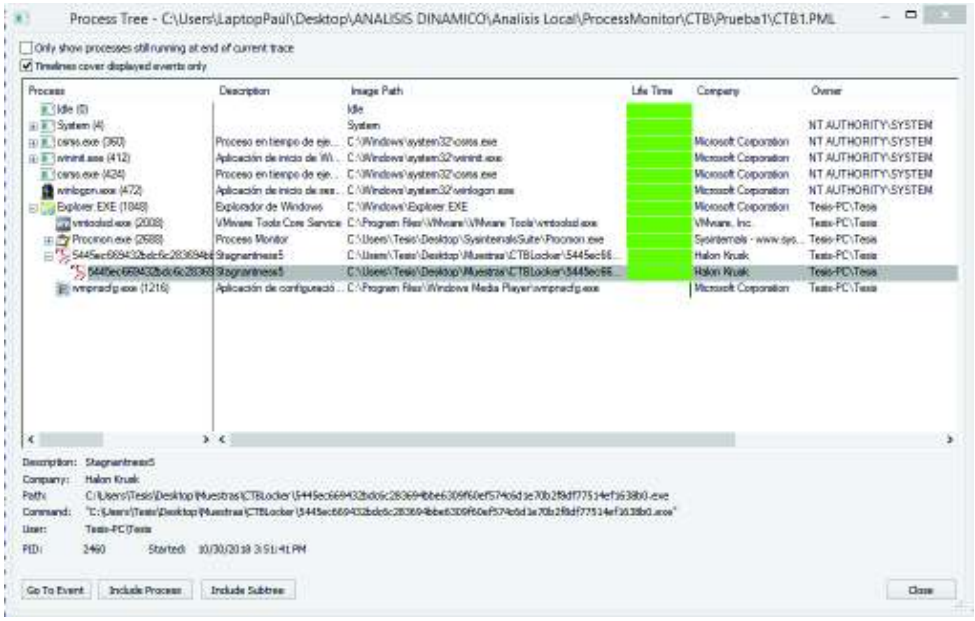

Figura 3.69. Árbol de procesos de la muestra original

Procesos creados:

- C:\Users\Tesis\AppData\Local\Temp\wybixad.exe
- C:\Windows\SysWOW64\vssadmin.exe

Aparentemente CTBLocker utiliza el proceso VSSAdmin para eliminar todos los respaldos de sistema existentes. Ya que en la descripción de Process Monitos se encuentra “PID: 712, Command line: vssadmin delete shadows all”

De entre los archivos creados es muy importante el relacionado con la creación del archivo “C:\Users\Tesis\AppData\Local\Temp\wybixad.exe” el cuál pudo observarse también dentro del resultado realizado con la herramienta Autoruns. Por lo que es importante analizar los resultados relacionados también con dicho archivo.

De entre las claves consultadas y modificadas por el malware CTB Locker, se puede inferir que podría realizar una lectura del lenguaje presente en el sistema, establecer una conexión remota y utilizar los servicios criptográficos del sistema Windows. Además para desactivar el envío de reportes de error de Windows.

Se procede a observar los resultados relacionados con el archivo “C:\Users\Tesis\AppData\Local\Temp\wybixad.exe” creado por la muestra de malware. De esto se encuentra que el virus falla en su proceso de crear los archivos:

- C:\Windows\SysWOW64\config\systemprofile\AppData\Local\Microsoft\Windows\History\History.IE5\MSHist012018103020181031\index.dat
- C:\$LogFile
- C:\ProgramData\Google\wkyajhh

Al igual que lo estudiado con RegShot se puede concluir que CTBLocker modifica la cantidad de información que puede almacenarse correspondiente a archivos de internet.

- **Resultado de la Inflexión**

Lo primero que se puede evidenciar es que el malware se muestra con un ícono simulando ser un archivo PDF, como se puede ver en la Figura 3.70.

Figura 3.70. Malware CTBLocker antes de ser ejecutado.

Una vez ejecutado el malware, demora entre 3 a 4 minutos en mostrar cambios en el equipo infectado, la primera señal de la infección es una ventana que indica que un programa dejó de funcionar, este programa tiene el mismo nombre del archivo PDF de la Figura 3.71.

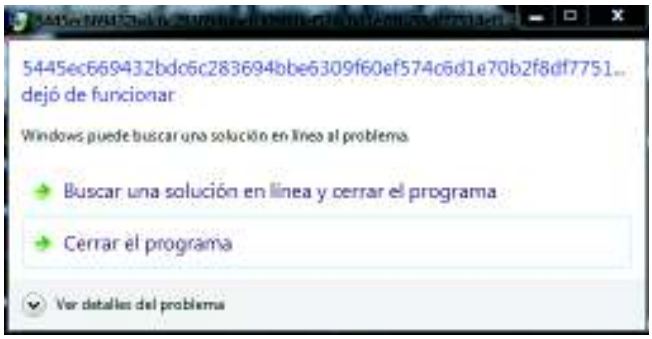

Figura 3.71. Mensaje de error en programa una vez infectado con CTBLocker.

Inmediatamente después, se evidencia una alteración más obvia, que es el cambio del fondo de pantalla del equipo infectado, como se puede observar en la Figura 3.72, en el fondo de pantalla indica que los archivos han sido cifrados y los pasos necesarios para recuperar la información.

Figura 3.72. Solicitud de recompensa con CTBLocker en fondo de pantalla.

Al revisar la reputación en Virus Total tanto del sitio web http://tmc2ybfqzgkaeilm.onion.cab así como de yhttp://tmc2ybfqzgkaeilm.tor2web.org se observa que los 2 se encuentran catalogados como sitios maliciosos por más de una solución antimalware.

Al analizar los archivos del equipo infectado, estos cambian de extensión a .ifevzjb, evitando que puedan ser recuperados o modificados. Se tomó una muestra de 68 distintos tipos de archivos con sus respectivas extensiones, de las cuales CTBLocker modificó y encriptó 23 de los archivos como se puede observar en la Figura 3.73.

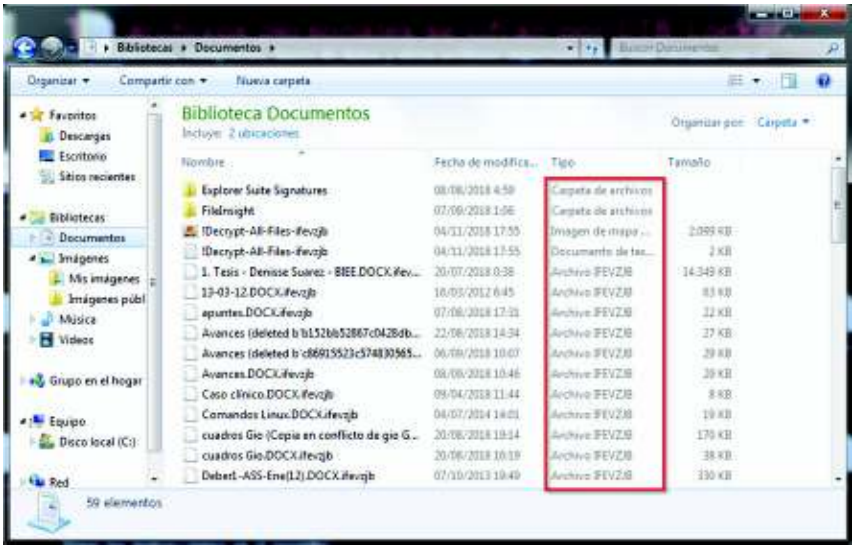

Figura 3.73. Documentos cifrados con CTBLocker.

- **Análisis de Actividad de Red**
    - **ApateDNS**

Se utilizó ApateDNS para capturar las solicitudes de DNS realizadas en el equipo local. Se fijó la dirección de respuesta a la de la máquina virtual Kali Linux de servicio con IP (192.168.117.169) y se inició el proceso de captura.

Se realizó en primera instancia una captura de solicitudes de DNS sin la ejecución de ninguna de las muestras de Ransomware, para filtrar los resultados y separarlos de los que pudieran aparecer al ejecutar una muestra de malware.

Para la muestra de CTB Locker se hallaron solicitudes de DNS para las siguientes URLs:

- Ip.telize.com
- zsn5qtrgfpu4tmpg.onion.lt
- zsn5qtrgfpu4tmpg.onion.gq

Se procedió a realizar un análisis para cada una de las URLs detectadas, que incluye una búsqueda de información los sitios a través del servicio WHOIS proporcionado por https://whois.icann.org, una búsqueda de malignidad a través de www.virustotal.com y finalmente una búsqueda de informe de malignidad dentro del sitio https://www.threatcrowd.org.

Para la URL ip.telize.com se encontró que la URL está registrada a nombre de Organización Frederic Cambus, en la ciudad de Rzeszów, Polonia. La URL se registró originalmente el 18 de junio del 2018 y se realizó una última actualización el 05 de Junio del 2018. Se hallaron 5 servidores relacionados con la URL anteriormente mencionada. Dentro de la plataforma VirusTotal, la URL fue detectada como maliciosa por 1 de los 67 motores de búsqueda. https://www.threatcrowd.org reportó que dicha URL se encuentra relacionada con 20 archivos distintos, de los cuales 12 se encuentra etiquetados como amenazas.

Para la URL zsn5qtrgfpu4tmpg.onion.lt no se hallaron reportes relacionados a WHOIS. Dentro de la plataforma VirusTotal, la URL fue detectada como maliciosa por 3 de 68 motores antimalware. Y https://www.threatcrowd.org reportó que dicha URL se encuentra relacionada con 20 archivos distintos, de los cuales 17 se encuentra etiquetados como amenazas.

Para la URL zsn5qtrgfpu4tmpg.onion.gq no se hallaron reportes relacionados a WHOIS. Dentro de la plataforma VirusTotal, la URL fue detectada como maliciosa por 1 de 68 motores antimalware. Y https://www.threatcrowd.org reportó que dicha URL se encuentra relacionada con 20 archivos distintos, de los cuales 17 se encuentran etiquetados como amenazas.

- **Wireshark**

Desde la máquina virtual de servicios se capturan paquetes provenientes de la máquina virtual víctima durante 12 minutos, lo que permitió la captura de 315 paquetes para su análisis

Es destacable un conjunto de paquetes que intentan establecer una comunicación a través del puerto 137, y utilizar el servicio NBNS para buscar la dirección web “IP.TELIZE.COM”, la misma que fue descubierta durante el análisis, como se indica en la Figura 3.74.

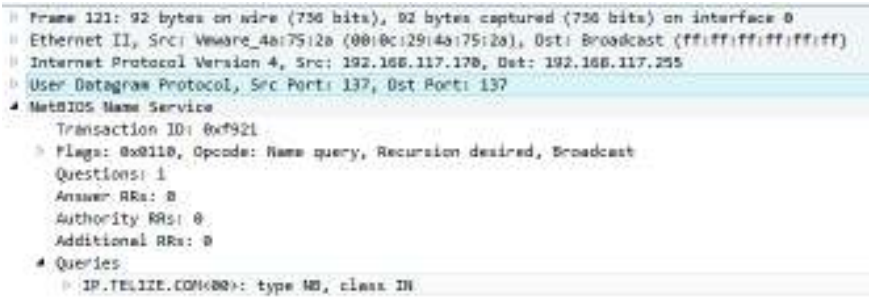

Figura 3.74. Peticiones por el puerto 137 realizadas por CTBLocker.

Posteriormente como se indica en la Figura 3.75 se observan una cadena de paquetes que intentan establecer una conexión con la dirección IP 208.83.223.34 a través del puerto 80 y con la dirección 154.35.32.5 a través del puerto 443.

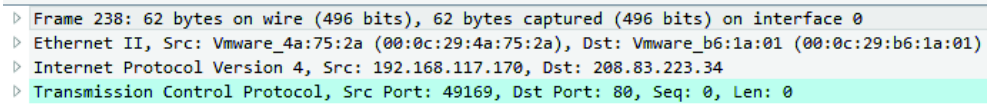

Figura 3.75. Peticiones por el puerto 80 realizadas por CTBLocker.

Se realizó un análisis de reputación de las direcciones IP mencionadas utilizando la herramienta online VirusTotal.com, en el cuál se busca el número de herramientas antivirus

que reportan la dirección IP como maliciosa con respecto al número total de búsquedas, y además la calificación que se les otorga por parte de usuarios de la herramienta Virustotal. Los resultados se muestran en la Tabla 3.11.

Tabla 3.11. Reputación de IPs destino, por parte de CTBLocker

Dirección IP	Detecciones	Calificación de la comunidad
208.83.223.34	3/67	-6
154.35.32.5	6/68	-2

- **TcpDump**

Al realizar un escaneo del puerto 137, se puede observar que este realiza peticiones a la IP 192.168.117.255, la cual es el NetBios, como se observa en la Figura 3.76.

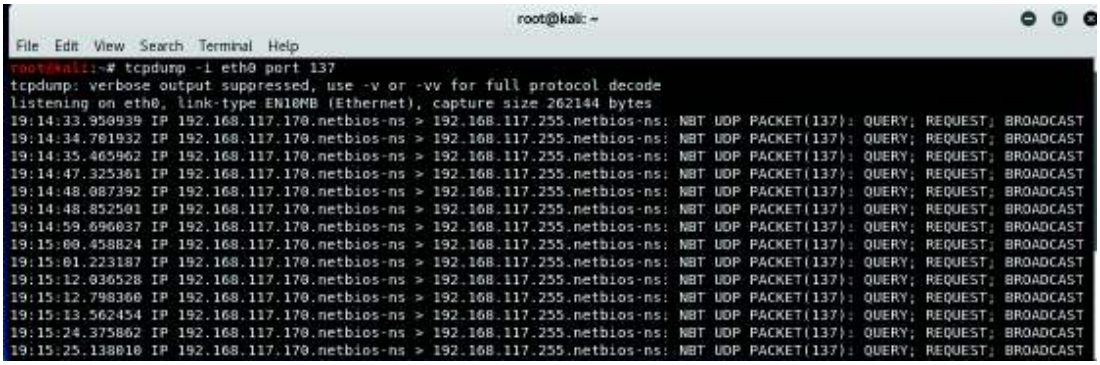

Figura 3.76. Peticiones por hacia el puerto 137 por el equipo infectado con CTBLocker

Al realizar un escaneo del puerto 80, se encuentra que realiza peticiones a la IP 208.83.223.34, la cual está catalogada como maliciosa en VirusTotal, como se puede observar en la Figura 3.77.

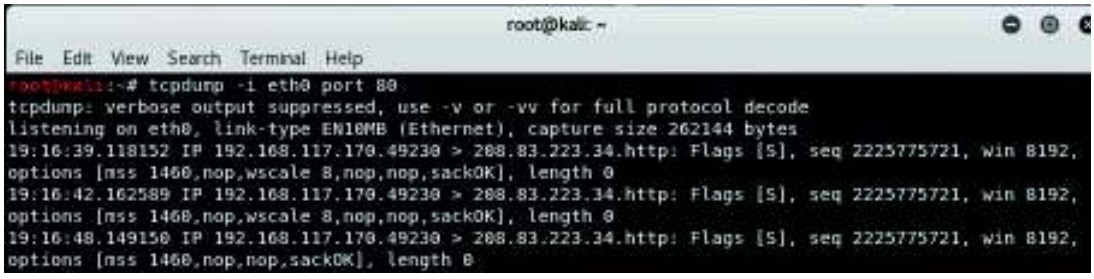

Figura 3.77. Peticiones por hacia el puerto 80 por el equipo infectado con CTBLocker

Al realizar un escaneo del puerto 443, como se puede observar en la Figura 3.78, se encuentra que realiza peticiones a las IPs 152.35.32.5 y 212.112.245.170 las cuales se encuentran como IPs maliciosas en VirusTotal.

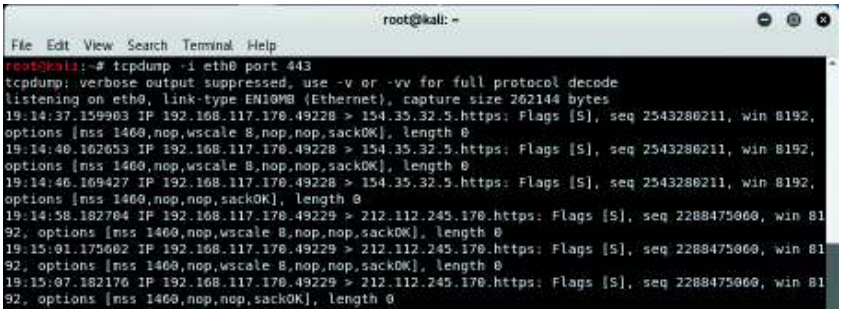

Figura 3.78. Peticiones por hacia el puerto 443 por el equipo infectado con CTBLocker

De esta manera se reafirma la información encontrada con WireShark.

Análisis de Locky

- **Análisis Automatizado**

Como se indica en la Figura 3.79. al subir las muestras al portal de Threat Analyzer, se obtiene la siguiente información:

Figura 3.79. Análisis de Locky en Hybrid Analysis

1. Entrega los hashes SHA256, SSDEP, MD5 y SHA1.
2. El número de cambios de archivos es 2500.
3. Está categorizado como un malware de tipo Ransomware de la familia Locky.
4. Se puede observar que entre los 2500 cambios realizados, se detectan amenazas de análisis de destino y archivos modificados cargados.
5. Se detectan 6 riesgos conocidos, 1 riesgo alto y 6 riesgos varios.

Al subir la muestra en el sitio de Hybrid Analysis se obtuvieron los resultados indicados en la Figura 3.80:

Figura 3.80. Análisis de Locky en Hybrid Analysis

1. Lee el nombre del equipo.
2. Realiza la petición a 1 host.

En la Figura 3.81, se muestra un mapa del host contactado por el malware, como se puede observar es Estados Unidos, el host contactado tiene la IP 8.252.65.126.

Figura 3.81. Países hacia donde realiza peticiones Locky.

- **Análisis Local**
  - **AutoRuns**

Para la obtención de resultados por parte de Autoruns se realizó y guardó un primer análisis del estado de la máquina virtual víctima del laboratorio de análisis, se ejecutó la muestra de virus durante siete minutos, se realizó un nuevo análisis con AutoRuns del estado de la máquina y se lo comparó con el análisis previamente guardado.

Dicho procedimiento fue repetido tres veces con la finalidad de notar si los nombres de los procesos, registros o archivos creados por el malware cambian de nombre dependiendo de distintas circunstancias con la finalidad de ocultarse.

Como se denota en la figura 3.82. AutoRuns no fue capaz de detectar cambios realizados por el malware Locky dentro de los aspectos analizados por dicho software de análisis

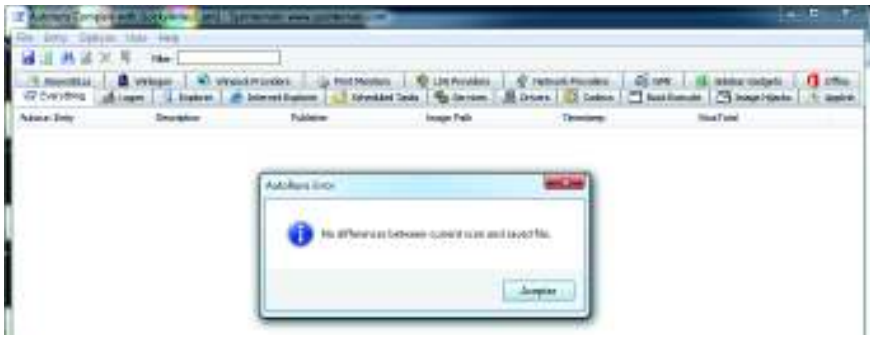

Figura 3.82. Comparación realizada por Autoruns para Locky.

- **Regshot**

Al analizar los cambios realizados por la muestra Locky con de RegShot, se observaron varios cambios significativos.

Se observan que existen registros, archivos y carpetas modificadas que implican que se realizaron cambios en las configuraciones, caché, historial y contenido temporal de los navegadores instalados en el equipo, estas modificaciones se las pueden observar en:

- HKU\ Software\Google\Chrome\BrowserExitCodes
- HKU\Software\Microsoft\Windows\CurrentVersion\Explorer
- HKU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\
- HKLM\SOFTWARE\Wow6432Node\Google\Update\ClientState\

Se observan cambios a nivel de las tareas programadas en HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Schedule\TaskCache\Tasks.

En cuanto a archivos añadidos, se detecta que existen 2 nuevos archivos en el escritorio del equipo, los cuales luego de la infección se puede observar que son los archivos tanto

de fondo de pantalla, como también un archivo HTML, que indican que el equipo se encuentra con los documentos cifrados y el proceso para recuperar la información.

- C:\Users\Tesis\Desktop\ykcol.bmp
- C:\Users\Tesis\Desktop\ykcol.htm

Se observan que se eliminan carpetas que contienen información sobre la navegación del equipo, la información es eliminada de los 2 navegadores instalados en el equipo infectado, las carpetas comprometidas son:

- C:\Users\Tesis\AppData\Local\Google\Chrome\User Data\Default\blob_storage\
- C:\Users\Tesis\AppData\Local\Microsoft\Windows\History\History.IE5\

Adicional se observan 1406 archivos eliminados, los cuales son desde documentos de Microsoft Office hasta archivos que se encontraban comprimidos, los archivos fueron tomados de varias capetas del equipo.

Hace cambios en algunos de los registros contenidos en la capeta de KU\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\, en donde se realizan cambios de tamaño, ícono, vista o posición de las capetas.

Una vez infectado el equipo se evidencia un total de 3127 cambios realizados, tanto en archivos como claves.

- **Process Explorer**

Al realizar la infección con Locky se monitoreó el equipo con la herramienta Process Explorer, evidenciando los siguientes cambios en el consumo de recursos, indicados en la Figura 3.83:

- A nivel de CPU el equipo tuvo un constante consumo llegando a ser del 100% de uso, causando lentitud en su normal rendimiento.
- El número de Bytes de entrada y salida mostró un pico durante la ejecución del malware, esto indica que existió un número inusual de procesos ejecutándose.
- El disco tuvo picos cercanos al 80% a causa de los procesos en segundo plano que Locky ejecuta.

Figura 3.83. Consumo de recursos durante la infección de Locky.

En la Figura 3.84, se muestran los procesos que se ejecutaron al momento de la infección indicando cuanto CPU ocupa cada uno de ellos, con una breve descripción del proceso.

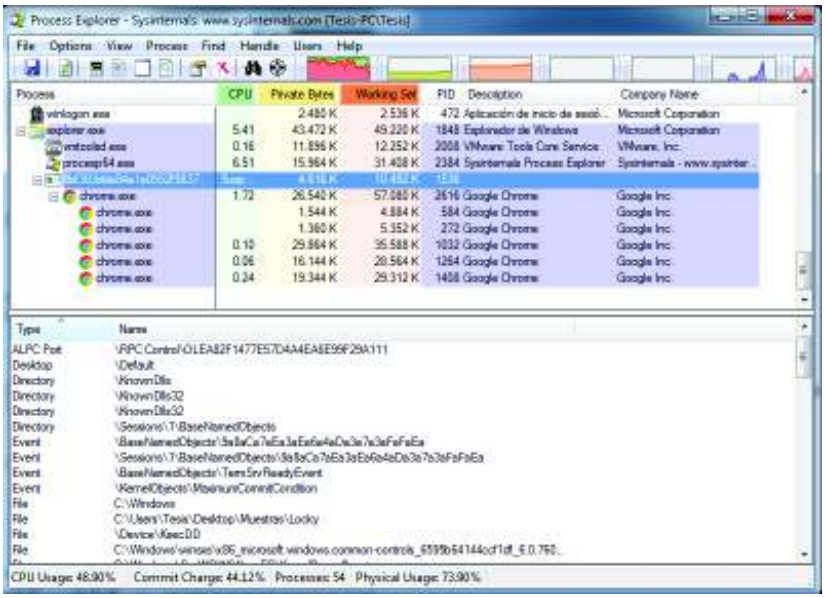

Figura 3.84. Procesos ejecutados durante la infección de Locky.

Adicional en la parte inferior se evidencian las librerías convocadas por el malware Locky. El proceso adicional llama a Chrome.exe, puesto que abre el navegador indicando que los archivos fueron encriptados.

Analizando el proceso levantado por el malware durante la infección se verificó que el ejecutable no tiene una firma certificada por Microsoft, indicando que es un archivo malicioso, como se puede observar en la Figura 3.85.

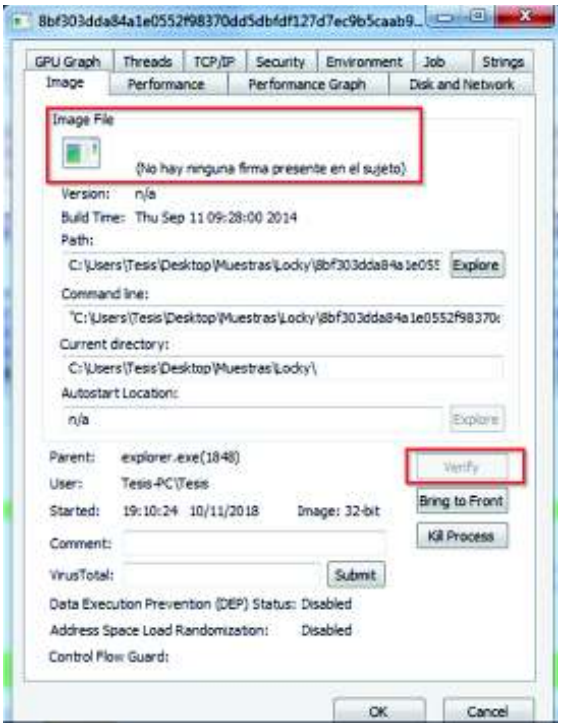

Figura 3.85. Verificación de firma de Locky.

- **Process Monitor.**

Durante los experimentos de análisis dinámico se ejecutó Process Monitor para la captura de eventos del sistema y se permitió al Malware ejecutarse durante 5 minutos antes de detener la captura con ProcMon.

De esta forma se obtuvo un número alto (en el orden de las centenas de miles) que resulta excesivo para el análisis. Es necesario filtrar la información capturada de manera que resulte útil para el análisis en proceso.

Process Monitor permite exportar la información capturada en formato ".CSV" (Comma Separated Values), lo que permite visualizar y filtrar de manera más rápida los resultados dentro de una hoja de cálculo. Se procede entonces a observar inicialmente los resultados relacionados con el archivo original de la muestra. A continuación se resumen los resultados más importantes obtenidos a partir de esta técnica.

En primer lugar se procede a explorar la información capturada referente al archivo original de la muestra, para observar los cambios realizados en el sistema y sus procesos asociados. Se encuentra en primer lugar en su lista de propiedades dos módulos, como se indica en la Figura 3.86:

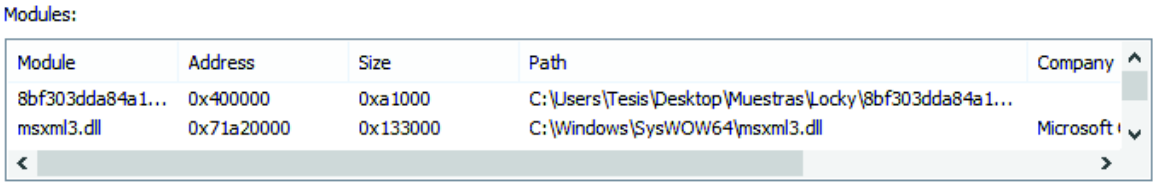

Figura 3.86. Módulos encontrados en Locky.

De donde se puede inferir que Locky hace uso del lenguaje XML en su versión 3

En la Figura 3.87 se puede observar el árbol de procesos realizados por Locky.

.

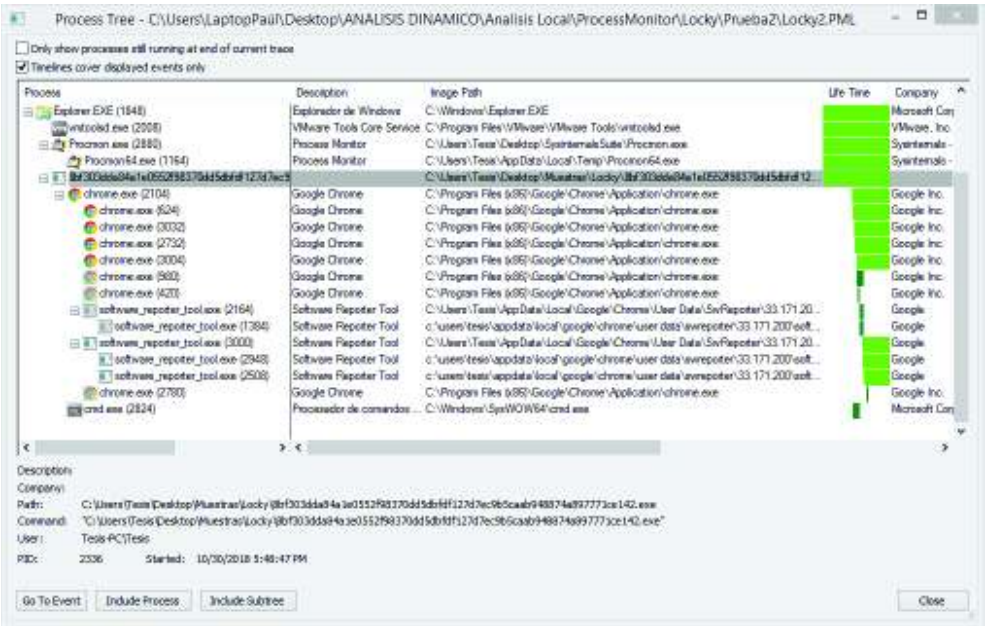

Figura 3.87. Árbol de procesos de Locky.

Se observa que Locky se ejecuta dentro de una sola instancia, y que para la máquina víctima utilizada en el experimento, usa procesos relacionados con Google Chrome.

Se procede a utilizar la opción “Ir al evento” para observar los detalles relacionados al mismo. Y detallar lo que hacen.

Los eventos relacionados con el explorador Google Chrome son utilizados para mostrar el mensaje de captura y rescate utilizado por el Ransomware para notificar que la información ha sido encriptada. En los detalles del evento se puede observar que el documento cargado dentro del explorador de internet es el html creado por Locky.

Los procesos posteriores tienen que ver con fallos de Google Chrome que requieren el reinicio de dicho programa, y la herramienta de reporte de errores del mismo, intentado establecer una comunicación.

Finalmente se puede ver un proceso relacionado con cmd. Revisando el evento relacionado con dicho proceso se encuentra que corresponde a la ejecución de la instrucción “cmd.exe /C del /Q /F "C:\Users\Tesis\AppData\Local\Temp\sys3D9C.tmp"
Es decir que utiliza el símbolo del sistema de Windows para eliminar un archivo temporal almacenado sin interacción del usuario.

Se realiza un listado de los archivos creados por el proceso “8bf303dda84a1e0552f98370dd5dbfdf127d7ec9b5caab948874a897771ce142.exe” y se encuentra que según la herramienta Process Monitor se produjeron 3240 cambios, entre los correspondientes a archivos y carpetas.

Entre las claves modificadas por el malware Locky, se observa que modifica el fondo de pantalla, utiliza Direct3D (su nombre queda registrado como última aplicación utilizasa), y finalmente programa una operación de manejo de un archivo temporal (C:\Users\Tesis\AppData\Local\Temp\sys3D9C.tmp) mediante Pending File Rename Operations.

De las claves consultadas se puede destacar que el archivo se interesa por información relacionada con las extensiones de archivo htm y bmp (las mismas que utiliza posteriormente para crear sus archivos de rescate). Además, consulta valores de claves

relacionadas con servicios OLE, COM3, RPC, SQM, VSS y el servicio criptográfico de Windows, de lo que se puede sospechar el interés del software por establecer comunicaciones o interactuar con las copias de seguridad del sistema operativo.

- **Resultado de la Inflexión**

Lo primero que se puede evidenciar es que el malware se muestra como un archivo ejecutable, como se puede ver en la Figura 3.88.

Figura 3.88. Malware Locky antes de ser ejecutado.

Una vez ejecutado el malware, demora entre 5 a 7 minutos en mostrar cambios en el equipo infectado, la primera señal de la infección se evidencia al abrir el navegador del equipo y se indica una imagen que señala que los archivos se encuentran cifrados, como se puede observar en la Figura 3.89 y en la Figura 3.90.

Figura 3.89. Solicitud de recompensa con Locky, mediante navegador.

Tanto en la imagen como en el navegador el mensaje muestra una identificación la cual es única para cada equipo infectado, el ID es 4K5XM1JZ30EXU40K.

Figura 3.90. Solicitud de recompensa con Locky, mediante imagen.

Inmediatamente después, se cambia automáticamente el fondo de pantalla del equipo infectado por la imagen que el virus indica en la primera interacción, como se observa en la Figura 3.91.

Figura 3.91. Solicitud de recompensa con Locky en fondo de pantalla.

Al revisar los archivos del equipo infectado, estos cambian de extensión a .ykcol, evitando que puedan ser recuperados o modificados. Adicional se puede observar en la Figura 3.92 que los nombres de los archivos cifrados cambian de nombre, empezando todos con el identificador que se dio al equipo infectado.

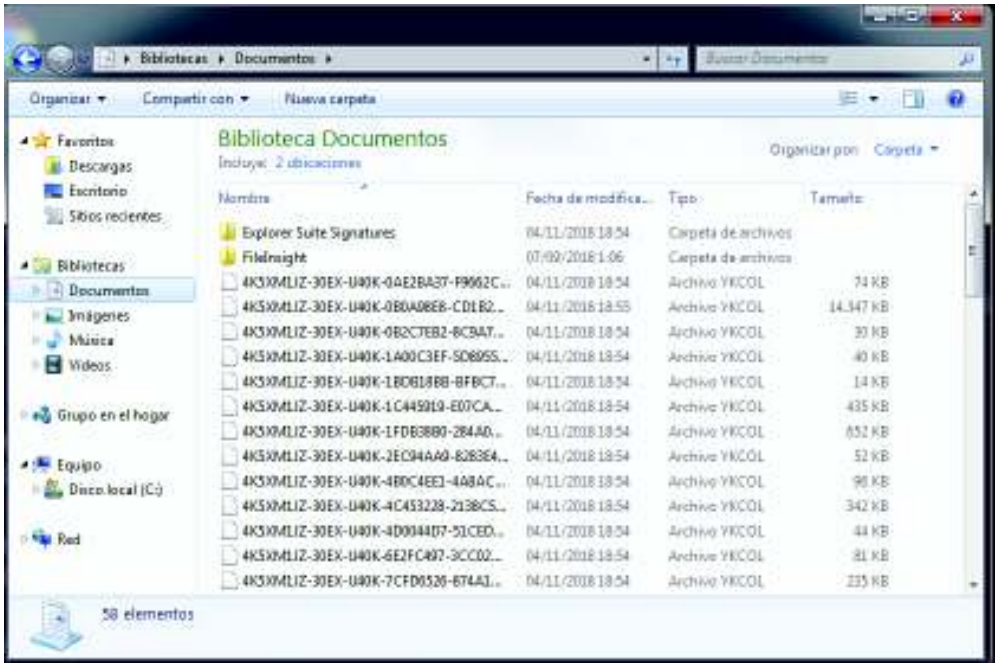

Figura 3.92. Documentos cifrados con Locky.

Se tomó una muestra de 68 distintos tipos de archivos con sus respectivas extensiones, de las cuales Locky modificó y encriptó 48 de los archivos como se puede observar en la Figura 3.92.

- **Análisis de Actividad de Red**
  - **ApateDNS**

Se utilizó ApateDNS para capturar las solicitudes de DNS realizadas en el equipo local. Se fijó la dirección de respuesta a la máquina virtual Kali Linux de servicio y se inició el proceso de captura.

Se realizó en primera instancia una captura de solicitudes de DNS sin la ejecución de ninguna de las muestras de ransomware, para filtrar los resultados de esta captura separándolos de los que pudieran aparecer al ejecutar una muestra de malware.

Para la muestra de Locky, no se hallaron solicitudes de DNS. Sin embargo, se debe destacar que el mensaje de rescate mostrado por este ransomware solicita la descarga de Tor Web Browser para contactarse con el centro de comando y control para proceder con el proceso de rescate de la información.

Se procedió a realizar un análisis para cada una de las URLs detectadas, este análisis también incluye: una búsqueda de información sobre dicha URL a través del servicio WHOIS proporcionado por https://whois.icann.org, una búsqueda de malignidad a través de www.virustotal.com y finalmente una búsqueda de informe de malignidad dentro del sitio https://www.threatcrowd.org.

Para la dirección g46mbrrzpfszonuk.onion/4K5XM1JZ30EXU40K No se halló información correspondiente al servicio de WHOIS provisto por icann.org. Dentro de la plataforma VirusTotal, la URL fue detectada como maliciosa por 1 de 63 motores de búsqueda.

- **Wireshark**

Desde la máquina virtual de servicios se capturan paquetes provenientes de la máquina virtual víctima durante 12 minutos, lo cual permitió la captura de 347 paquetes para su análisis

Como se indica en la Figura 3.93 es destacable un conjunto de paquetes que intentan establecer una comunicación a través del puerto 137, y utilizar el servicio NBNS para buscar información sobre grupos de trabajo, proxy y distinta información sobre la red.

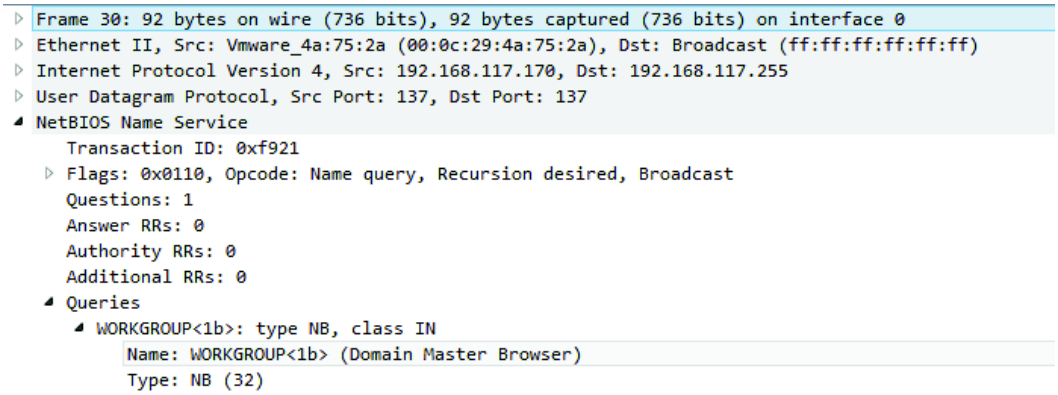

Figura 3.93. Peticiones por el puerto 137 realizadas por Locky.

Se encuentran paquetes que aparentemente utilizarían el protocolo NBNS para averiguar la dirección de las URLs SSL.GSTATIC.COM y WWW.GSTATIC.COM, a través del puerto UDP 137, como se muestra en la Figura 3.94.

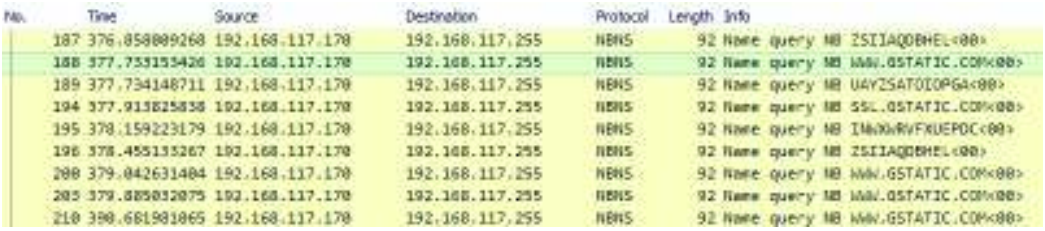

Figura 3.94. Peticiones realizadas por Locky.

Se debe destacar además que utiliza el protocolo LLMR desde distintos puertos con dirección al puerto UDP 5355 para intentar establecer comunicaciones.

Se realizó un análisis de reputación de las URLs mencionadas con la herramienta online VirusTotal.com, en el cuál buscó el número de herramientas antivirus que reportan la dirección IP como maliciosa con respecto al número total de búsquedas, y además la calificación que les es dada por parte de usuarios de la herramienta Virustotal. Los resultados se indican en la Tabla 3.12.

Tabla 3.12. Reputación de IPs destino, por parte de Locky

URL	Detecciones	Calificación de la comunidad
SSL.GSTATIC.COM	0/70	-39
WWW.GSTATIC.COM	0/70	-68

- **TcpDump**

Al realizar un escaneo del puerto 137, se encuentra que realiza peticiones a la IP 192.168.117.255 como se observa en la Figura 3.95, las cuales son peticiones al NetBios.

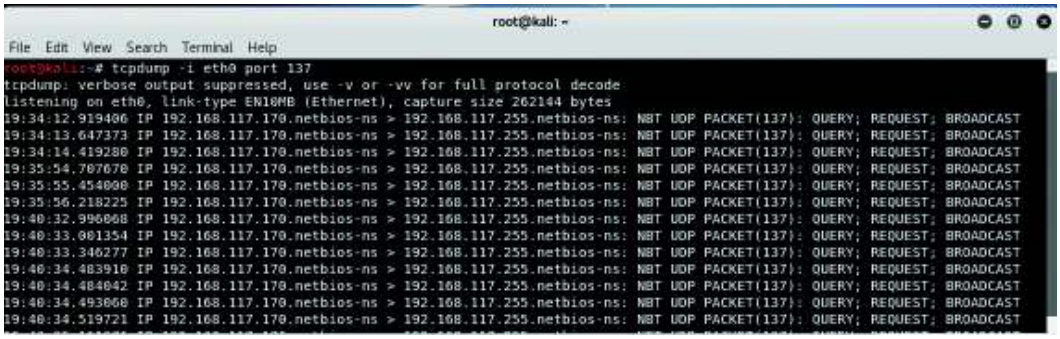

Figura 3.95. Peticiones por hacia el puerto 137 por el equipo infectado con Locky

Al realizar un escaneo del puerto 5355, se encuentra que realiza peticiones a la IP 224.0.0.253, como se puede observar en la Figura 3.96.

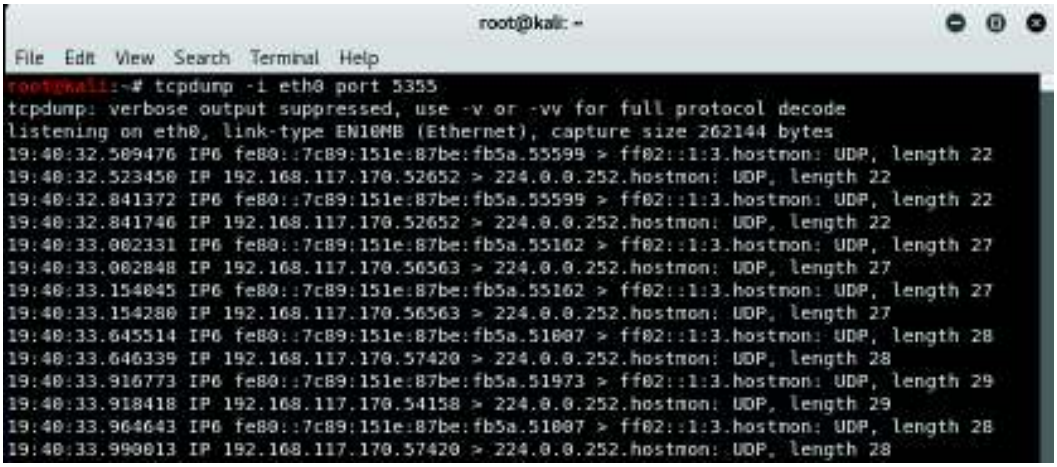

Figura 3.96. Peticiones por hacia el puerto 80 por el equipo infectado con Locky

De esta manera se reafirma la información encontrada con WireShark.

Análisis de Spora

- **Análisis Automatizado**

Al subir las muestras al portal de Threat Analyzer, se obtiene la información que se indica en la Figura 3.97:

Figura 3.97. Análisis de Spora en Threat Analyzer

1. Entrega los hashes SHA256, SSDEP, MD5 y SHA1.
2. El número de cambios de archivos es 56 y 35 de registros.
3. Está categorizado como un malware de tipo ransomware, de la familia Spora.
4. Se puede observar que entre los 56 cambios realizados se incluyen procesos y servicios.
5. Se detectan 2 riesgos conocidos, 1 riesgo alto y 21 riesgos variados.

Como se muestra en la Figura 3.98 al subir la muestra en el sitio de Hybrid Analysis se obtuvieron los siguientes resultados:

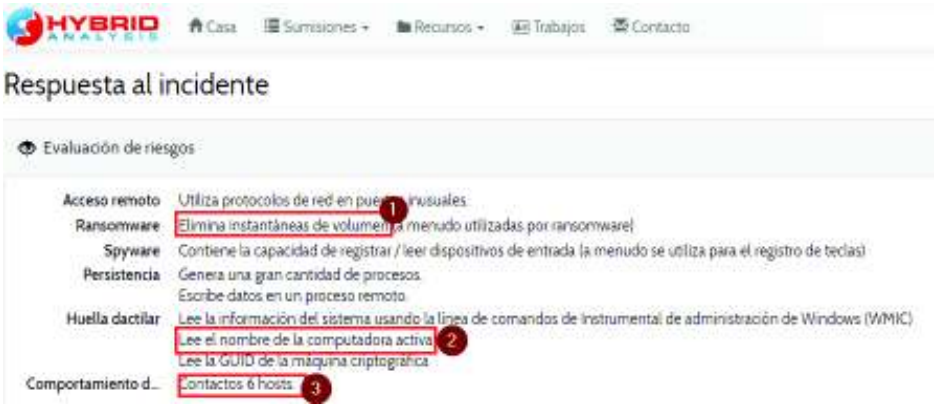

Figura 3.98. Análisis de Spora en Hybrid Analysis

1. Elimina snapshots del equipo, evitando restaurar el equipo.
2. Lee datos como nombre del equipo.
3. Realiza la petición a 6 host.

En la Tabla 3.13, se muestran las direcciones hacia donde realiza las peticiones:

Tabla 3.13. Países hacia donde realiza peticiones Spora.

HOST	PAÍS
216.58.208.206	Estados Unidos
216.58.209.228	Estados Unidos
216.58.209.238	Estados Unidos
216.58.209.228	Estados Unidos
216.58.209.238	Estados Unidos
31.192.105.180	Rusia

En la Figura 3.99, se muestra un mapa de los 6 host contactados por el malware, como se puede observar la mayoría corresponden a peticiones hacia Estados Unidos:

Figura 3.99. Países hacia donde realiza peticiones Spora.

- **Análisis Local**
  - **AutoRuns**

Para la obtención de resultados por parte de Autoruns se realizó y guardó un primer análisis del estado de la máquina virtual víctima del laboratorio de análisis, después se ejecutó la muestra de virus durante siete minutos, se realizó un nuevo análisis con AutoRuns del estado de la máquina y se lo comparó con el análisis previamente guardado.

Dentro de los resultados, para las tres ejecuciones realizadas se obtuvo que:

El virus añadió un programa a los ejecutables de inicio “README_sTlLoTpq” que ejecuta un programa localizada en la carpeta del menú de inicio “c:\Users\Tesis\AppData\Roaming\Microsoft\ Windows\Start Menu\Programs\Startup”, no se registró cambio en el nombre del archivo para las distintas instalaciones.

En la Figura 3.100, se observa la pantalla del análisis comparativo realizado por Autoruns.

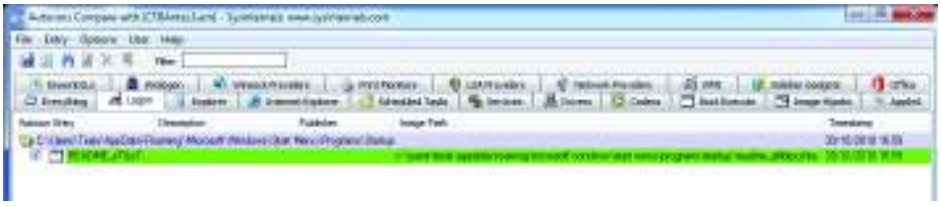

Figura 3.100. Comparación realizada por Autoruns para Spora.

Como se indica en la Figura 3.101 Autoruns permite acceder de manera directa a la pantalla de propiedades del archivo en cuestión, lo cual posibilita obtener la siguiente información sobre el mismo.

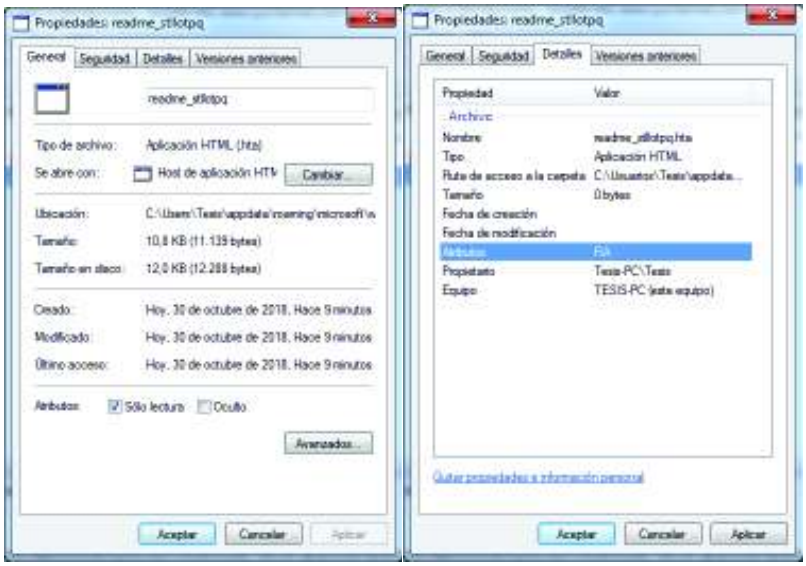

Figura 3.101. Propiedades del archivo ejecutado por Spora.

Se puede observar dentro de estas propiedades que el archivo en cuestión es una aplicación HTML, y que ha sido guardada como un archivo de tipo solo lectura.

A continuación se procede a tomar las huellas de dicho archivo de acuerdo con los algoritmos MD5, SHA1 y SHA256, como se muestra en la Tabla 3.14.

Tabla 3.14. Huellas de Spora.

MD5	cc8f62577653ea32be45fcde9c5de394
SHA1	72B20C5774459DFB3C8EBD12377F796CD46D8CB4
SHA256	A0C74590BD2DF131C241929309B639F6EF7FD8158BDDB8ABF1CD95B809BCEBEC

- **Regshot**

Al analizar los cambios realizados por la muestra Spora con RegShot, se observaron varios cambios significativos.

Se observan que existen registros, archivos y carpetas modificadas, añadidas o eliminadas que implica que se realizaron cambios en las configuraciones, caché, historial y contenido temporal de los navegadores instalados en el equipo, estas modificaciones se las pueden observar en:

- HKU\Software\Microsoft\Windows\CurrentVersion\Explorer
- HKU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\

Se observan cambios a nivel de las tareas programadas en HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Schedule\TaskCache\Tasks

Adicional se observan 191 archivos modificados, los cuales en su mayoría son documentos de Microsoft Office, así como imágenes y archivos comprimidos, los archivos fueron tomados de varias capetas del equipo, al intentar ingresar a estos archivos, estos se encontraban encriptados.

Se añaden algunas claves, en donde se puede configurar de tamaño, ícono, vista o posición de las capetas, así como el de servicio y configuración del sistema:

- HKLM\SYSTEM\ControlSet001\services\
- HKLM\SYSTEM\CurrentControlSet\Control\Class\
- HKU\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\

Se detectó que se agregaron varios archivos, 2 de ellos suponen una alerta ya que toman el nombre de Aplicación HTML y Barra de Favoritos, estos generalmente son los que utiliza el malware para indicar que el equipo se encuentra con archivos comprometidos:

- HKU\.DEFAULT\Software\Classes\Local Settings\MuiCache\2B\A7EAB198\@C:\Windows\System32\mshta.exe,-6412: "Aplicación HTML"
- HKU\Software\Classes\Local Settings\MuiCache\2B\A7EAB198\@C:\Windows\System32\ieframe.dll,-12385: "Barra de favoritos"

A nivel de carpetas modificadas, se observa que existe un cambio en las propiedades de las 2 capetas de los usuarios del equipo:

- C:\Users\Public
- C:\Users\Tesis

Una vez infectado el equipo se evidencia un total de 240 cambios realizados, tanto en archivos como claves.

- **Process Explorer**

Al realizar la infección con Spora se monitoreó el equipo con la herramienta Process Explorer, evidenciando los siguientes cambios en el consumo de recursos:

- A nivel de CPU el equipo tuvo varios picos del 100% de uso, causando lentitud en su normal rendimiento.
- Se detectaron 2 picos de uso inusual de la red.
- El disco tuvo picos cercanos al 80% a causa de los procesos en segundo plano que Spora ejecuta.

Esto se muestra en la Figura 3.102.

Figura 3.102. Consumo de recursos durante la infección de Spora.

En la Figura 3.103, se muestran los procesos que se ejecutaron al momento de la infección indicando cuanto CPU ocupa cada uno de ellos, con una breve descripción del proceso. Adicional en la parte inferior se evidencian las librerías convocadas por el malware Spora, así como los cambios en varios de los registros.

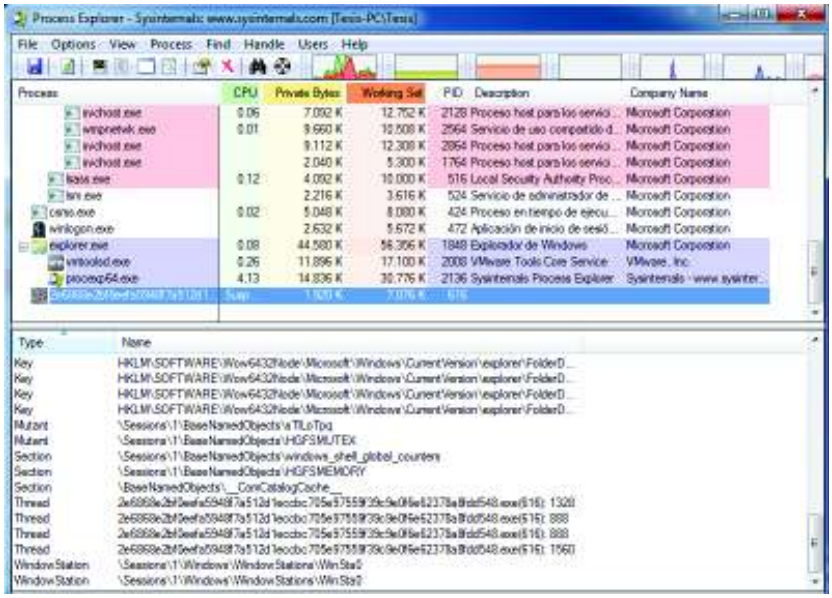

Figura 3.103. Procesos ejecutados durante la infección de Spora.

Analizando el proceso levantado por el malware durante la infección se verificó que el ejecutable no tiene una firma certificada por Microsoft, e indicando que es un archivo malicioso, como se puede observar en la Figura 3.104.

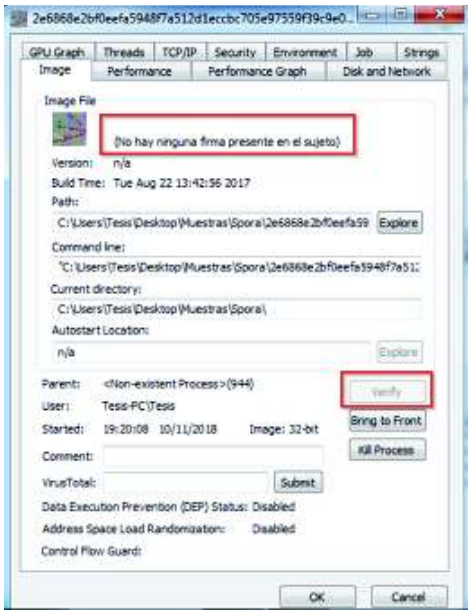

Figura 3.104. Verificación de firma de Spora.

- **Process Monitor**

Durante los experimentos de análisis dinámico se ejecutó Process Monitor para la captura de eventos del sistema y se permitió al Malware ejecutarse durante 5 minutos antes de detener la captura con ProcMon.

De esta forma se obtuvo un número alto (en el orden de las centenas de miles) que resulta excesivo para el análisis. Será necesario filtrar la información capturada, separándola del resto de manera que resulte útil para el análisis en proceso.

Process Monitor permite exportar la información capturada en formato ".CSV" (Comma Separated Values), lo que permite visualizar y filtrar de manera más rápida los resultados dentro de una hoja de cálculo.

Se procede entonces a observar inicialmente los resultados relacionados con el archivo original de la muestra. A continuación se resumen los resultados más importantes de esta manera.

En la Figura 3.105, se puede observar el árbol de procesos realizados por Spora.

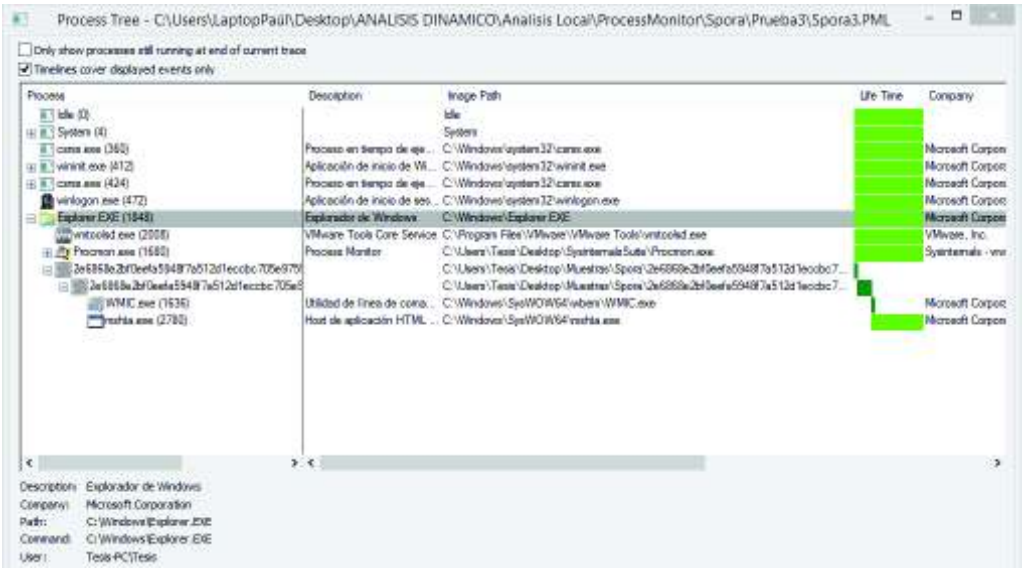

Figura 3.105. Árbol de procesos de Spora.

Es posible observar que Spora se ejecuta dentro de una sola instancia, y que para la máquina víctima utilizada en el experimento, usa procesos relacionados con Google Chrome.

Se procede a utilizar la opción “Ir al evento” para observar los detalles relacionados al mismo. Y detallar lo que hacen.

En primer lugar se analiza el proceso relacionado con “WMIC.exe”. WMIC, estos son es una utilidad de línea de comandos para la infraestructura de administración de Intrumentación de Windows (WMI). Mediante un conjunto de comandos, esta utilidad permite realizar consultas y cambios críticos relacionados con el funcionamiento del sistema.

Spora utiliza “wmic.exe” para ejecutar el comando “"C:\Windows\System32\wbem\ WMIC.exe" process call create "cmd.exe /c vssadmin.exe delete shadows /quiet /all"”, es decir que intenta eliminar las copias de seguridad.

Se analiza a continuación el proceso relacionado con “mshta.exe”. Esta aplicación es la responsable del lanzamiento de aplicaciones HTML mediante un explorador de internet como Microsoft Internet Explorer. El evento concerniente a dicha aplicación ejecuta el comando “C:\Windows\SysWOW64\mshta.exe" "C:\Users\Tesis\AppData\Roaming\ Microsoft\Windows\Start Menu\Programs\Startup\README_sTlLoTpq.hta"”, es decir que ejecuta “README_sTlLoTpq.hta”, aplicación HTML discutida anteriormente en los resultados obtenidos mediante el análisis con AutoRuns.

Se procede a realizar un listado de los archivos creados por el proceso “2e6868e2bf0eefa5948f7a512d1eccbc705e97559f39c9e0f6e62378a8fdd548.exe” y se encuentra que según la herramienta Process Monitor se produjeron 5148 eventos relacionados con la creación o modificación de archivos y carpetas.

En la carpeta que ejecuta la muestra se observa que se crean varios archivos; entre los archivos, la mayoría corresponde a librerías dinámicas las cuales pueden ser utilizadas durante el proceso de encriptación e intentos del archivo por comunicarse con un Servidor de Control y comando.

Es importante mencionar la presencia del programa “sTlLoTpq.exe” por lo que se analizaron los eventos registrados por Process Monitor para dicho ejecutable, sin encontrar resultados relevantes.

De las claves consultadas se puede destacar que el archivo se interesa por información relacionada con los paquetes de lenguaje presentes en el sistema y la configuración del explorador de internet. Además consulta valores de claves relacionadas con servicios OLE, COM3, RPC, y el servicio criptográfico de Windows, de lo que se puede sospechar el interés del software por establecer comunicaciones con equipos distantes, posiblemente un servicio de comando y control.

- **Resultado de la Inflexión**

Lo primero que se puede evidenciar es que el malware se muestra como un archivo ejecutable, como se puede ver en la Figura 3.106.

Figura 3.106. Malware Spora antes de ser ejecutado.

Una vez ejecutado el malware, demora entre 2 a 4 minutos en realizar el primer cambio en el equipo con un mensaje sobre fondo de pantalla del escritorio, como se puede observar en la Figura 3.107.

Figura 3.107. Solicitud de recompensa con Spora.

En el mensaje mostrado solicita enviar un correo a spora.help@gmail.com, para tener mayor información del rescate de los archivos comprometidos.

Al revisar los archivos del equipo infectado, estos no muestran ningún cambio aparente, pero al querer ingresar a uno de ellos con extensión .doc, se despliega una ventana de Microsoft Word indicando que existe un problema con el contenido, como se muestra en la Figura 3.108. Se prueba con varios archivos, pero no se puede acceder a varios de ellos.

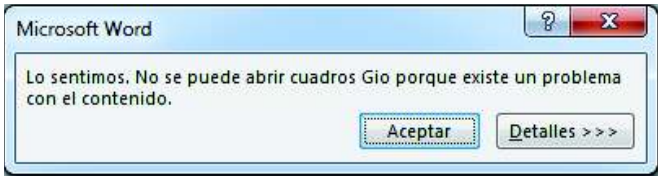

Figura 3.108. Mensaje al ingresar a un archivo comprometido.

Se tomó una muestra de 68 distintos tipos de archivos con sus respectivas extensiones, de las cuales Spora modificó y encriptó 14 de los archivos.

- **Análisis de Actividad de Red**
  - **ApateDNS**

Se utilizó ApateDNS para capturar las solicitudes de DNS realizadas en el equipo local. Se fijó la dirección de respuesta a la de la máquina virtual Kali Linux de servicio y se inició el proceso de captura.

Se realizó en primera instancia una captura de solicitudes de DNS sin la ejecución de ninguna de las muestras de ransomeware, para posteriormente filtrar los resultados de esta captura de los que pudieran aparecer al ejecutar una muestra de malware.

No se hallaron solicitudes de DNS para Spora. Para solicitar el rescate, Spora indica una dirección de email (spora.help@gmail.com) con la que debe comunicarse la víctima.

- **Wireshark**

Desde la máquina virtual de servicios se capturan paquetes provenientes de la máquina virtual víctima durante 12 minutos, lo cual permite la captura de 347 paquetes para su análisis

Es destacable un conjunto de paquetes que intentan establecer una comunicación a través del puerto 137, y utilizar el servicio NBNS para buscar información sobre grupos de trabajo, proxy y distinta información sobre la red, como se indica en la Figura 3.109.

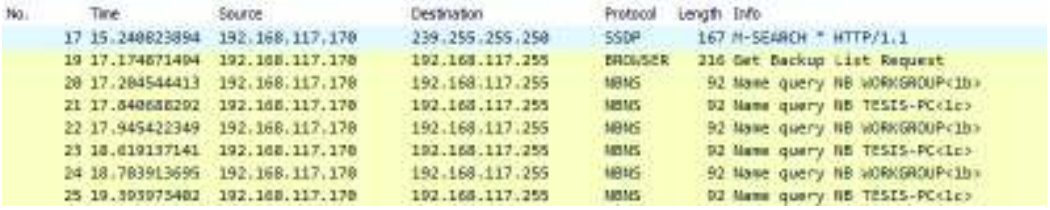

Figura 3.109. Peticiones por el puerto 137 realizadas por Spora.

No se detectan intentos de establecer una resolución de dominio, pero si se encuentran intentos de comunicación con la IP 31.192.105.180 a través del puerto destino 8123. Como se muestra en la Figura 3.110.

Figura 3.110. Peticiones por el puerto 8123 realizadas por Spora.

Se realizó un análisis de reputación de la dirección IP mencionadas utilizando la herramienta online VirusTotal.com, en el cuál se buscó el número de herramientas antivirus que reportan la dirección IP como maliciosa con respecto al número total de búsquedas, y también toma en cuenta la calificación de los usuarios de la herramienta Virustotal, como se muestra en la Tabla 3.15.

Tabla 3.15. Reputación de IPs destino, por parte de Spora

URL	Detecciones	Calificación de la comunidad
31.192.105.180	1/70	N/A

- **TcpDump**

Al realizar un escaneo del puerto 137, se encuentra que realiza peticiones a la IP 192.168.117.255 como se puede observar en la Figura 3.111, la petición es a NetBios.

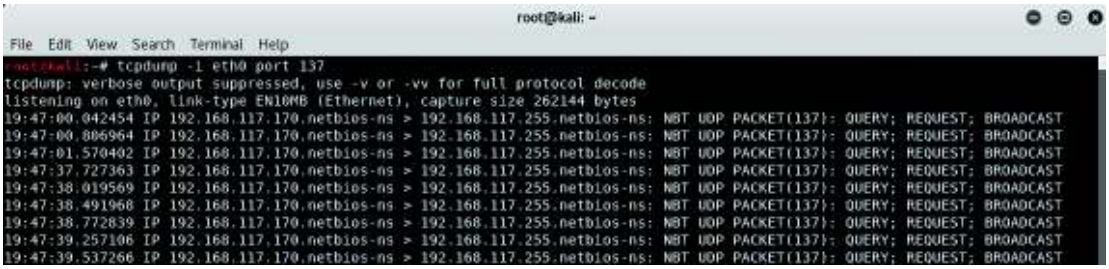

Figura 3.111. Peticiones por hacia el puerto 137 por el equipo infectado con Spora

Al realizar un escaneo del puerto 8123, se encuentra que realiza peticiones a la IP 31.192.105.180 como se puede observar en la Figura 3.112, la IP se encuentra como maliciosa en VirusTotal.

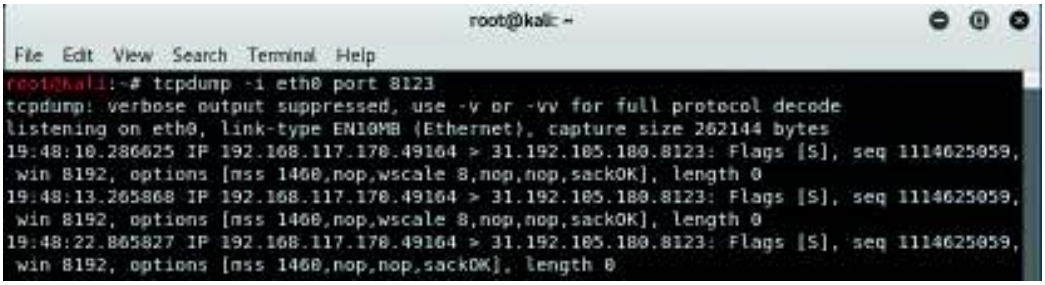

Figura 3.112. Peticiones por hacia el puerto 8123 por el equipo infectado con Spora

Análisis de WannaCry

- **Análisis Automatizado**

Al subir las muestras al portal de Threat Analyzer, se obtiene la información señalada en la Figura 3.113:

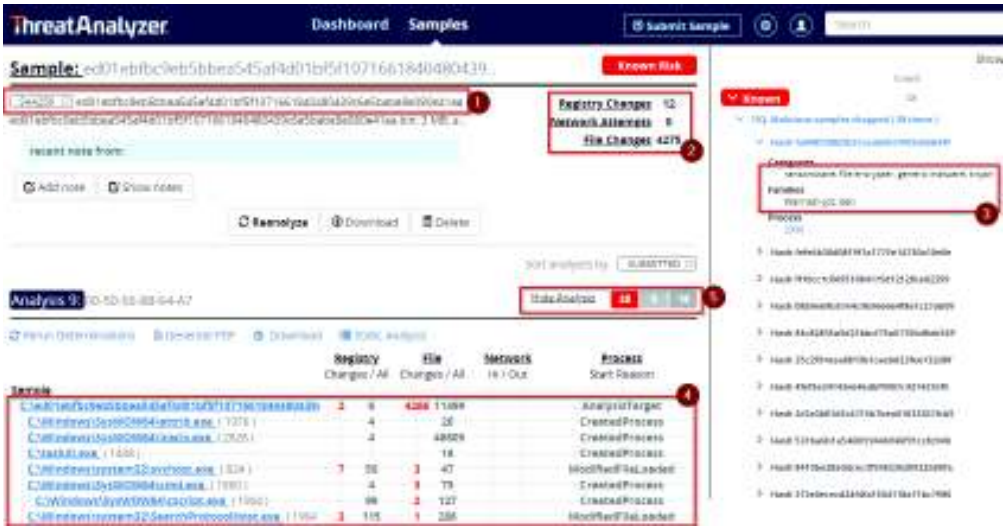

Figura 3.113. Análisis de WannaCry en Threat Analyzer

1. Entrega los hashes SHA256, SSDEP, MD5 y SHA1.
2. El número de cambios de archivos es 4275.
3. Está categorizado como un malware de tipo ransomware, de la familia WannaCrypt.
4. Se puede observar que entre los 4275 cambios realizados, se detectan creación de procesos, modificación de archivos, determinando que es un archivo malicioso.
5. Se detectan 38 riesgos conocidos y 18 riesgos variados.

Al subir la muestra en el sitio de Hybrid Analysis se obtienen los resultados señalados en la Figura 3.114:

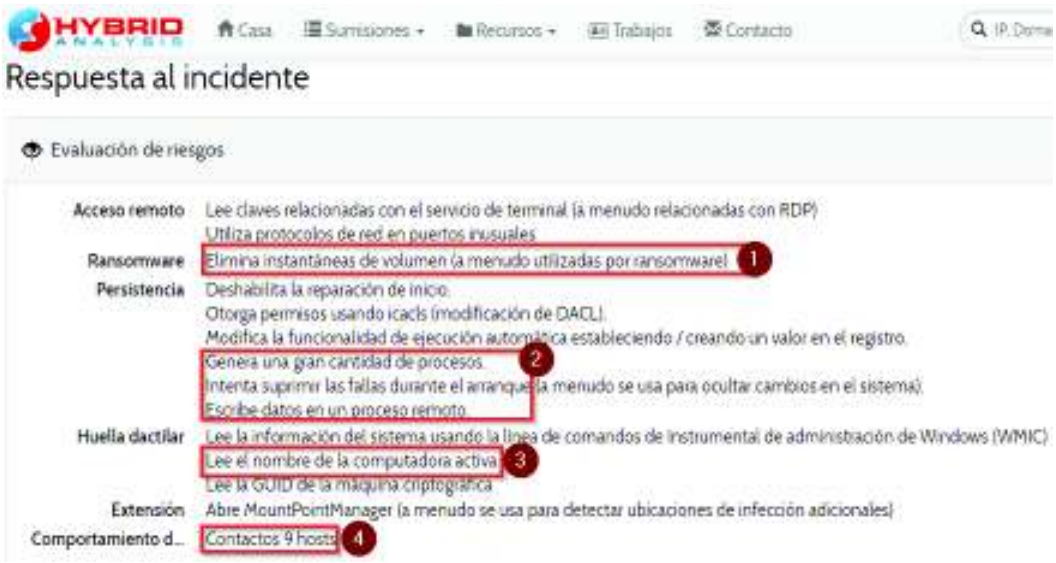

Figura 3.114. Análisis de WannaCry en Hybrid Analysis.

1. Elimina snapshots del equipo, evitando que se restaure la información comprometida.
2. Genera una gran cantidad de procesos y escribe datos en un proceso remoto.
3. Lee datos como nombre del equipo.
4. Realiza la petición a 9 host.

En la Figura 3.115, se muestra un mapa de los 9 host contactados por el malware, como se puede observar la mayoría de peticiones son hacia países de Europa.

Figura 3.115. Países hacia donde realiza peticiones WannaCry.

En la Tabla 3.16, se puede observar las direcciones de los hosts hacia donde se realizan las peticiones:

Tabla 3.16. Países hacia donde realiza peticiones WannaCry.

HOST	PAÍS
95.100.252.51	Unión Europea
46.101.151.222	Países Bajos
154.35.175.225	Estados Unidos
195.154.164.243	Francia
193.23.244.244	Alemania
85.214.206.219	Alemania
178.63.18.25	Alemania
179.43.168.166	Panamá
81.7.18.84	Alemania

- **Análisis Local**
  - **AutoRuns**

Para la obtención de resultados por parte de Autoruns se realizó y guardó un primer análisis del estado de la máquina virtual víctima del laboratorio de análisis, se ejecutó la muestra de virus durante siete minutos, se realizó un nuevo análisis con AutoRuns del estado de la máquina y se lo comparó con el análisis previamente guardado.

Este procedimiento fue repetido tres veces para notar si los nombres de los procesos, registros o archivos creados por el malware cambian de nombre dependiendo de distintas circunstancias con la finalidad de ocultarse.

Dentro de los resultados, para las tres ejecuciones realizadas se obtuvo que:
El malware añadió un registro correspondiente a los archivos ejecutables durante el arranque bajo el nombre “zbzqrtnzpab665” que ejecuta un programa localizado en la carpeta donde se hallaba la muestra desempaquetada “c:\Users\Tesis\Desktop\Muestras\Wannacry2\”, sin embargo es destacable que el archivo en cuestión aparece como no encontrado en la carpeta mencionada.

No se registró cambio en el nombre del registro o el archivo al que está enlazado dentro de las distintas instalaciones.

En la Figura 3.116, se observa la pantalla del análisis comparativo realizado por Autoruns.

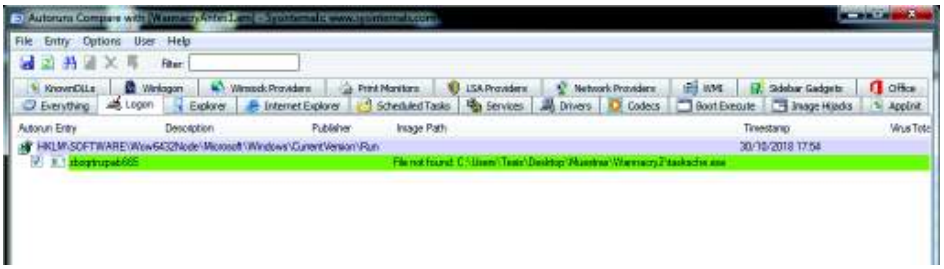

Figura 3.116. Comparación realizada por Autoruns para WannaCry.

Autoruns abre el editor del registro directamente, con lo que se observa el nombre del registro agregado y la información escrita dentro del mismo que se corresponden con las observadas anteriormente dentro de Autoruns.

Además se puede observar la ubicación dentro del registro a la que fue agregada la clave en cuestión, la cual corresponde a la subclave, que indica la ejecución de programas de 32 bits durante el arranque de Windows de 64 bits cada vez que un usuario ingresa al sistema, la clave es “HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Run” Esto se indica en la Figura 3.117.

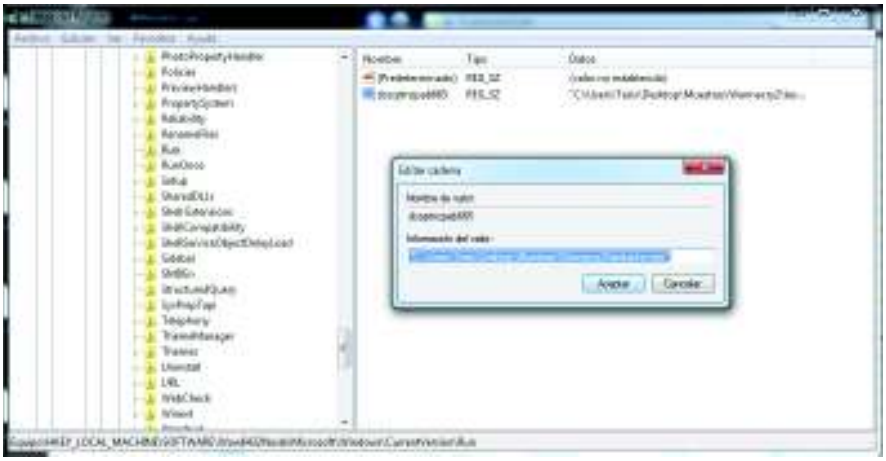

Figura 3.117. Verificación del registro realizado por Autoruns para WannaCry.

Sin embargo, el programa al que se referencia en la entrada no se muestra en la ruta especificada. Es posible que el mismo haya servido para ejecutar una única tarea que haya desencadenado la eliminación del mismo.

- **Regshot**

Al analizar los cambios realizados por la muestra Spora con de RegShot, se observaron varios cambios significativos.

Se observan que existen registros, archivos y carpetas modificadas, añadidas o eliminadas que implican cambios en las configuraciones, caché, historial y contenido temporal de los navegadores instalados en el equipo, estas modificaciones se las pueden observar en:

- HKU\Software\Microsoft\Windows\CurrentVersion\Explorer
- HKU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\

Se observan cambios a nivel de las tareas programadas en HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Schedule\TaskCache\Tasks

Se detecta que se añaden varios archivos, entre los cuales 2 de ellos resaltan ya que toman el nombre Archivo por lotes de Windows y Archivo de secuencia de comandos de VBScript:

- HKU\Software\Classes\Local Settings\ MuiCache\ 2B\ A7EAB198\ @C:\Windows\ System32\acppage.dll,-6002: "Archivo por lotes de Windows"
- HKU\Software\Classes\Local Settings\MuiCache\2B\A7EAB198\@C:\Windows\ System32\wshext.dll,-4802: "Archivo de secuencia de comandos de VBScript"

Se observan varias claves añadidas en la capeta HKLM\SYSTEM\ControlSet001\ services\VSS que almacena información sobre los respaldos y restauración del equipo, esto implica que WannaCry intenta evitar que el usuario pueda restaurar la información comprometida en los archivos encriptados.

Una vez infectado el equipo se evidencia un total de 3220 cambios realizados, tanto en archivos como claves.

- **Process Explorer**

Como se muestra en la Figura 3.118 al realizar la infección con WannaCry se monitoreó el equipo con la herramienta Process Explorer, evidenciando los siguientes cambios en el consumo de recursos:

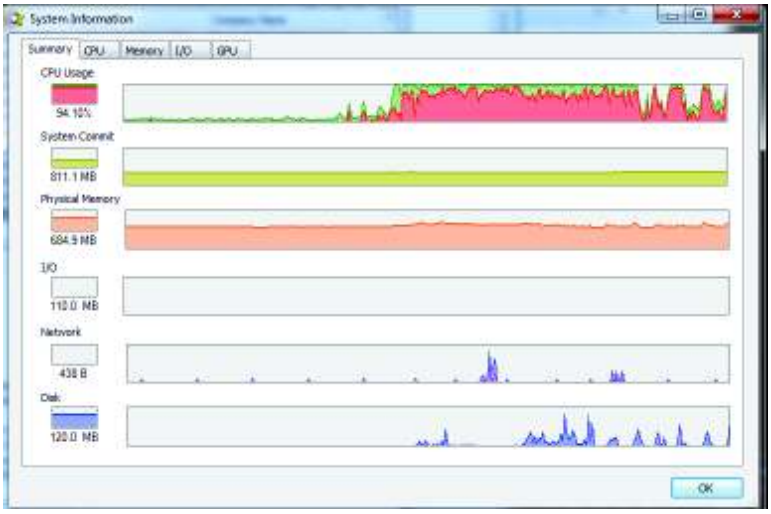

Figura 3.118. Consumo de recursos durante la infección de WannaCry.

- A nivel de CPU el equipo tuvo un consumo constante 100%, causando lentitud en su normal rendimiento.

- En cuanto a la red, se notaron varias peticiones fuera de lo normal.
- El disco tuvo picos cercanos al 100% a causa de los procesos en segundo plano que WannaCry ejecuta.

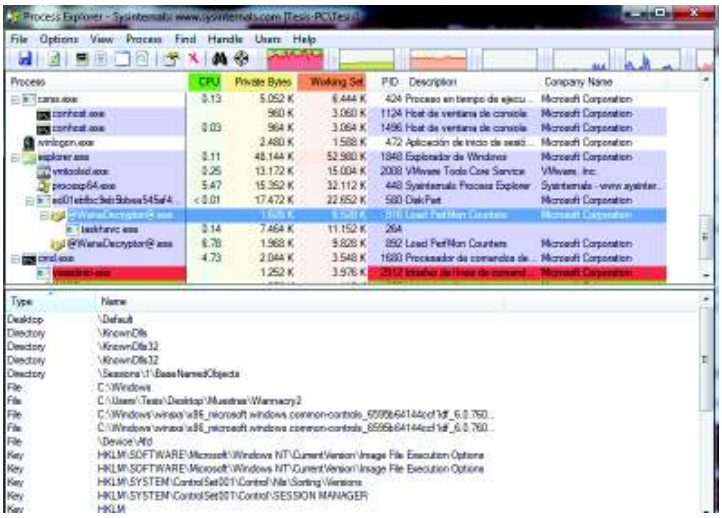

Figura 3.119. Procesos ejecutados durante la infección de WannaCry.

En la Figura 3.119, se muestran los procesos que se ejecutaron al momento de la infección indicando cuanto CPU ocupa cada uno de ellos, con una breve descripción del proceso. Adicional en la parte inferior se evidencian las librerías y llaves convocadas por el malware WannaCry.

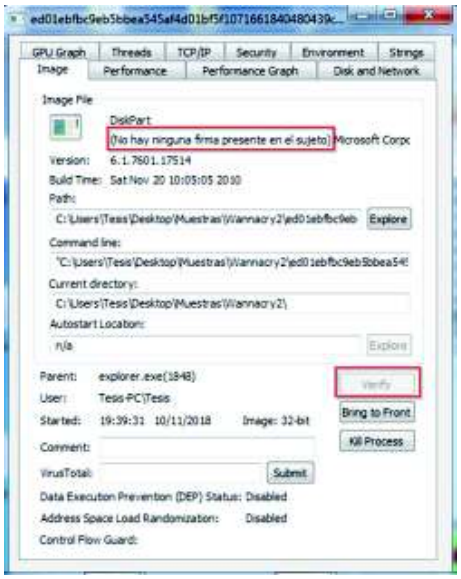

Figura 3.120. Verificación de firma de WannaCry.

Analizando el proceso levantado por el malware durante la infección se verificó que el ejecutable no tiene una firma certificada por Microsoft, que indica que es un archivo malicioso, como se puede observar en la Figura 3.120.

- **Process Monitor.**

Durante los experimentos de análisis dinámico se ejecutó Process Monitor para la captura de eventos del sistema y se permitió al Malware ejecutarse durante 5 minutos antes de detener la captura con ProcMon.

De esta forma se obtuvo un numero alto (en el orden de las centenas de miles) que resulta excesivo para el análisis. Será necesario filtrar la información capturada para que sea útil para el análisis en proceso.

Process Monitor exporta la información capturada en formato ".CSV" (Comma Separated Values), que permite visualizar y filtrar de manera más rápida los resultados dentro de una hoja de cálculo.

Se procede entonces a observar inicialmente los resultados relacionados con el archivo original de la muestra. A continuación, se resumen los resultados más importantes obtenidos de esta manera.

En la Figura 3.121 y Figura 3.122, se puede observar el árbol de procesos realizados por WannaCry.

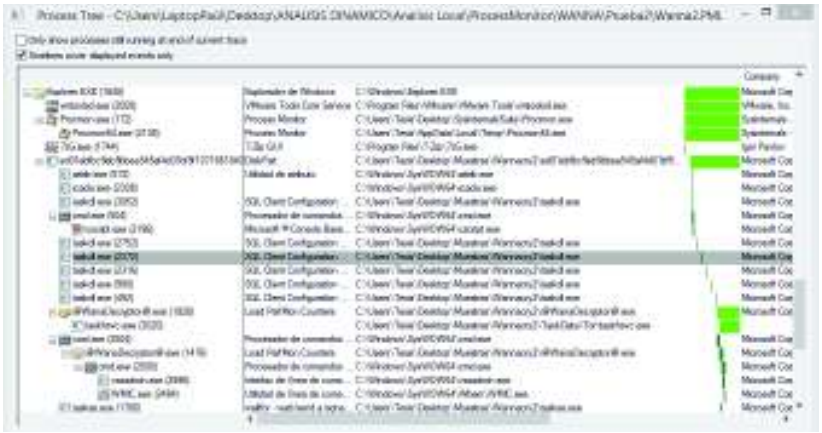

Figura 3.121. Árbol de procesos de WannaCry.

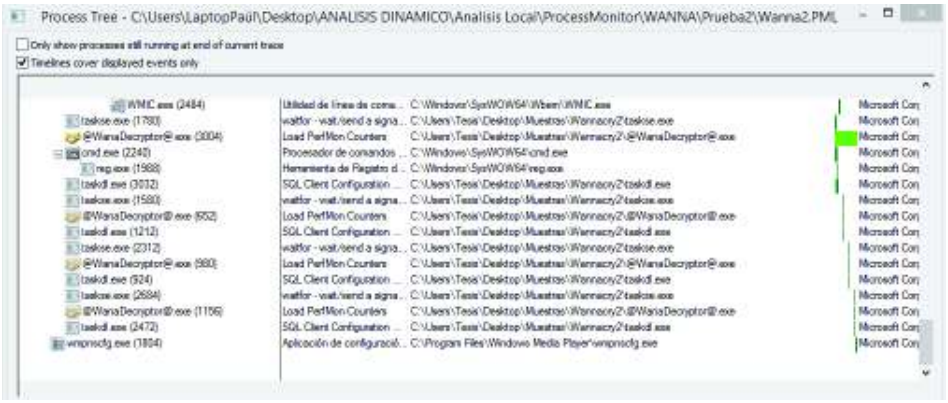

Figura 3.122. Continuación del árbol de procesos de WannaCry.

A primera vista lo más evidente es que la lista de procesos asociados con la muestra del malware es mucho más extensa que las de otras muestras. A continuación, se detallan dichos procesos y sus implicaciones:

- Primero se tiene el proceso correspondiente a Attrib.exe, que es utilizado con el propósito de cambiar los atributos de archivo en la carpeta de ejecución del malware.
- Se ejecuta posteriormente el comando “icacls . /grant Everyone:F /T /C /Q” que garantiza a todos los usuarios permisos de acceso completo y omite el despliegue de resultados de esta operación en la carpeta en la que se encuentra la muestra de malware y todos los archivos contenidos en el mismo.
- Posteriormente se ejecuta el programa “taskdl.exe” contenido dentro de la carpeta donde se encuentra la muestra y posiblemente extraída por dicha muestra. Será importante realizar un análisis de los eventos relacionados con el ejecutable en cuestión. A primera vista dentro del árbol de procesos es destacable que dichos eventos inician aproximadamente cada 30 segundos y se cierran casi enseguida.
- Posteriormente se puede ver un evento relacionado con el símbolo del sistema que al analizar sus propiedades indica que ejecuta el comando “cmd /c 262371541091063.bat” es decir utiliza el símbolo del sistema para ejecutar un archivo batch (conjunto de instrucciones para cmd) y posteriormente cierra esa instancia de línea de comandos.
- Como un subproceso relacionado con el anterior se observa el uso de cscript para ejecutar un programa de visual basic llamado “m.vbs” contenido dentro de la carpeta de la muestra.

- El siguiente evento al analizarse está relacionado con “@WanaDecryptor@.exe” ejecutándose con argumento “co” e incluyendo como subproceso a “taskhsvc.exe”. Al inspeccionar los detalles de este evento se descubre que el ejecutable se halla en la ruta C:\Users\Tesis\Desktop\Muestras\Wannacry2\ TaskData\Tor\”. Se sospecha que este sea utilizado para establecer algún tipo de comunicación remota con un servicio de comando y control mediante el servicio “Tor”.
- Posteriormente se observa un proceso correspondiente a cmd, que utiliza el comando start /b para iniciar el ejecutable “@WanaDecryptor@.exe” con argumento “vs”.

Dentro de dicho proceso se ejecuta el siguiente comando en el símbolo del sistema “cmd.exe /c vssadmin delete shadows /all /quiet & wmic shadowcopy delete & bcdedit /set {default} bootstatuspolicy ignoreallfailures & bcdedit /set {default} recoveryenabled no & wbadmin delete catalog -quiet”, que debe analizarse como cinco instrucciones separadas.

  - “vssadmin delete shadows /all /quiet” es utilizado para eliminar todas las copias de seguridad para restauración del sistema, sin la necesidad de una confirmación por parte del usuario. Este comando se aplica para Windows 8.1 y 10.
  - “wmic shadowcopy delete” elimina copias de seguridad para restauración del sistema. Este comando es aplicable a Windows Vista y 7.
  - “bcdedit /set {default} bootstatuspolicy ignoreallfailures” se utiliza para ignorar los fallos durante el arranque del sistema.
  - “bcdedit /set {default} recoveryenabled no” inhabilita el aparecimiento de la pantalla de restauración de Windows durante el arranque del sistema.
  - El comando “wbadmin delete catalog -quiet” elimina el listado de copias de seguridad almacenados en el sistema.

- “taskse.exe” crea un proceso relacionado con “@WanaDecryptor@.exe”, que encuentra este nombre de archivo en la descripción del evento. Será importante analizar eventos relacionados con dicho ejecutable para analizar sus acciones.
- Posteriormente se encuentra un evento relacionado con el símbolo del sistema para la creación de un valor en el subregistro “HKLM\SOFTWARE\Microsoft\Windows \CurrentVersion\Run” (programas que se ejecutan al iniciarse sesión) con el valor "zbzqrtnzpab665" para que ejecute el programa "C:\Users\Tesis\Desktop\

Muestras\Wannacry2\tasksche.exe\". Esta información coincide con la obtenida durante el análisis realizado con AutoRuns.

Es notable que las descripciones de los procesos relacionados con la muestra de Malware parecieran estar encaminadas a ocultar su verdadera función.

Es importante destacar que la ejecución de la muestra de Wannacry obtenida es de una estructura muy compleja en comparación con la de otras muestras que modifican menos archivos y crean menos procesos. Por esta razón es importante analizar los efectos relacionados con distintos archivos creados durante la ejecución de la muestra para entender los efectos en el sistema causados conjuntamente por estos procesos.

De las claves consultadas se puede destacar que el archivo se interesa por información relacionada con los paquetes de lenguaje presentes en el sistema y la configuración del explorador de internet. Además, consulta valores de claves relacionadas con servicios OLE, RPC, LDAP y el servicio criptográfico de Windows, de lo que se puede sospechar el interés del software por establecer comunicaciones con equipos distantes, posiblemente un servicio de comando y control, además de proceder con la encriptación y la búsqueda y manejo de archivos y carpetas.

Se realiza un listado de los archivos creados por el proceso taskdl.exe y se encuentra que según la herramienta Process Monitor se crearon 1388 archivos con nombre numérico en la carpeta temporal (“C:\Users\Tesis\AppData\Local\Temp”) con la extensión “.WNCRYT”, fuera de esto es destacable la creación de los siguientes archivos

- C:\Users\Tesis\AppData\Local\Temp\hibsys.WNCRYT
- C:\Users\Tesis\Desktop\Muestras\Wannacry2\MSVCP60.dll
- C:\Windows\Prefetch\TASKDL.EXE-428FF763.pf

La carpeta Prefetch dentro de “C:\Windows\” cumple la función de permitir a ciertos programas correr más rápido, en este caso se aplica para el archivo taskdl.exe

Por su parte el ejecutable Taskse.exe modifica información para las varias claves del registro. De lo que se puede prever posibles intentos de conexión por escritorio remoto, lectura del nombre activo del equipo y leer información sobre sesiones.

- **Resultado de la Inflexión**

Lo primero que se puede evidenciar es que el malware al ser ejecutado inmediatamente despliega varios archivos en la misma carpeta en la que se encuentra la muestra, como se puede ver en la Figura 3.123.

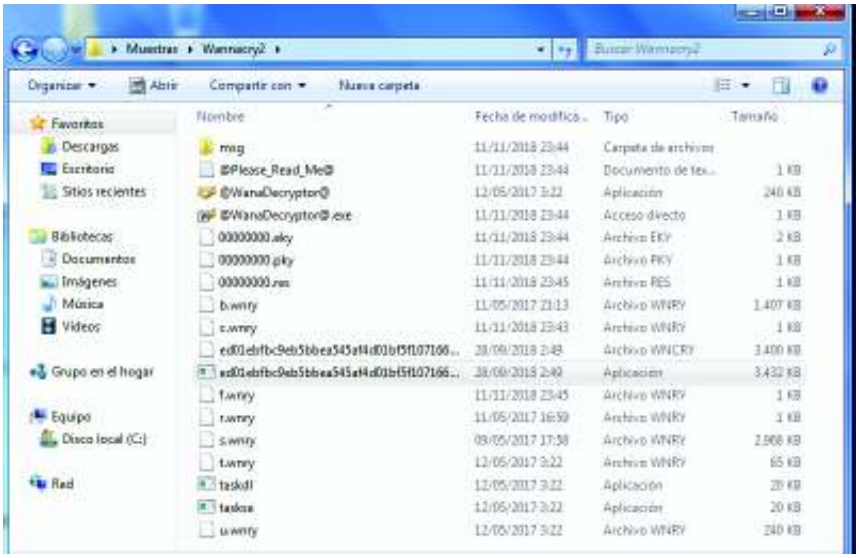

Figura 3.123. Malware WannaCry al ser ejecutado.

Una vez ejecutado el malware, demora entre 5 a 7 minutos en mostrar cambios en el equipo infectado, la primera señal de la infección es modificar el fondo de pantalla, indicando que los archivos se encuentran cifrados y que se deberá ejecutar el archivo WannaDrecryptor@.exe para recuperar la información, como se puede observar en la Figura 3.124.

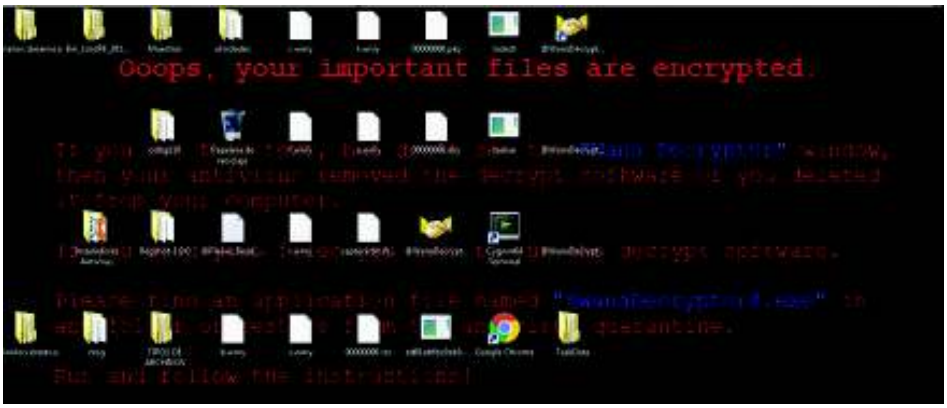

Figura 3.124. Solicitud de recompensa con WannaCry en fondo de pantalla.

Inmediatamente después, se despliega una ventana como se observa en la Figura 3.125, la cual entrega varia información como el tiempo en que se deberá pagar el rescate de la información comprometida.

Figura 3.125. Solicitud de recompensa con WannaCry mediante aplicación emergente.

En la Figura 3.126 se observan los documentos cifrados pro WannaCry.

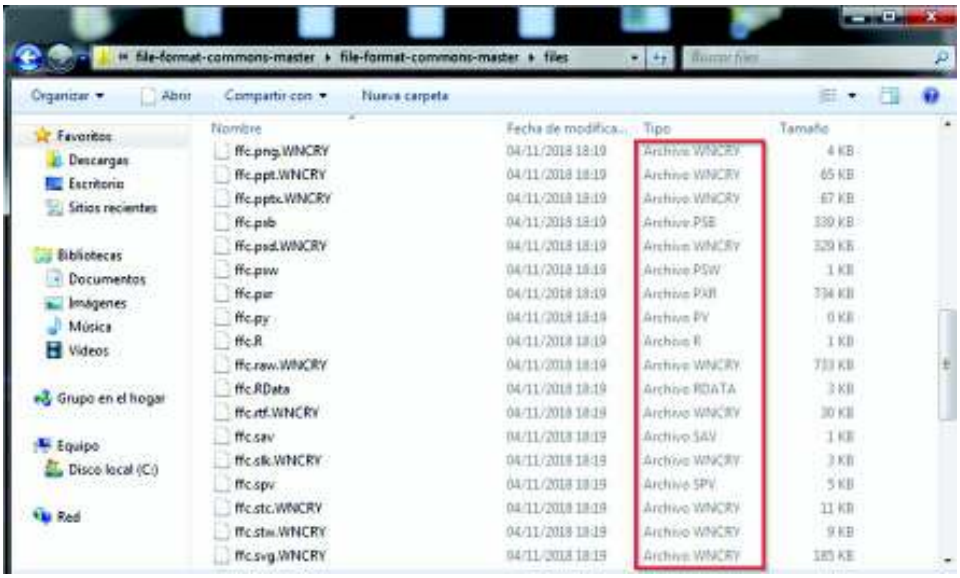

Figura 3.126. Documentos cifrados con WannaCry.

Al revisar los archivos del equipo infectado, estos cambian de extensión a .WNCRY evitando que puedan ser recuperados o modificados. Se tomó una muestra de 68 distintos tipos de archivos con sus respectivas extensiones, de las cuales WannaCry modificó y encriptó 40 de los archivos.

- **Análisis de Actividad de Red**
  - **ApateDNS**

Se utilizó ApateDNS para capturar las solicitudes de DNS realizadas en el equipo local. Se fijó la dirección de respuesta a la de la máquina virtual Kali Linux de servicio con IP (192.168.117,169) y se inició el proceso de captura.

Se realizó en primera instancia una captura de solicitudes de DNS sin la ejecución de ninguna de las muestras de ransomware, para filtrar los resultados de esta captura separándolos de los que pudieran aparecer al ejecutar una muestra de malware.

No se hallaron solicitudes de DNS para Wannacry mientras se ejecutaba de manera normal. Para pedir rescate, Wannacry muestra una interfaz. La interacción con dicha interfaz no genera solicitudes de DNS, aunque al hacer click en “Check Payment”, aparentemente se intenta contactar con un servidor.

  - **Wireshark**

Desde la máquina virtual de servicios se capturan paquetes provenientes de la máquina virtual víctima durante 12 minutos, lo que permite la captura de 347 paquetes para su análisis.

Como se observa en la Figura 3.127, es destacable un conjunto de paquetes que intentan comunicarse a través del puerto 137, y utilizar el servicio NBNS para buscar información sobre grupos de trabajo, proxy y distinta información sobre la red.

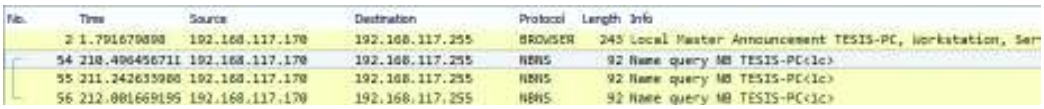

Figura 3.127. Peticiones por el puerto 137 realizadas por WannCry.

Posteriormente se puede observar que la máquina víctima intenta establecer comunicaciones TCP con diversas direcciones IP a través de un conjunto de puertos. Se muestra a continuación las capturas de los paquetes correspondientes a tales comunicaciones, como se muestra en la Figura 3.128.

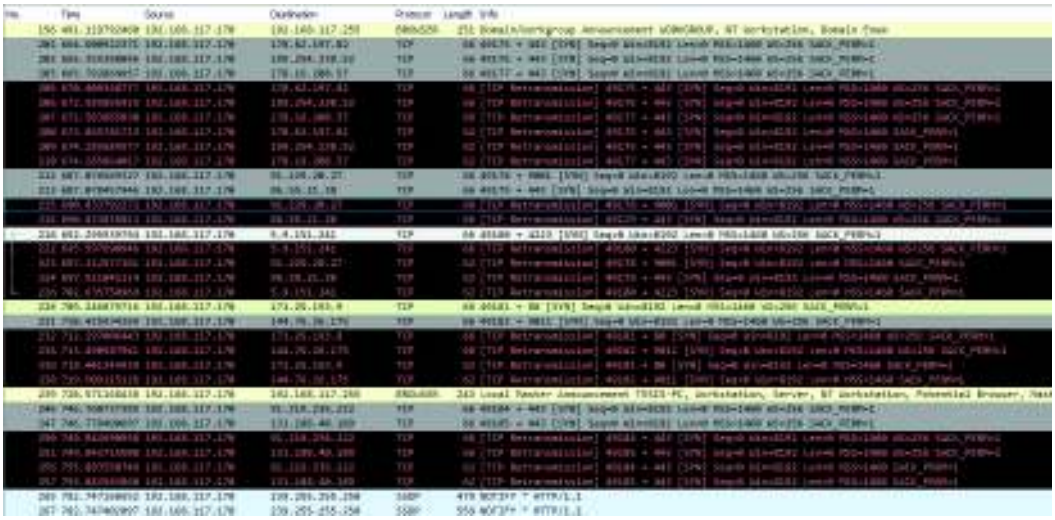

Figura 3.128. Tráfico realizado por WannaCry

A continuación, en la Tablar 3.17 se reúne las direcciones IP con las que el malware intenta establecer una comunicación, y los respectivos puertos destino

Tabla 3.17. Peticiones realizadas por WannaCry

Dirección IP destino	Puerto destino
178.62.197.82	443
199.254.238.52	443
178.16.208.57	443
91.229.20.27	9001
86.59.21.38	443
5.9.151.241	4223
144.76.26.175	9011
171.25.193.9	80
91.216.236.222	443
131.188.40.189	443
194.109.206.212	443

Se realizó un análisis de reputación de las URLs mencionadas utilizando la herramienta online VirusTotal.com, en el que se buscará el número de herramientas antivirus que reportan la dirección IP como maliciosa con respecto al número total de búsquedas, y además la calificación que les es dada por parte de usuarios de la herramienta Virustotal. Los resultados se visualizan en la Tabla 3.18.

Tabla 3.18. Reputación de IPs destino, por parte de WannaCry

URL	Detecciones	Calificación de la comunidad
178.62.197.82	1/67	N/A
199.254.238.52	0/70	-36
178.16.208.57	0/67	N/A
91.229.20.27	0/67	N/A
86.59.21.38	1/67	-3
5.9.151.241	0/67	N/A
144.76.26.175	0/67	N/A
171.25.193.9	3/67	-38
91.216.236.222	0/70	N/A
131.188.40.189	2/70	-1
194.109.206.212	1/67	-65

- **TcpDump**

Al realizar un escaneo del puerto 137, se encuentra que realiza peticiones a la IP 192.168.117.255 que es NetBios como se observa en la Figura 3.129.

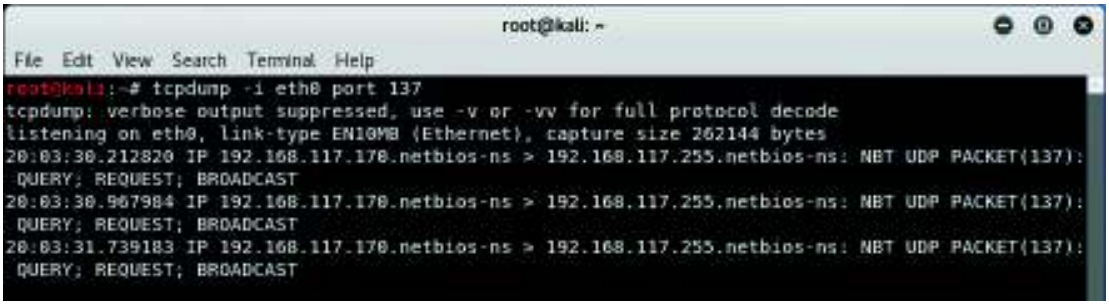

Figura 3.129. Peticiones por hacia el puerto 137 por el equipo infectado con WannaCry

Al realizar un escaneo del puerto 443, se encuentra que realiza peticiones como se observa en la Figura 3.130 a las IPs 154.35.175.225, 5.9.146.203 y 217.12.199.208, las cuales se encuentran como IPs maliciosas en Virus Total.

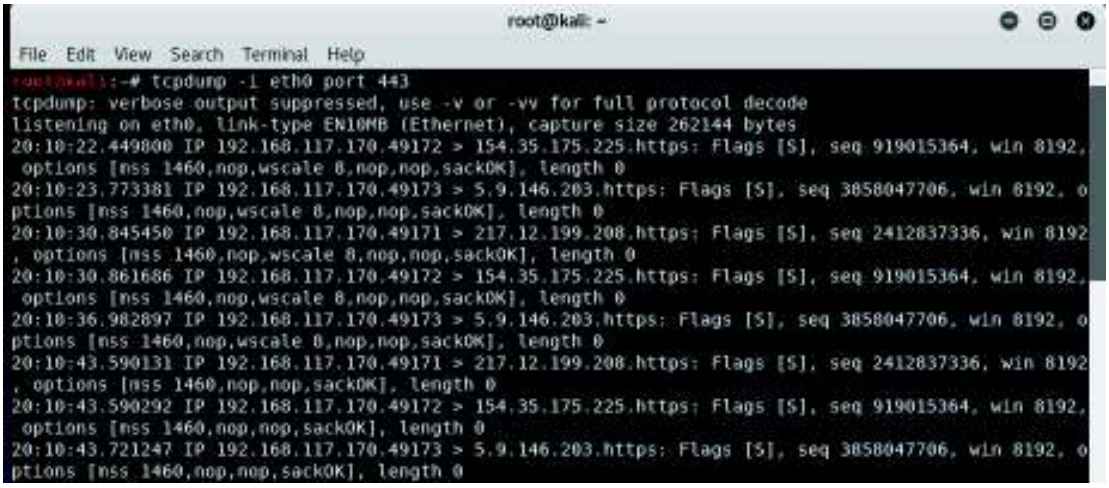

Figura 3.130. Peticiones por hacia el puerto 443 por el equipo infectado con WannaCry

Al realizar un escaneo del puerto 9001, se encuentra que realiza peticiones a las IPs 192.87.28.82, 37.221.162.226 7 138.201.130.32 cómo se puede observar en la Figura 3.131.

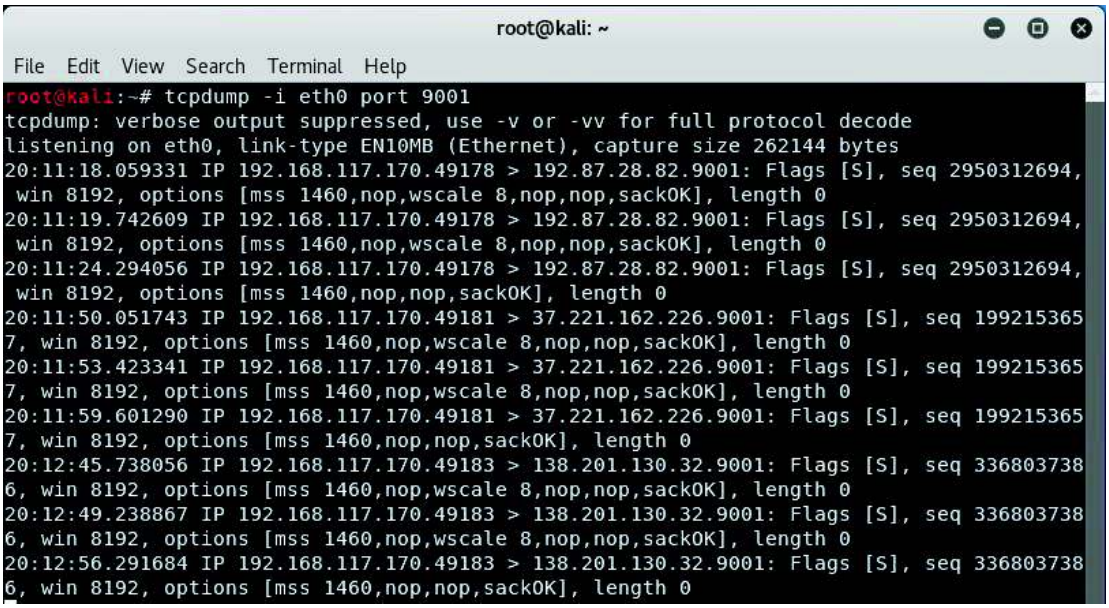

Figura 3.131. Peticiones por hacia el puerto 9001 por el equipo infectado con WannaCry

De esta manera se reafirma la información encontrada con WireShark.

3.3. Comparación de resultados

Comparación del comportamiento de las muestras

Se realiza una comparación del comportamiento de las muestras analizadas, las cuales pueden ser detectadas por un usuario final. Los resultados se indican en la Tabla 3.19.

Tabla 3.19. Comportamiento del malware

COMPORTAMIENTO	CTB	LOCKY	SPORA	WANNACRY
Abre un navegador		X		
Muestra mensaje de recuperar archivos en el escritorio	X	X	X	X
Intento de ejecución de alguna aplicación	X		X	X
Cambia extensión de archivos	X	X		X
Cambia nombre de los archivos	X	X		
Solicita pago por rescate	X		X	X
Crea varios archivos a la vista del usuario				X
Bloquea visualización de escritorio			X	
Causa molestias en el trabajo con el equipo			X	X
Alto consumo del disco duro	X	X	X	X
Consumo de los recursos del equipo	X	X	X	X

En el Anexo II, se puede evidenciar una tabla comparativa de los tipos de archivos que fueron cifrados durante la infección.

Comparación de CTBLocker con un trabajo previo

En esta sección se realiza una comparación entre los resultados obtenidos en este estudio técnico y el proyecto de titulación “Análisis digital de una infección de malware en sistemas Windows” realizado por Diego Arce. Los resultados se indican en la Tabla 3.20.

Tabla 3.20. Comparación de características CTBLocker

CARACTERÍSTICAS	TRABAJO ACTUAL	TRABAJO COMPARADO
Extensiones modificadas	.cpp, .dbf, .eps, .jpg, .ods, .odt, .pdf, .ppt, .psd, .raw, .rtf, .txt, .xls, .doc, .zip	pdf, .cer, .der, .zip, .txt, .doc, .pdf, .config, .js, .jpg, .eps, .ppt, .xls, .c, bs, config
Cambio de extensión	.ifevzjb	.kgkpflh
Mensaje de aviso de cifrado	El mensaje se muestra como un fondo de pantalla en el equipo infectado	Se muestra un programa al arrancar el equipo que tiene un reloj que indica el tiempo restante para rescatar la información
Veracidad del proceso ejecutado por el malware	No tiene firma, indicando que no es un legítimo	No tiene firma, indicando que no es un legítimo
Nombre del proceso principal	Stagnantness5.exe	Stagnantness5.exe

4. CONCLUSIONES

A continuación, se expondrán las conclusiones obtenidas una vez finalizado el estudio técnico en base a los objetivos planteados inicialmente:

De manera general se concluye que:

- Al seguir recomendaciones de buenas prácticas en cuanto a la seguridad informática se puede mitigar el riesgo de infecciones, adoptando parámetros simples que ayudan a proteger la información.

- El análisis de malware mostró mayor complejidad en muestras recientes con respecto a muestras creadas con anterioridad. Las muestras más recientes exhibieron mayor agresividad, una gran cantidad de información a analizar y técnicas más sofisticadas para ocultar su huella en el equipo infectado. Pronosticando así una evolución en futuras familias de malware.

Por otro lado, a nivel específico este estudio permite concluir que:

- Los reportes elaborados por las compañías dedicadas a la seguridad informática, constituyen una herramienta importante para la evaluación actual y la predicción de la evolución de incidentes relacionados con ataques de malware.

- Aunque existen varios tipos de Ransomware, el que ha tenido mayor impacto en Ecuador es aquel cuyo objetivo es encriptar la información con el fin de pedir un rescate para la recuperación de la misma; siendo un caso específico WannaCry el cual constituye un 23.48% de las muestras reportadas en el ranking realizado en el presente estudio técnico.

- Implementar un ambiente virtual seguro, siguiendo una metodología adecuada, permitió evitar posibles infecciones en el equipo físico, así como también una propagación del malware hacia otros equipos vinculados a la red.

- La toma de huellas en el análisis estático constituyó un proceso útil para buscar información en línea sobre el archivo a analizarse y de esta manera determinar si el mismo era malicioso.

- En el análisis estático, los nombres de las funciones extraídas de la toma de cadena de caracteres, así como de librerías de las que dependían las muestras, constituyen información útil para especular sobre la funcionalidad del malware analizado.

- Es importante que, al momento de realizar un análisis dinámico el equipo infectado tenga acceso a una red local simulada, puesto que de esta manera el malware intentará realizar con normalidad las peticiones que tiene programadas, logrando así analizar una mayor cantidad de funcionalidades con respecto a un equipo sin conexiones de red.

- Mediante el análisis dinámico se pudo observar la cantidad de tipos de archivos que fueron encriptados cuantificando el imparto que tuvieron las muestras de malware, siendo WannaCry el que tuvo mayor cantidad de cambios registrados modificando más de 3200 archivos y encriptando archivos con 40 de las 68 extensiones de prueba.

- De las herramientas antimalware utilizadas, todas detectaron las muestras analizadas como archivos maliciosos, cada una bajo nombres distintos. Las soluciones gratuitas utilizadas demostraron la misma efectividad que una de pago.

- Las soluciones de Sandbox entregan la información de una manera clasificada y es labor del analista de malware interpretar adecuadamente los datos proporcionados por estas soluciones.

- Se observó que es un comportamiento común entre las familias de Ransomware, eliminar copias de seguridad del sistema operativo con el fin de prevenir una restauración total del equipo afectado.

- De las cuatro muestras analizadas, todas presentaron cambios en el escritorio del equipo, indicando la infección del mismo y dando indicaciones para el rescate de la información comprometida. Spora fue la única muestra que presentó un

componente lockscreen imposible de minimizarse o cerrarse impidiendo la visualización completa del escritorio del equipo afectado.

- Los autores del presente estudio técnico por medio de un previo análisis estático y dinámico pudieron familiarizarse con el comportamiento, características y funcionalidades de muestras de malware tipo Ransomware, lo que permitió la formulación de recomendaciones de buenas prácticas con el fin de evitar infecciones que puedan causar la pérdida parcial o total de la información del usuario final, así como también ocasional un mal funcionamiento del equipo.

5. REFERENCIAS BIBLIOGRÁFICAS

[1] Kaspersky Lab, «KASPERSKY SECURITY BULLETIN 2015,» Great, 2015.

[2] M. Garnaeva, F. Sinitsyn, Y. Namestnikov, D. Makrushin and A. Liskin, "OVERALL STATISTICS FOR 2016," GREAT, 2016.

[3] Kaspersky Lab, "OVERALL STATISTICS FOR 2017," 2017.

[4] ESET, «ESET Security Report Latinoamérica 2017,» 2017.

[5] A. Korsakov, Cryptovirology and Malicious Software, University of Eastern Finland.

[6] A. Perakalin, «WannaCry: Are you safe?,» Kaspersky lab., 16 05 2017. [En línea]. Available: https://www.kaspersky.com/blog/wannacry-ransomware/16518/. [Último acceso: 3 Julio 2018].

[7] SANS Institute Reading Room, «SANS,» 14 12 2007. [En línea]. Available: https://www.sans.org/reading-room/whitepapers/malicious/malware-analysis-introduction-2103. [Último acceso: 18 1 2018].

[8] Web Security, «Websecurity.es,» [En línea]. Available: www.websecurity.es/que-es-un-ataque-denegaci-n-servicio. [Último acceso: 3 Julio 2018].

[9] Kaspersky Lab, «Kaspersky Lab,» [En línea]. Available: https://www.kaspersky.com/blog/rootkit/1508/1508/. [Último acceso: 12 Marzo 2018].

[10] J. M. Kizza, Guide to Computer Network Security, Chattanooga: Springer, 2015.

[11] Borton by Symantec Corporation, «Norton,» Symantec Corporation, [En línea]. Available: https://us.norton.com/internetsecurity-malware.html. [Último acceso: 1 Junio 2018].

[12] W. Stallings, Cryptography and Network Security Principles and Practice, New Jersey: Pearson, 2014.

[13] L. Z. Ed Skoudis, Malware: Fighting Malicious Code, Prentice Hall PTR, 2003.

[14] Kaspersky Lab, «Kaspersky Lab,» [En línea]. Available: https://usa.kaspersky.com/resource-center/definitions/social-engineering.

[15] Cisco, «Cisco,» [En línea]. Available: https://www.cisco.com/c/en/us/about/security-center/virus-differences.html#adware. [Último acceso: 20 Abril 2018].

[16] Kaspersky Lab, «Kaspersky Lab,» [En línea]. Available: https://usa.kaspersky.com/resource-center/threats/ransomware. [Último acceso: 20 Abril 2018].

[17] M. . Á. Mendoza, «We live security,» Eset, [En línea]. Available: https://www.welivesecurity.com/la-es/2018/03/01/impacto-ransomware-latinoamerica-2017/. [Último acceso: 3 Julio 2018].

[18] McAfee Labs, «2016 Threats Predictions,» McAfee Labs, 2015.

[19] Kaspersky Lab, «STORY OF THE YEAR 2017,» Kaspersky Lab, 2017.

[20] R. Unuchek, «Mobile malware evolution 2017,» Kaspersky Lab, 2018.

[21] Eset, «Eset,» [En línea]. Available: https://support.eset.com/kb2563/?locale=en_US&viewlocale=es_ES. [Último acceso: 13 Mayo 2018].

[22] A. Stern, «Kaspersky Lab,» [En línea]. Available: https://www.kaspersky.es/blog/10-sintomas-de-una-infeccion-maliciosa/1348/. [Último acceso: 13 Mayo 2018].

[23] I. Rijnetu, «Heimdal Security,» [En línea]. Available: https://heimdalsecurity.com/blog/warning-signs-operating-system-infected-malware/. [Último acceso: 14 Mayo 2018].

[24] M. Sikorski y A. Honig, Practical Malware Analysis, San Francisco: no starch press, 2012.

[25] S. Yusirwa, Y. Prayudi y I. Riadi, «Implementation of Malware Analysis using Static and Dynamic Analysis Method,» *International Journal of Computer Applications,* vol. 117, nº 6, p. 11, 2016.

[26] L. Zeltser, «Digital Forensics and Incident,» SANS Institute, 2014.

[27] C. Lomont, «Introduction to x64 Assembly,» Intel Software, 19 Marzo 2012. [En línea]. Available: https://software.intel.com/en-us/articles/introduction-to-x64-assembly. [Último acceso: 12 Junio 2018].

[28] T. Skybakmoen y M. Dhanraj, «ADVANCED ENDPOINT PROTECTION,» NSS Labs, 2017.

[29] NSS LABS, «Security Value Map Advanced Endpoint Protection (AEP),» NSS LABS, 2018.

[30] Gartner, «Gartner,» [En línea]. Available: https://www.gartner.com/en/research/methodologies/magic-quadrants-research. [Último acceso: 21 07 2018].

[31] Hybrid Analysis, «Hybrid Analysis,» [En línea]. Available: https://www.hybrid-analysis.com. [Último acceso: 21 Agosto 2018].

[32] Kaspersky , «Secure List,» [En línea]. Available: https://securelist.lat/statistics. [Último acceso: 21 Agosto 2018].

[33] Virus Total, «Virus Total,» [En línea]. Available: https://www.virustotal.com. [Último acceso: 21 Agosto 2018].

[34] D. Distler, Malware Analysis: An introduction, 2013.

[35] C. Nguyen y J. Goldman, «Malware Analysis Reverse Engineering (MARE) Methology & Malware Defense Timeline,» 2011.

[36] B. Schneier, «Schneier on Security,» [En línea]. Available: https://www.schneier.com/essays/archives/2004/08/cryptanalysis_of_md5.html. [Último acceso: 10 Agosto 2018].

[37] Microsoft, «Microsoft,» [En línea]. Available: https://msdn.microsoft.com/en-us/library/ms235265.aspx. [Último acceso: 11 Agosto 2018].

[38] T. Fenster , «GitHub,» [En línea]. Available: https://github.com/lucasg/Dependencies/blob/master/README.md. [Último acceso: 12 Agosto 2018].

[39] Inetsim, «Inetsim,» [En línea]. Available: https://www.inetsim.org/about.html. [Último acceso: 13 Agosto 2018].

6. ANEXOS

ANEXO I	Listado de las librerías y sus funciones que fueron llamadas por las muestras analizadas.
ANEXO II	Listado de todos los tipos de archivos y sus extensiones que fueron modificados por las distintas muestras analizadas
ANEXO III	Recomendaciones de buenas prácticas informáticas dirigida a usuarios finales de estaciones de trabajo con sistema operativo Windows.

ANEXO I

En el Anexo I se encuentra el listado de las librerías y sus funciones que son utilizadas por cada una de las muestras analizadas.

ETIQUETAS DE FILA	CTBLOCKER	LOCKY	SPO-RA	WANNACRY
KERNEL32.DLL				
CloseHandle	X		X	X
CopyFileA				X
CreateDirectoryA				X
CreateDirectoryW				X
CreateFileA				X
CreateFileMappingA		X		
CreateFileW			X	
CreateMailslotA		X		
CreateProcessA				X
DecodePointer			X	
DeleteCriticalSection			X	X
DeleteFileW		X		
EncodePointer			X	
EnterCriticalSection			X	X
ExitProcess			X	
FindResourceA				X
FlushFileBuffers			X	
FreeEnvironmentStringsW			X	
FreeLibrary				X
GetACP		X	X	
GetBinaryTypeA	X			
GetCommandLineA		X	X	
GetComputerNameA	X			
GetComputerNameW				X
GetConsoleAliasA		X		
GetConsoleAliasW	X			
GetConsoleCP			X	
GetConsoleMode			X	
GetCPInfo			X	
GetCurrentDirectoryA				X
GetCurrentProcess	X		X	
GetCurrentProcessId			X	
GetCurrentThreadId			X	
GetDateFormatA		X		

GetEnvironmentStringsW			X	
GetEnvironmentVariableA	X			
GetExitCodeProcess				X
GetFileAttributesA				X
GetFileAttributesW		X		X
GetFileSize				X
GetFileSizeEx				X
GetFileType			X	
GetFullPathNameA				X
GetGeoInfoA	X			
GetLastError			X	
GetLogicalDriveStringsW		X		
GetLongPathNameA	X			
GetModuleFileNameA			X	X
GetModuleFileNameW		X	X	
GetModuleHandleA		X	X	X
GetModuleHandleExW			X	
GetModuleHandleW			X	
GetOEMCP			X	
GetPrivateProfileStructW	X			
GetProcAddress	X	X	X	X
GetProcessHeap			X	X
GetProcessId	X			
GetStartupInfoA				X
GetStartupInfoW			X	
GetStdHandle			X	
GetStringTypeA	X			
GetStringTypeW			X	
GetSystemTimeAdjustment			X	
GetSystemTimeAsFileTime			X	
GetTempPathW				X
GetThreadSelectorEntry			X	
GetTickCount			X	
GetTimeFormatA	X			
GetWindowsDirectoryW				X
GlobalAlloc				X
GlobalFree				X
HeapAlloc			X	X
HeapFree			X	X
HeapReAlloc			X	
HeapSize			X	
HeapValidate	X			
InitializeCriticalSection		X		X
InitializeCriticalSection			X	

AndSpinCount				
IsBadReadPtr				X
IsDebuggerPresent			X	
IsProcessorFeaturePresent			X	
IsValidCodePage			X	
LCMapStringW			X	
LeaveCriticalSection			X	X
LoadLibraryA	X	X		X
LoadLibraryExW			X	
LoadResource				X
LocalFileTimeToFileTime				X
LockResource				X
lstrcmpiA		X		
lstrcpynA	X			
MultiByteToWideChar			X	X
OpenMutexA				X
OutputDebugStringW			X	
QueryPerformanceCounter			X	
RaiseException			X	
ReadConsoleA	X			
ReadConsoleW		X		
ReadFile				X
RtlUnwind			X	
SearchPathA		X		
SetCurrentDirectoryA				X
SetCurrentDirectoryW				X
SetEnvironmentVariableW	X			
SetErrorMode		X		
SetFileAttributesW				X
SetFilePointer				X
SetFilePointerEx			X	
SetFileTime				X
SetLastError			X	X
SetStdHandle			X	
SetUnhandledExceptionFilter			X	
SizeofResource				X
Sleep			X	X
SystemTimeToFileTime				X
TerminateProcess			X	X
TlsAlloc			X	
TlsFree			X	
TlsGetValue			X	
TlsSetValue			X	
UnhandledExceptionFilter			X	

UpdateResourceA	X			
VirtualAlloc				X
VirtualFree				X
VirtualProtect				X
WaitForSingleObject	X	X		X
WideCharToMultiByte			X	
WriteConsoleW			X	
WriteFile			X	X
ADVAPI32.DLL				
ChangeServiceConfigA			X	
ClearEventLogA		X	X	
CloseServiceHandle			X	X
ControlService		X	X	
CreateServiceA				X
CryptReleaseContext				X
CryptSignHashW		X		
InitializeAcl		X		
LogonUserW		X		
OpenEventLogW		X		
OpenSCManagerA				X
OpenServiceA				X
RegCloseKey				X
RegCreateKeyExW		X		
RegCreateKeyW				X
RegDeleteValueA		X		
RegEnumKeyW		X		
RegLoadKeyA		X		
RegOpenKeyA		X		
RegQueryValueExA				X
RegReplaceKeyW		X		
RegSetValueExA				X
RegUnLoadKeyA		X		
StartServiceA				X
MSIMG32.DLL				
AlphaBlend	X			
DllInitialize	X			
GradientFill	X			
TransparentBlt	X			
WTSAPI32.DLL				
WTSEnumerateProcessesA	X			
WTSFreeMemory	X			
WTSLogoffSession	X			
WTSOpenServerW	X			
WTSQuerySessionInformationA	X			

WTSQueryUserToken	X			
WTSRegisterSessionNotification	X			
WTSSendMessageA	X			
WTSSetUserConfigW	X			
WTSUnRegisterSessionNotification	X			
WTSVirtualChannelClose	X			
WTSVirtualChannelPurgeInput	X			
WTSVirtualChannelRead	X			
WTSVirtualChannelWrite	X			
SHLWAPI.DLL				
PathCompactPathA	X			
UrlCanonicalizeA	X			
UrlCombineA	X			
UrlCompareA	X			
UrlCreateFromPathA	X			
UrlEscapeA	X			
UrlGetLocationA	X			
UrlHashA	X			
UrlIsA	X			
UrlIsNoHistoryW	X			
UrlIsOpaqueA	X			
UrlUnescapeA	X			
NDDEAPI.DLL				
NDdeShareAddA	X			
NDdeShareEnumA	X			
NDdeShareSetInfoA	X			
MPRAPI.DLL				
MprAdminBufferFree		X		
MprAdminConnectionEnum		X		
MprInfoBlockAdd		X		
SHELL32.DLL				
DllRegisterServer		X		
DragFinish			X	
ExtractIconW		X		
FindExecutableW		X		
SHCreateShellItem		X		
ShellAboutW		X		
SHFree		X		
SHGetFileInfoW		X		
SHGetFolderPathA		X		
StrChrA		X		
StrStrA		X		
USER32.DLL				
CharToOemA		X		

CloseClipboard			X	
CreateDesktopW		X		
DialogBoxParamA		X		
DispatchMessageW		X		
DrawStateW		X		
GetAltTabInfoA			X	
GetClassLongW		X		
GetDialogBaseUnits			X	
GetDlgCtrlID			X	
GetDlgItemTextW		X		
GetMenuState			X	
GetMessageA		X		
GetNextDlgGroupItem			X	
GetNextDlgTabItem			X	
GetPropA		X		
GetRegisteredRawInputDevices			X	
InsertMenuW		X		
IsDialogMessageA		X		
LoadAcceleratorsA			X	
LoadBitmapA			X	
LoadBitmapW		X		
LoadCursorA			X	
LoadCursorFromFileA			X	
LoadIconW		X	X	
LoadMenuA		X	X	
LoadMenuIndirectA			X	
LoadStringA			X	
LoadStringW		X		
LockWindowUpdate			X	
LookupIconIdFromDirectory			X	
LookupIconIdFromDirectoryEx			X	
MapDialogRect			X	
MapVirtualKeyA			X	
MapVirtualKeyExA			X	
MapWindowPoints			X	
PostMessageA		X		
RegisterRawInputDevices			X	
wsprintfA		X		X
GDI32.DLL				
PlayMetaFileRecord			X	
SetPixel			X	
SetPolyFillMode			X	
SetStretchBltMode			X	
StretchBlt			X	

StretchDIBits			X	
MSVCRT.DLL				
??0exception@@QAE @ABQBD@Z				X
??0exception@@QAE @ABV0@@Z				X
??1exception@@UAE@XZ				X
??1type_info@@UAE@XZ				X
??2@YAPAXI@Z				X
??3@YAXPAX@Z				X
_CxxThrowException				X
_XcptFilter				X
__CxxFrameHandler				X
__getmainargs				X
__p___argc				X
__p___argv				X
__p__commode				X
__p__fmode				X
__set_app_type				X
__setusermatherr				X
_acmdln				X
_adjust_fdiv				X
_controlfp				X
_except_handler3				X
_exit				X
_initterm				X
_local_unwind2				X
_mbsstr				X
_stricmp				X
calloc				X
exit				X
fclose				X
fopen				X
fread				X
free				X
fwrite				X
malloc				X
memcmp				X
memcpy				X
memset				X
rand				X
realloc				X

ANEXO II

En el Anexo II se encuentra el listado de todos los tipos de archivos y sus extensiones que fueron modificados por las distintas muestras analizadas

DESCRIPCIÓN	EXTENSIÓN	CTB	LOCKY	SPORA	WANNA
Assembly x86	asm		X		X
Batchfile	bat		X		X
Windows bitmap	bmp		X		X
c_cpp	cpp	X	X		X
Comma-Separated Values	csv		X		X
DataBase File	dbf	X	X		X
Data Interchange Format	dif				X
Microsoft Document	doc	X		X	X
Open XML Microsoft document	docx	X	X	X	X
Ecapsulated PostScript	eps	X	X		
Graphics Interchange Format	gif		X		X
HyperText Markup Language	html		X		
Interchange File Format	iff		X		
Joint Photographic Experts Group	jpg	X	X	X	X
Open Office Spreadsheet	ods	X	X		X
Open Office Text Document	odt	X	X		X
Open Office Spreadsheet Template	ots		X		X
Open Office Text Template	ott		X		X
Netpbm format	pbm		X		
Macintosh picture metafile	pct		X		
Portable Document Format	pdf	X	X		X
Portable Network Graphics	png		X	X	X
Microsoft PowerPoint	ppt	X	X	X	X
Open XML Microsoft PowerPoint	pptx	X	X	X	X
Photoshop document	psd	X	X		X
Python programming language	py		X		
Photoshop Raw File	raw	X	X		X
Rich Text Format	rtf	X	X		X
SPSS Data	sav		X		
SYmbolic LinK	slk		X		X
Open Office XML Spreadsheet Template	stc		X		X
Open Office XML Text Template	stw		X		X
Scalable Vector Graphics	svg		X		X

Open Office XML Spreadsheet	sxc		X		X
Open Office XML Text Document	sxw		X		X
Truevision TGA	tga		X		
Tagged Image File Format	tif		X		X
Text File	txt	X	X		X
Excel Binary	xls	X	X		X
Open XML Microsoft spreadsheet	xlsx	X	X	X	X
Microsoft Word 2003 XML	xml		X	X	
Uniform Office Format Text	uot		X		X
Uniform Office Format 2 Text	uot		X		X
Microsoft 6.0 Document	doc	X	X	X	X
Microsoft 95 Document	doc	X	X	X	X
Microsoft 97, 2000, and XP Document	doc	X	X	X	X
AportisDoc (Palm)	pdb		X		
Microsoft Word 2003 XML	txt	X	X	X	X
Microsoft Word 2003 XML	xml		X	X	
Compresión	zip	X	X		X

ANEXO III

RECOMENDACIONES DE BUENAS PRÁCTICAS INFORMÁTICAS DIRIGIDA A USUARIOS FINALES DE ESTACIONES DE TRABAJO CON SISTEMA OPERATIVO WINDOWS.

Una vez realizado el estudio y análisis de las muestras de malware, se detectaron varias vulnerabilidades, las cuales pueden ser corregidas mediante buenas prácticas de seguridad informática. A continuación, se detallan varias recomendaciones con el fin de evitar infecciones que puedan causar la pérdida parcial o total de la información del usuario final, así como también ocasional un mal funcionamiento del equipo.

A nivel de sistema operativo

- Asegurarse que el sistema operativo adquirido sea original, para de esta manera garantizar que todas las propiedades de seguridad de Windows se encuentran activadas.

- Mantener activadas tanto las descargas como instalación Windows Update, puesto que las actualizaciones, parches y service packs ayudan a corregir vulnerabilidades del sistema a medida que estas se van descubriendo. La incapacidad para obtener actualizaciones hace que el sistema sea vulnerable ante ataques que pudiesen haber sido evitados.

- Utilizar sistemas operativos que no se encuentren obsoletos, puesto que, al no recibir soporte constante, dichos sistemas operativos pueden considerarse vulnerables a partir de la fecha de detención de soporte de Microsoft.

A nivel de uso del equipo

- Tener habilitado la visualización de las extensiones de los archivos, puesto que muchas variantes de software malicioso utilizan técnicas como la doble extensión, para hacer pasar un archivo ejecutable como un documento benigno.

- Respaldar la información más importante de manera periódica, para prevenir la pérdida total de la información en caso de que se llega a sufrir una infección.

- Configurar un perfil de usuario del equipo con privilegios limitados para su uso cotidiano, ya que el empleo habitual de cuentas de usuario con privilegios de administrador incrementa la probabilidad de infecciones.

A nivel de software adicional

- Tener instalado un antimalware original, ya sea de versión gratuita o licenciada. En caso de provenir de descargas, el antimalware debe ser obtenido únicamente desde las páginas oficiales de la solución. De no desearse realizar la instalación de una herramienta antimalware complementaria, Windows ofrece un sistema de protección contra malware llamado Windows Defender, el cual es distribuido y activado por defecto con sus sistemas operativos a partir de Windows Vista.

- Utilizar software original, evitando la descarga de los mismo a partir de sitios web que ofrecen contenido de manera ilegal, puesto que muchas veces estos sitios sirven para la propagación de software malicioso.

- Mantener respaldos de la información en soluciones en la nube, muchas de estas soluciones permiten la recuperación de estados previos de los datos del usuario. Como por ejemplo el caso de Dropbox, en el cual la información puede ser recuperada hasta 30 días después de que la misma haya sido comprometida.

A nivel de red

- Es necesario tener integrado al navegador web de preferencia un complemento para el bloqueo de publicidad, de esta manera se evita que el navegador abra ventanas emergentes las cuales pueden estar enlazadas a contenido malicioso.

- No acceder a enlaces que se encuentran en el cuerpo de los correos electrónicos, ya muchos de ellos suelen estar enmascarados sobre otra URL la cual lleva a sitios maliciosos.

- En caso de que un equipo ya se encuentre infectado, es primordial retirarlo de la red local para de esta manera prevenir la infección de otros equipos conectados a la misma red.

ORDEN DE EMPASTADO