ANÁLISIS FORENSE DE MALWARE

TRABAJO DE GRADO

ROBERT VELOZA GONZALEZ

INSTITUCIÓN UNIVERSITARIA POLITÉCNICO GRANCOLOMBIANO

FACULTAD DE INGENIERÍA Y CIENCIAS BÁSICAS

ESPECIALIZACIÓN EN SEGURIDAD DE LA INFORMACIÓN

2017

ANÁLISIS FORENSE DE MALWARE

TRABAJO DE GRADO

ROBERT VELOZA GONZALEZ

Asesor:

Alejandro Castiblanco

INSTITUCIÓN UNIVERSITARIA POLITÉCNICO GRANCOLOMBIANO

FACULTAD DE INGENIERÍA Y CIENCIAS BÁSICAS

ESPECIALIZACIÓN EN SEGURIDAD DE LA INFORMACIÓN

2017

A mi madre,

Y mi hija Alejandra

Con todo mi amor.

Nota de aceptación

_____

_____

_____

_____

_____

_____

_____

_____

_____

Firmas de los jurados

Bogotá, 05/Mayo/2017

CONTENIDO

Pág

1 INTRODUCCIÓN

Definitivamente internet se ha convertido en una herramienta cotidiana de uso masivo en donde se encuentra alojada una gran cantidad de información fundamental para el ser humano en pro de su crecimiento intelectual; pero al transcurrir de los años internet se ha desarrollado para facilitar la comunicación entre cualquier ser humano.

Internet también es una herramienta que facilita las tareas de cualquier usuario como por ejemplo transacciones bancarias, intercambio de información personal con las redes sociales entre otras, pero esta importante herramienta se ha convertido en un medio delincuencial bastante concurrido y con efectividad alta.

Las organizaciones pueden presentar incidentes informáticos que puede repercutir en daños a sus activos informáticos, fuga de información confidencial o denegación de servicio de activos corporativos importantes, es allí donde los analistas forenses realizan una tarea importante para verificar como se produjo el incidente informático y que mecanismos utilizaron los delincuentes informáticos para ingresar al sistema o al perímetro informático.

En esta ocasión realizaremos un análisis forense de una pieza de malware en un ambiente simulado con técnicas forenses actuales recorriendo y describiendo todos los pasos para determinar la ocurrencia del caso.

2 RESUMEN EJECUTIVO

El trabajo de grado está diseñado para ayudar a los profesionales de la seguridad a desarrollar una estrategia para atender un incidente informático en una organización que involucre malware.

Es de interés para los administradores de recursos de información, los directores de seguridad informática y los administradores, y tiene un valor especial para todos aquellos que intentan estrategias para incident response en una organización.

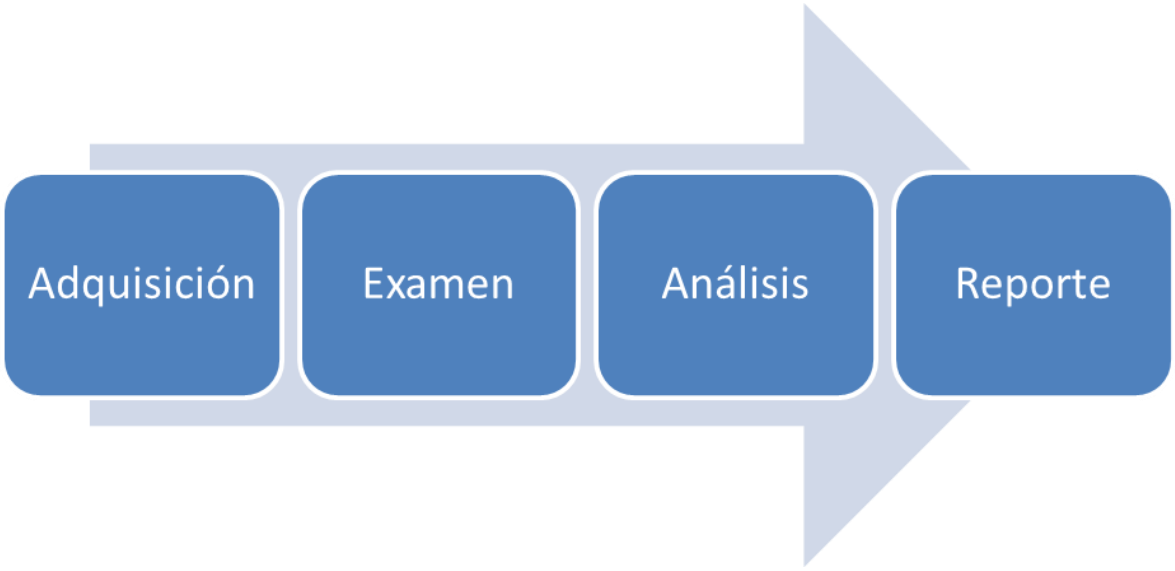
La realización de ejercicios de incident response que permita determinar la naturaleza de los ataques informáticos permite aprender a conocer nuevas técnicas de ataque, conocer en mayor medida el conocimiento del funcionamiento de los sistemas informáticos, establecer procedimientos internos de análisis forense y diagnosticar la ocurrencia de los ataques informáticos.

La carencia de laboratorios y equipos de pruebas a causa de restricciones presupuestarias puede imposibilitar la realización de ataques simulados de gran escala, adicionalmente las herramientas forenses pueden tener un alto costo como por ejemplo Encase. Para asegurar los fondos necesarios para las pruebas, es importante que los directivos sean conscientes de los riesgos y consecuencias de los ataques, así como de las medidas de seguridad que se pueden adoptar para proteger al sistema, incluidos los procedimientos de las pruebas.

Si es posible, se deben probar físicamente y documentar todos los casos de ataque para determinar las mejores directivas y controles de seguridad posibles

que se van a implementar de acuerdo a las lecciones aprendidas obtenidas en cada incidente.

Para realizar este trabajo de grado se utilizó como guía base la referencia de NIST SP 800-86 Guide To Integrating Forensic Techniques Into Incident Response en la cual realizamos 4 fases:



Bajo estos pasos realizamos el análisis de una muestra de malware de manera estática, adicionalmente se realiza una matriz con las herramientas importantes que se pueden realizar para el análisis forense de malware.

En el presente proyecto utilizamos una muestra de malware de un troyano que obtiene una alta cantidad de información de su víctima, se evidencia procesos y métodos del troyano y como se engaña a la víctima para su infección.

Esperamos este trabajo de investigación y análisis de un alto contenido técnico sea del agrado de los lectores y esperamos que sea una motivación para enriquecer la investigación forense en el país.

3 JUSTIFICACIÓN

Un ataque informático consiste en aprovechar alguna debilidad o falla en el software; el ataque de ser efectivo puede causar serios problemas para una organización desde todos los frentes, incluso en su imagen para sus clientes y proveedores.

Cada día se descubren nuevos puntos débiles y, por lo general, son pocos los responsables en el área informática que comprenden en su justa medida la importancia que tiene la seguridad y cómo pueden abordar el grave problema que existe detrás de vulnerabilidades que permiten a un atacante ingresar a un perímetro informático y sustraer datos organizacionales sensibles o generar saboteo en los activos informáticos.

Teniendo en cuenta este escenario donde los principales actores son las organizaciones de cualquier magnitud, los sistemas de información, el dinero, y delincuentes informáticos se torna realmente necesario y fundamental idear estrategias de seguridad que permita establecer barreras defensivas orientadas a mitigar efectivamente ataques tanto externos como internos.

Pero para lograr mitigar de manera eficaz el impacto provocado por los ataques informáticos, es de capital importancia conocer de qué manera atacan y cuáles son los puntos débiles de un sistema comúnmente explotado en los que se deben enfocar los esfuerzos de seguridad tendientes a la prevención de los mismos.

Bajo esta investigación se desarrollará un ejercicio de análisis forense con una pieza de malware donde con ayuda de herramientas forenses se diagnosticará su funcionamiento, actividades dentro de la máquina objetivo, y toda evidencia que

permita determinar la ocurrencia del incidente simulado, adicionalmente se detallará las herramientas forenses utilizadas que pueda enriquecer una investigación forense ante una eventualidad que involucre malware.

4 MARCO TEÓRICO

El análisis forense lo podemos definir como “un conjunto de principios y técnicas que comprende el proceso de adquisición, conservación, documentación, análisis y presentación de evidencias digitales”[1], en este caso llevamos el análisis forense al campo de la examinación proactiva de malware en el cual nos preparamos para evaluar paquetes maliciosos y examinar su comportamiento, tal como lo realizan diferentes laboratorios de prestigiosos fabricantes para construir sus patrones de remediación en sus antivirus.

La evidencia digital es la materia prima para los investigadores donde la tecnología informática es parte fundamental del proceso. Sin embargo, y considerando el ambiente tan cambiante y dinámico de las infraestructuras de la computación y comunicaciones, es preciso detallar las características propias de dicha evidencia en este entorno. La evidencia digital, para aquellos que la identifican y analizanen la búsqueda de la verdad, posee entre otros elementos que la hacen un constante desafío, las características siguientes:

- Es volátil
- Es Anónima
- Es duplicable
- Es alternable y modificable
- Es eliminable[2]

[1] DELGADO, Miguel. Análisis Forense Digital, CriptoRed, p5

[2] CANO, Jeimy. Computación Forense, Alfaomega, p22

La estrategia proactiva es un conjunto de pasos predefinidos que deben seguirse para evitar ataques antes de que ocurran. Entre estos pasos se incluye observar cómo podría afectar o dañar el sistema, y los puntos vulnerables.

Estos son los tres pasos de la estrategia proactiva:

- Determinar el daño que causará el ataque.
- Establecer los puntos vulnerables y las debilidades que explotará el ataque.
- Reducir los puntos vulnerables y las debilidades que se ha determinado en el sistema para ese tipo de ataque específico[3].

El seguimiento de estos pasos para analizar los distintos tipos de ataques tiene una ventaja adicional: comenzará a emerger un modelo, ya que en los diferentes factores se superponen para diferentes ataques. Este modelo puede ser útil al determinar las áreas de vulnerabilidad que plantean el mayor riesgo para la empresa. También es necesario tomar nota del costo que supone la pérdida de los datos frente al de la implementación de controles de seguridad.

Las directivas y controles de seguridad no serán, en ningún caso, totalmente eficaces al eliminar los ataques. Éste es el motivo por el que es necesario desarrollar planes de recuperación y de contingencia en caso de que se quebranten los controles de seguridad.

4.1 Determinar el daño posible que puede causar un ataque

Los daños posibles pueden oscilar entre pequeños fallos del equipo y una pérdida, considerable de datos. El daño causado al sistema dependerá del tipo de ataque.

[3] BENSON, Cristopher. Security strategies. Microsoft Technet

Si es posible, utilice un entorno de prueba o de laboratorio para clarificar los daños que provocan los diferentes tipos de ataques. Ello permitirá al personal de seguridad ver el daño físico que causan los ataques experimentales. No todos los ataques causan el mismo daño.

4.2 Proceso de Análisis

El proceso de análisis se puede categorizar de diferentes maneras según las técnicas que se usen. Por ejemplo existen análisis vivos en los cuales se analiza el sistema comprometido online. En este tipo de casos se trata de analizar el contenido volátil (principalmente la memoria) aunque también el no volátil es analizado, ya que cuanta más información se tenga mayor detalle del ataque se sabrá. Cuando se hace el análisis volátil hay que tener mucho cuidado con la confiabilidad de la información que nos devuelve el sistema operativos ya que hay ataques en los que el intruso llega a modificar al kernel y de esta manera puede interceptar nuestras peticiones al SO y ocultar información.

Otro detalle importante del análisis forense es el grado de abstracción de la información recolectada. Aunque uno puede leer bit a bit la evidencia, resulta ineficiente por el tiempo que se tarda en encontrar la información (aunque en algunos casos sea la única manera). Para eso uno se ayuda con herramientas (algunas provistas por el sistema operativo). Hay varias herramientas en el mercado y muchas de ellas libre y otras comerciales.

El análisis forense no se limita solamente a una computadora aislada sino que todo lo contrario. Tener información sobre las conexiones realizadas en una

red o el análisis de varias computadoras a la vez, puede ayudar a reconstruir acciones del intruso[4].

4.3 Técnicas de análisis forense

El análisis de un archivo binario puede dividirse (como en todo análisis forense) en dos partes: El análisis estático y el análisis dinámico.

Haciendo un análisis estático podemos conocer un montón de información del binario y poder predecir el comportamiento. En cambio, el análisis dinámico sirve para entender mejor la interacción con el usuario y el sistema operativo.

El análisis estático implica varias formas inspección que no involucra la ejecución del binario, lo que es realizado por el análisis dinámico.

El análisis dinámico permite hacer el seguimiento o alterar el flujo de ejecución utilizando aplicaciones de monitoreo.

Como el comportamiento del binario es desconocido, es importante aislar la ejecución del mismo. Una buena forma seria creando una máquina virtual con una vmware.

Potencialmente, las técnicas de análisis estático hacen posible conocer todo acerca del binario, mientras que las técnicas de análisis dinámico están restringidas por la capacidad de interactuar presente en el ejecutable. No

[4] Análisis Forense, Fiuba, p6

obstante, en muchas situaciones para complementar un análisis estático completo se requiere además aplicar alguna técnica de análisis dinámico.[5]

4.4 Guías de investigación forense

A continuación se enuncian siete guías existentes a nivel mundial de mejores prácticas en computación forense.

- ✓ **RFC 3227**

El "RFC 3227: Guía Para Recolectar y Archivar Evidencia" (Guidelines for Evidence Collection and Archiving) [GuEvCo02], escrito en febrero de 2002 por Dominique Brezinski y Tom Killalea, ingenieros del Network Working Group.

Es un documento que provee una guía de alto nivel para recolectar y archivar datos relacionados con intrusiones. Muestra las mejores prácticas para determinar la volatilidad de los datos, decidir que recolectar, desarrollar la recolección y determinar cómo almacenar y documentar los datos.

También explica algunos conceptos relacionados a la parte legal. Su estructura es:
a) Principios durante la recolección de evidencia: orden de volatilidad de los datos, cosas para evitar, consideraciones de privacidad y legales.
b) El proceso de recolección: transparencia y pasos de recolección.
c) El proceso de archivo: la cadena de custodia y donde y como archivar[6].

[5] Análisis Forense, Fiuba, p13

[6] ZUCCARDI, Giovanni, Gutiérrez Juan David. Informática Forense, p12

- ✓ **Guía de la IOCE**

La IOCE [IOCE06], publico “Guía para las mejores prácticas en el examen forense de tecnología digital” (Guidelines for the best practices in the forensic examination of digital technology) [IOCE02]. El documento provee una serie de estándares, principios de calidad y aproximaciones para la detección prevención, recuperación, examinación y uso de la evidencia digital para fines forenses.

Cubre los sistemas, procedimientos, personal, equipo y requerimientos de comodidad que se necesitan para todo el proceso forense de evidencia digital, desde examinar la escena del crimen hasta la presentación en la corte. Su estructura es:

a) Garantía de calidad (enunciados generales de roles, requisitos y pruebas de aptitud del personal, documentación, herramientas y validación de las mismas y espacio de trabajo).
b) Determinación de los requisitos de examen del caso.
c) Principios generales que se aplican a la recuperación de la evidencia digital (recomendaciones generales, documentación y responsabilidad).
d) Prácticas aplicables al examen de la evidencia de digital.
e) Localización y recuperación de la evidencia de digital en la escena: precauciones, búsqueda en la escena, recolección de la evidencia y empaquetado, etiquetando y documentación.
f) Priorización de la evidencia.
g) Examinar la evidencia: protocolos de análisis y expedientes de caso.
h) Evaluación e interpretación de la evidencia
i) Presentación de resultados (informe escrito).
j) Revisión del archivo del caso: Revisión técnica y revisión administrativa.

k) Presentación oral de la evidencia.
l) Procedimientos de seguridad y quejas. [7]

- ✓ **Investigación en la Escena del Crimen Electrónico (Guía DoJ 1)**

El Departamento de Justicia de los Estados Unidos de América (DoJ EEUU), publico “Investigación En La Escena Del Crimen Electrónico” (Electronic Crime Scene Investigation: A Guide for First Responders) [ElCr01]. Esta guía se enfoca más que todo en identificación y recolección de evidencia. Su estructura es:

a) Dispositivos electrónicos (tipos de dispositivos se pueden encontrar y cual puede ser la posible evidencia).
b) Herramientas para investigar y equipo.
c) Asegurar y evaluar la escena.
d) Documentar la escena.
e) Recolección de evidencia.
f) Empaque, transporte y almacenamiento de la evidencia.
g) Examen forense y clasificación de delitos.
h) Anexos (glosario, listas de recursos legales, listas de recursos técnicos y listas de recursos de entrenamiento)[8].

[7] ZUCCARDI, Giovanni, Gutiérrez Juan David. Informática Forense, p12

[8] ZUCCARDI, Giovanni, Gutiérrez Juan David. Informática Forense, p12

- ✓ **Examen Forense de Evidencia Digital (Guía DoJ 2)**

Otra guía del DoJ EEUU, es “Examen Forense de Evidencia Digital” (Forensic Examination of Digital Evidence: A Guide for Law Enforcement) [FoEx04].

Esta guía está pensada para ser usada en el momento de examinar la evidencia digital. Su estructura es:

a) Desarrollar políticas y procedimientos con el fin de darle un buen trato a la evidencia.
b) Determinar el curso de la evidencia a partir del alcance del caso.
c) Adquirir la evidencia.
d) Examinar la evidencia.
e) Documentación y reportes.
f) Anexos (casos de estudio, glosario, formatos, listas de recursos técnicos y listas de recursos de entrenamiento)[9].

- ✓ **Computación Forense - Parte 2: Mejores Prácticas (Guía Hong Kong)**

El ISFS, Information Security and Forensic Society (Sociedad de Seguridad Informatica y Forense) creada en Hong Kong, publico “Computación Forense Parte 2: Mejores Practicas” (Computer Forensics – Part 2: Best Practices)

[CoFor04] Esta guía cubre los procedimientos y otros requerimientos necesarios involucrados en el proceso forense de evidencia digital, desde el

[9] ZUCCARDI, Giovanni, Gutiérrez Juan David. Informática Forense, p12

examen de la escena del crimen hasta la presentación de los reportes en la corte. Su estructura es:

a) Introducción a la computación forense.
b) Calidad en la computación forense.
c) Evidencia digital.
d) Recolección de Evidencia.
e) Consideraciones legales (orientado a la legislación de Hong Kong).
f) Anexos[10].

- ✓ **Guía De Buenas Prácticas Para Evidencia Basada En Computadores (Guía Reino Unido)**

La ACPO, Association of Chief Police Officers (Asociación de Jefes de Policía), del Reino Unido mediante su departamento de crimen por computador, publico "Guía de Buenas Prácticas para Evidencia basada en Computadores" (Good Practice Guide For Computer Based Evidence) [GoPra99].

La policía creó este documento con el fin de ser usado por sus miembros como una guía de buenas prácticas para ocuparse de computadores y de otros dispositivos electrónicos que puedan ser evidencia.

Su estructura es:
a) Los principios de la evidencia basada en computadores.
b) Oficiales atendiendo a la escena.
c) Oficiales investigadores.

[10] ZUCCARDI, Giovanni, Gutiérrez Juan David. Informática Forense, p12

d) Personal para la recuperación de evidencia basada en computadores.
e) Testigos de consulta externos.
f) Anexos (legislación relevante, glosario y formatos)

Guía Para El Manejo De Evidencia En IT (Guía Australia) Standards Australia (Estándares de Australia) publico “Guía Para El Manejo De Evidencia En IT” (HB171:2003 Handbook Guidelines for the management of IT evidence) [HBIT03].

Esta guía no está disponible para su libre distribución, por esto para su investigación se consultaron los artículos “Buenas Prácticas En La Administración De La Evidencia Digital” [BueAdm06] y “New Guidelines to Combat ECrime” [NeGu03].

Es una guía creada con el fin de asistir a las organizaciones para combatir el crimen electrónico. Establece puntos de referencia para la preservación y recolección de la evidencia digital.

Detalla el ciclo de administración de evidencia de la siguiente forma:
a) Diseño de la evidencia.
b) Producción de la evidencia.
c) Recolección de la evidencia.
d) Análisis de la evidencia.
e) Reporte y presentación.
f) Determinación de la relevancia de la evidencia[11].

[11] ZUCCARDI, Giovanni, Gutiérrez Juan David. Informática Forense, p14

- ✓ **NIST SP800-86**

La serie NIST SP 800 la cual es un conjunto de documentos que facilita desde el gobierno federal de los Estados Unidos, describe directrices en seguridad de la información por el Instituto Nacional de Estándares y Tecnología.

En el documento SP800-86 proporciona la gestión y buenas prácticas para el análisis forense en las etapas de adquisición, examen, análisis y reportes.[12]

4.5 Códigos maliciosos

Los códigos maliciosos o malware constituyen también una de las principales amenazas de seguridad para cualquier institución u organizaciones y aunque parezca un tema trivial, suele ser motivo de importantes pérdidas económicas.

Esta amenaza se refiere a programas que causan algún tipo de daño o anomalía en el sistema informático. Dentro de esta categoría se incluyen los programas troyanos, gusanos, virus informáticos, spyware, backdoors, rootkits, keyloggers, entre otros[13].

Actualmente, casi el 80% de los ataques informáticos llevados a cabo por códigos maliciosos, se realizan a través de programas troyanos[14]. Este tipo de malware

[12] Disponible en http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-86.pdf

[13] GRAVES, Kimberly. Official Certified Ethical Hacker Review Guide. Indianapolis: Wiley Publishing Inc. p. 91

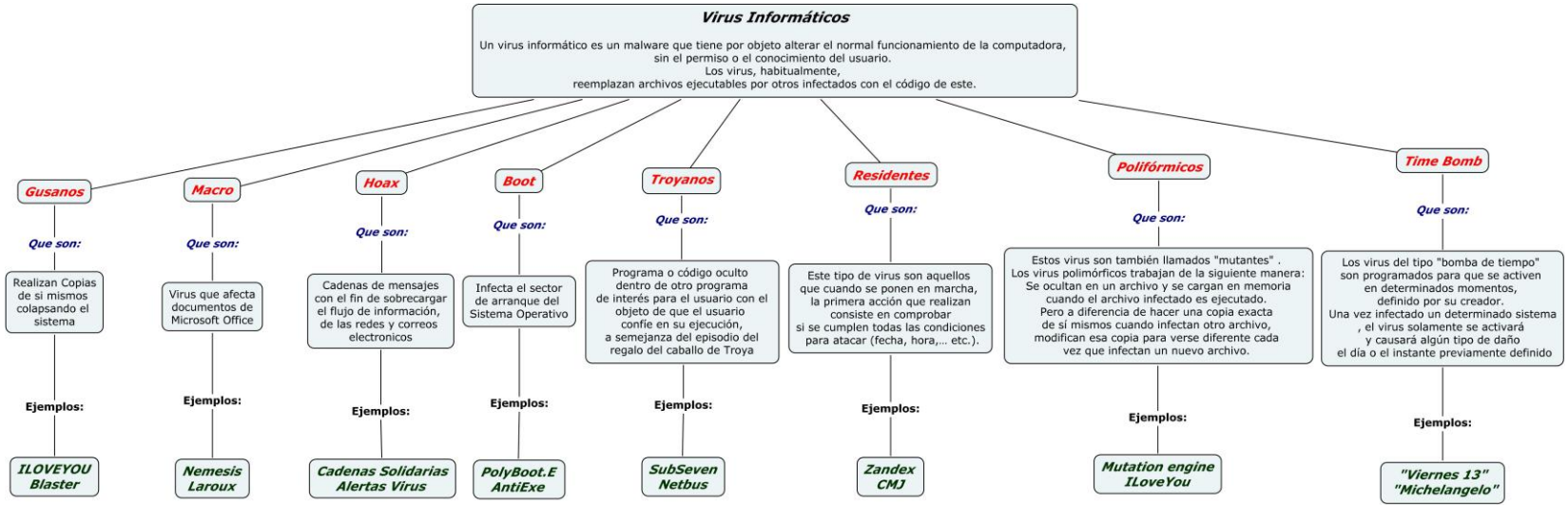
[14] Informe sobre malware en América Latina, Laboratorio ESET Latinoamérica.

ingresa a un sistema de manera completamente subrepticia activando una carga dañina, denominada payload, que despliega las instrucciones maliciosas.

La carga dañina que incorporan los troyanos de manera combinada junto a otros tipos de códigos maliciosos. Por ejemplo, cuando han ganado acceso a través del troyano, implantan en el sistema otros códigos maliciosos como rootkits que permite esconder las huellas que el atacante va dejando en el equipo, y backdoors para volver a ingresar al sistema cuantas veces considere necesario; todo, de manera remota y sin que, en la mayoría de los casos, los administradores de la red adviertan su actividad.

Si bien cualquier persona con conocimientos básicos de computación puede crear un troyano y combinar su payload con programas benignos a través de aplicaciones automatizadas y diseñados para esto, los troyanos poseen un requisito particular que debe ser cumplido para que logren el éxito, simplemente ser ejecutado por el usuario.

Es por ello que estas amenazas se diseminan por medio de diferentes tecnologías como dispositivos USB, mensajería instantánea, redes P2P, e-mail, etcétera; a través de alguna metodología de engaño (Ingeniería social), aparentando ser programas inofensivos bajo coberturas como protectores de pantalla, tarjetas virtuales, juegos en flash, diferentes tipos de archivos, simulando ser herramientas de seguridad, entre tantos otros.



Tomado de: https://delwingranados.wordpress.com/2013/02/26/mapa-conceptual-virus-informaticos/

4.6 Inteligencia De Fuentes Abiertas

Los atacantes, sobre todo los atacantes externos, aprenden constantemente técnicas de ataque que le permiten penetrar los esquemas de seguridad por más complejos que sean.

Dada la gran cantidad de información existente en la web y los diversos buscadores con algoritmos de búsqueda altamente efectivos hace que la investigación de información sea la primera acción a realizar en un ataque informático recolectando datos con técnicas como reconnaissance, discovery, footprinting o Google hacking[15].

La información recolectada por el atacante, no es más que la consecuencia de una detallada investigación sobre el objetivo, enfocada a obtener toda la información pública disponible sobre la organización desde recursos públicos. En este aspecto, un atacante gastará más del 70% de su tiempo en actividades de reconocimiento y obtención de información porque cuando más se aprende sobre el objetivo será más fácil llevar a cabo el ataque con un alto índice de éxito.

Generalmente los atacantes hacen inteligencia sobre sus objetivos durante varios meses antes de comenzar las primeras interacciones lógicas contra el objetivo a través de diferentes herramientas y técnicas como el scanning, captura de banderas y rastreo de servicios.

Algunos de los datos que se pueden encontrar y capturar son los siguientes:

---

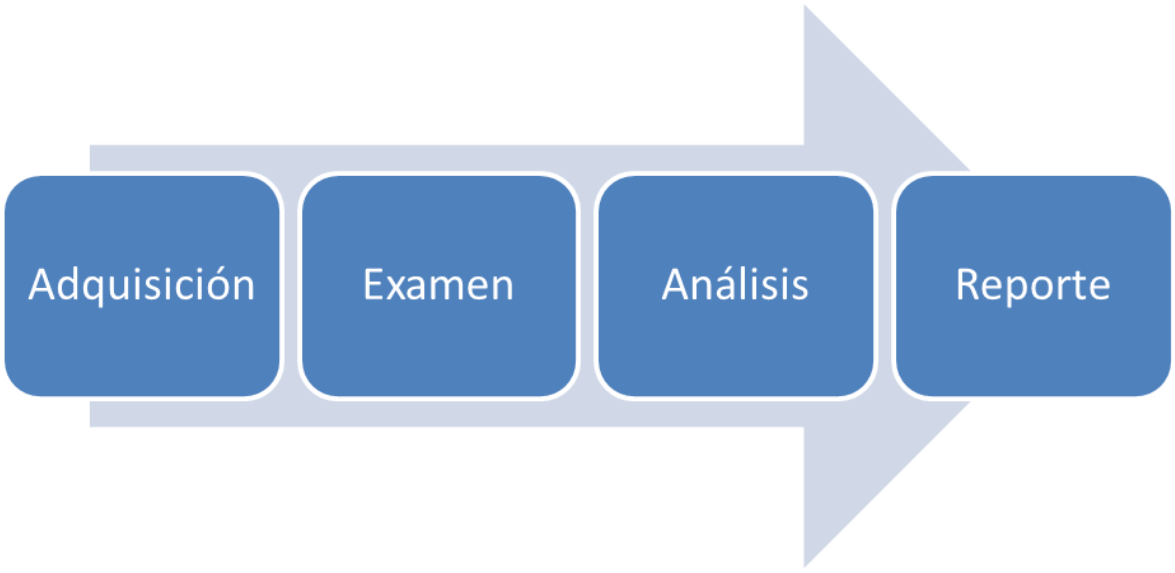
[15] GRAVES, Kimberly. Official Certified Ethical Hacker Review Guide. Indianapolis: Wiley Publishing Inc. p. 19

- Nombres de empleados y altos ejecutivos de la compañía
- Direcciones y números telefónicos
- Empresas de ISP que proveen el servicio del objetivo.
- Datos vitales de los empleados como números de teléfono, datos de familiares, curriculum vitae, antecedentes penales entre otros.
- Sistemas operativos, software utilizado, archivos, estructuras, plataformas de los servidores.
- Imágenes satelitales, accesspoint, endpoint.
- Documentos confidenciales que se encuentran sobre la web la cual se denomina fuga de información.
- Servicios de inteligencia competitiva.

Como se puede observar, no hay limite a la información que un atacante puede obtener desde fuentes públicas abiertas dónde, además, cada dato obtenido puede llevar al descubrimiento de más información.

5 METODOLOGÍA

Para realizar el análisis de malware nos basamos en NIST SP 800-86 Guide to Integrating Forensic Techniques into Incident Response[16] y NIST SP 800-83 Guide to Malware Incident Prevention and Handling[17] la cual sigue el siguiente procedimiento:



- Adquisición: En la adquisición vamos a tomar todos los elementos necesarios a evaluar, en este caso tenemos una máquina virtual con Windows 7 y descargamos un paquete de malware que será objeto de evaluación.

- Examen: Vamos a evaluar todos los aspectos a analizar, e identificar problemáticas que se pueden presentar en el análisis, en este caso estamos

[16] Disponible en: http://csrc.nist.gov/publications/nistpubs/800-86/SP800-86.pdf

[17] Disponible en http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-83.pdf

tratando con un Troyano altamente intrusivo y se deben ajustar las medidas de seguridad para no perder control del objeto a analizar.

- Análisis: En este ítem realizaremos un análisis de malware con las herramientas disponibles para ello, conforme se vaya tomando evidencias se irá construyendo el reporte que consolidará el comportamiento, los procesos y funciones del paquete malicioso.

- Reporte: En este punto realizaremos el detalle de las evidencias coleccionadas en el análisis.

Si se tratara de la naturaleza de una atención a un incidente de seguridad por malware NIST SP 800-83 recomienda el siguiente ciclo:

Fuente: http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-83.pdf

En este punto se utilizaría para actividades de “Post-Incident Activity” una labor de análisis forense dinámico donde vamos a identificar que procesos está corriendo el malware, el origen de la infección y si existe un método de propagación que pueda afectar otros activos informáticos.

Adicionalmente la guía nos entrega una tabla que puede correlacionar el tipo de malware al cual estamos sujetos:

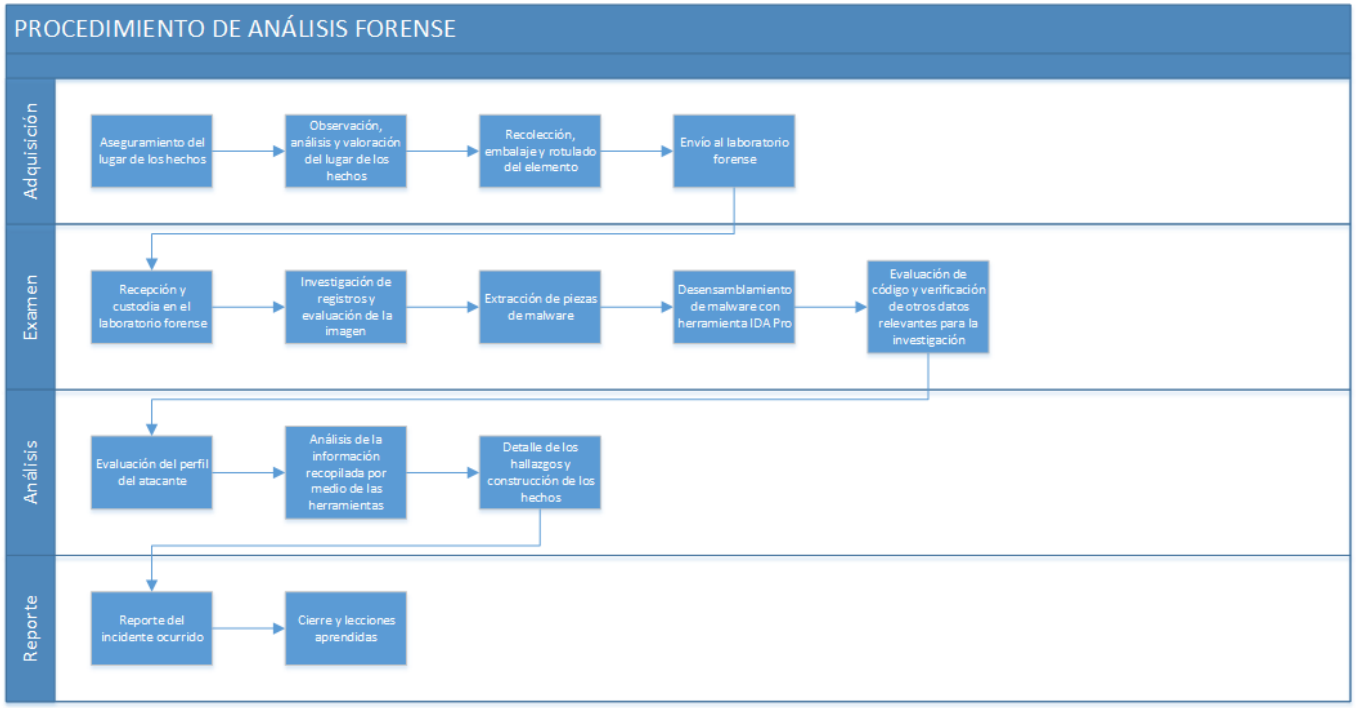
Indication	Malware Type						Attacker Tool Type				
	Multipartite Virus	Macro Virus	Network Service Worm	Mass Mailing Worm	Trojan Horse	Malicious Mobile Code	Backdoor[35]	Keystroke Logger	Rootkit	Malicious Browser Plug-Ins	E-mail Generators
Security Tools											
Antivirus software alerts	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Spyware detection and removal utility alerts					✓	✓				✓	
Network-based intrusion prevention alerts			✓	✓			✓				
Host-based intrusion detection alerts for changes to files					✓				✓		
Firewall and router log entries			✓				✓				
Observed Host Activity											
System cannot boot	✓								✓		
Error message displayed during system boot	✓								✓		
System instability and crashes occur		✓	✓		✓		✓		✓		
Programs start slowly, run slowly, or do not run at all	✓	✓	✓		✓				✓	✓	
Unknown processes are run at system startup					✓		✓	✓			✓
Unusual and unexpected ports open							✓				
Sudden increase occurs in the number of e-mails being sent and received		✓		✓					✓		
Changes are made in templates for word processing documents, spreadsheets, etc.		✓									
Web browser configuration is changed, such as different home page and new toolbars						✓				✓	
Files are deleted, corrupted, or inaccessible	✓	✓			✓				✓		
Unusual items appear on the screen, such as odd messages, graphics, and overlapping or overlaid message boxes		✓				✓			✓		✓
Unexpected dialog boxes appear, requesting permission to do something						✓				✓	
Observed Network Activity											
Significantly increased network usage			✓	✓			✓				✓
Port scans and failed connection attempts targeted at the vulnerable service (e.g., open Windows shares, HTTP)			✓				✓				
Network connections between the host and unknown remote systems			✓		✓	✓	✓	✓	✓	✓	✓

Fuente: http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-83.pdf

6 RESULTADOS

6.1 Procedimiento de Análisis Forense de Malware

El análisis forense de malware se basa en el siguiente procedimiento:



Procedimiento 1: Análisis Forense de Malware

Creado por el autor

6.2 Análisis Forense de Malware

6.2.1 Fase de Adquisición y Examen

Para esta fase vamos a utilizar técnicas para tomar una muestra de malware y realizar su respectivo análisis con el fin de compararlo contra bases de datos de firmas de malware, lo cual nos dará una idea del tipo de código malicioso al cual nos estamos enfrentando.

Para el análisis forense de malware se ha utilizado una máquina virtual con Windows 7 en un Vmware Fusion versión 8.

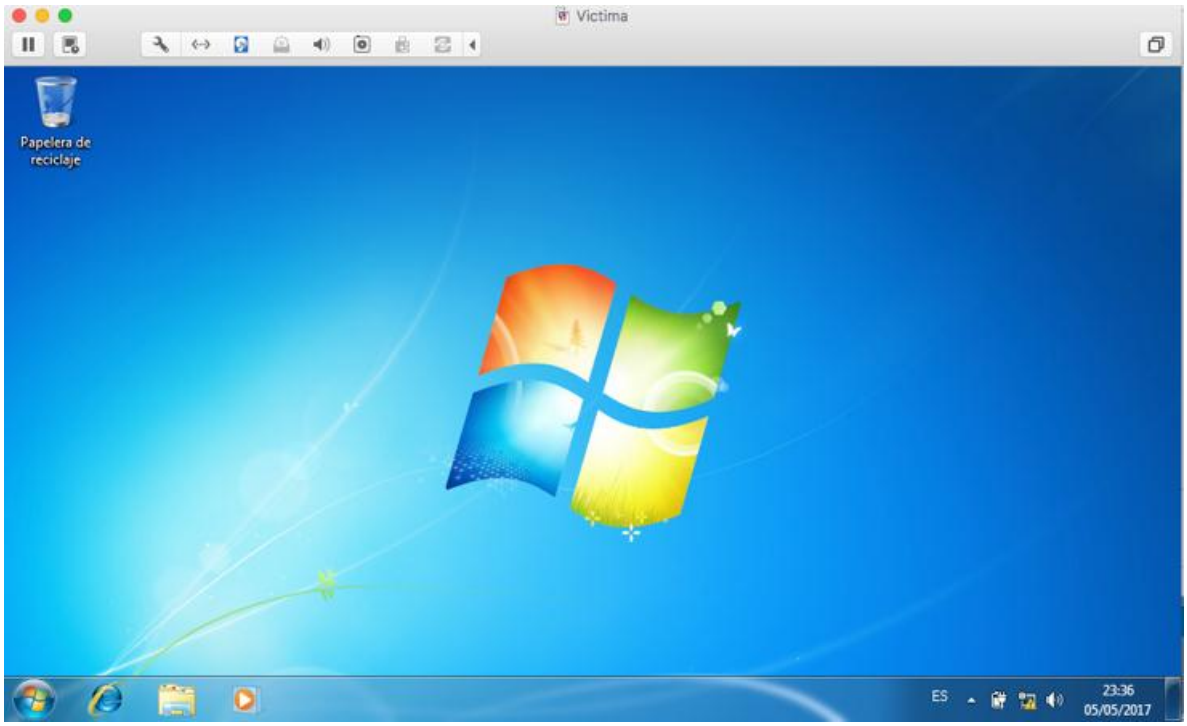


Imagen 1: Interfaz de Windows 7 donde se analizará el paquete malicioso

Imagen 2: Configuración del sistema operativo huesped

La página www.hybrid-analysis.com nos ofrece una amplia gama de tipos de malware para descarga con fines de análisis forense, en la página podemos encontrar malware de tipo Troyano, virus Macro, e inclusive virus con ransomware que podrían ser examinados en laboratorios para análisis forense de tipo dinámico.

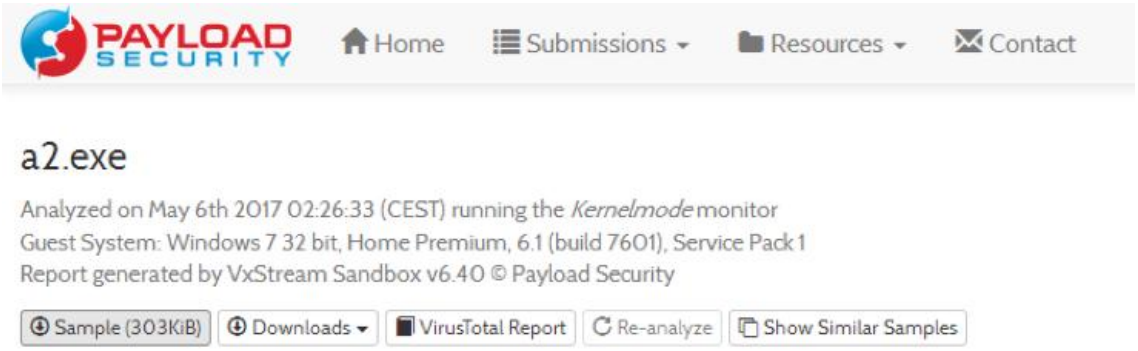


Imagen 3: Payload seleccionado para el análisis[18]

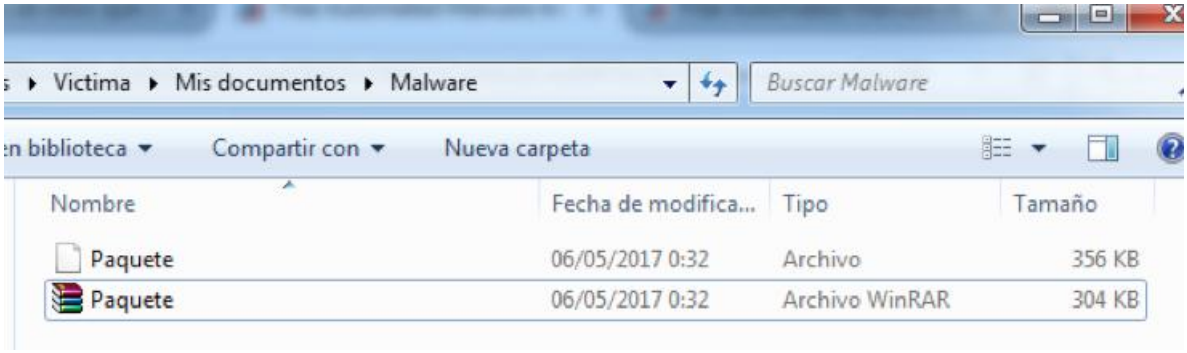


Imagen 4: Ubicación del paquete malicioso en la máquina virtual

La página web www.virustotal.com nos ofrece una evaluación de archivos de todo tipo que pueden ser evaluados contra una amplia base de datos de diferentes fabricantes en busca de registros que evidencien si el archivo seleccionado puede tener contenido malicioso.

[18] Tomado de https://www.hybrid-analysis.com/

Imagen 5: Análisis de Malware en www.virustotal.com

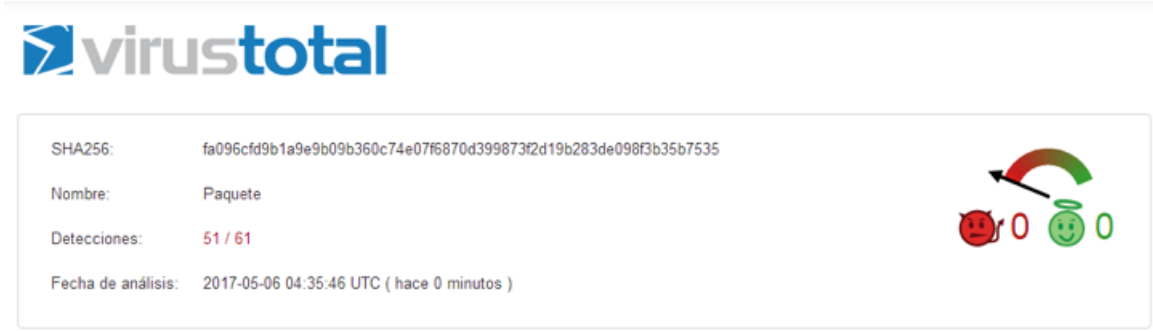


Imagen 6: Resultado del análisis en www.virustotal.com

Virus Total nos muestra que el paquete malicioso realiza match con 51 fabricantes y se evidencia que vamos a examinar un paquete altamente malicioso, se trata de un Troyano aparentemente genérico, los registros nos indican que el virus es nuevo ya que data de mayo de 2017, aunque si se trata de un troyano genérico es

posible que pueda pertenecer a una familia de troyanos y este paquete en especial fue modificado con algunas características especiales.

A continuación vamos a revisar a profundidad este troyano, identificar su funcionamiento y algunas características que nos den indicios de su comportamiento en un host infectado.

Virus Total nos muestra una larga lista de información relevante para nuestra investigación como lo es:

- Sistema operativo afectado y arquitectura objetivo
- Lenguaje de desarrollo del malware
- Información del archivo
- Metadatos del malware
- Firmas Hash del malware
- DLL’s presentes en el malware
- Conexiones remotas al ejecutar el malware

Antivirus	Resultado	Actualización
Ad-Aware	Trojan.GenericKD.4768477	20170506
AegisLab	Troj.W32.Generic!c	20170506
AhnLab-V3	Trojan/Win32.Starter.R198075	20170505
ALYac	Trojan.Dropper.Kovter	20170506
Antiy-AVL	Trojan/Win32.TSGeneric	20170506
Arcabit	Trojan.Generic.D48C2DD	20170506
Avast	Win32:Trojan-gen	20170506
AVG	GenericX.1694	20170506
Avira (no cloud)	TR/Crypt.Xpack.idawy	20170505
AVware	Trojan.Win32.Generic!BT	20170506
Baidu	Win32.Trojan.WisdomEyes.16070401.9500.9764	20170503
BitDefender	Trojan.GenericKD.4768477	20170506
Bkav	W32.MoratisLTG.Trojan	20170505
Comodo	UnclassifiedMalware	20170506
CrowdStrike Falcon (ML)	malicious_confidence_97% (W)	20170130
Cyren	W32/Kovter.T.gen!Eldorado	20170506
DrWeb	Trojan.Kovter.297	20170506
Emsisoft	Trojan.GenericKD.4768477 (B)	20170506
Endgame	malicious (high confidence)	20170503
ESET-NOD32	a variant of Win32/Kryptik.FRDR	20170505
F-Prot	W32/Kovter.T.gen!Eldorado	20170506
F-Secure	Trojan.GenericKD.4768477	20170506
Fortinet	W32/Kryptik.FQTO!tr	20170506
GData	Trojan.GenericKD.4768477	20170506
Ikarus	Trojan.Win32.Krypt	20170505
Invincea	generic.a	20170413
K7AntiVirus	Riskware (0040eff71)	20170505
K7GW	Riskware (0040eff71)	20170505
Kaspersky	HEUR:Trojan.Win32.Generic	20170506
Malwarebytes	Trojan.Kovter	20170506
McAfee	GenericRXBE-ZW!9F1DBE4D91E8	20170506
McAfee-GW-Edition	GenericRXBE-ZW!9F1DBE4D91E8	20170505

Microsoft	Trojan:Win32/Kovter!rfn	20170506
eScan	Trojan.GenericKD.4768477	20170505
NANO-Antivirus	Trojan.Win32.Kryptik.ensqdl	20170505
Palo Alto Networks (Known Signatures)	generic.ml	20170506
Panda	Trj/Genetic.gen	20170505
Qihoo-360	Win32/Trojan.a44	20170506
Rising	Malware.Generic.1ltfe (cloud:FpCQi1f3QyS)	20170506
SentinelOne (Static ML)	static engine - malicious	20170330
Sophos	Mal/Kovter-Z	20170506
Symantec	Trojan.Gen	20170505
Tencent	Win32.Trojan.Kryptik.Hupp	20170506
TrendMicro	TROJ_GEN.R047C0RD717	20170506
TrendMicro-HouseCall	TROJ_GEN.R047C0RD717	20170506
VBA32	Trojan.Poweliks	20170505
VIPRE	Trojan.Win32.Generic!BT	20170506
ViRobot	Trojan.Win32.Z.Kovter.363983[h]	20170506
Webroot	W32.Trojan.Gen	20170506

Yandex	Trojan.GenKryptik!	20170504
ZoneAlarm by Check Point	HEUR:Trojan.Win32.Generic	20170506
Alibaba		20170505
CAT-QuickHeal		20170505
ClamAV		20170505
CMC		20170505
Jiangmin		20170506
Kingsoft		20170506
nProtect		20170506
SUPERAntiSpyware		20170506
Symantec Mobile Insight		20170504
TheHacker		20170505
Trustlook		20170506
WhiteArmor		20170502
Zillya		20170505
Zoner		20170506

Imagen 7: Tipo de infecciones detectadas por fabricante

Imagen 8: Características del archivo malicioso

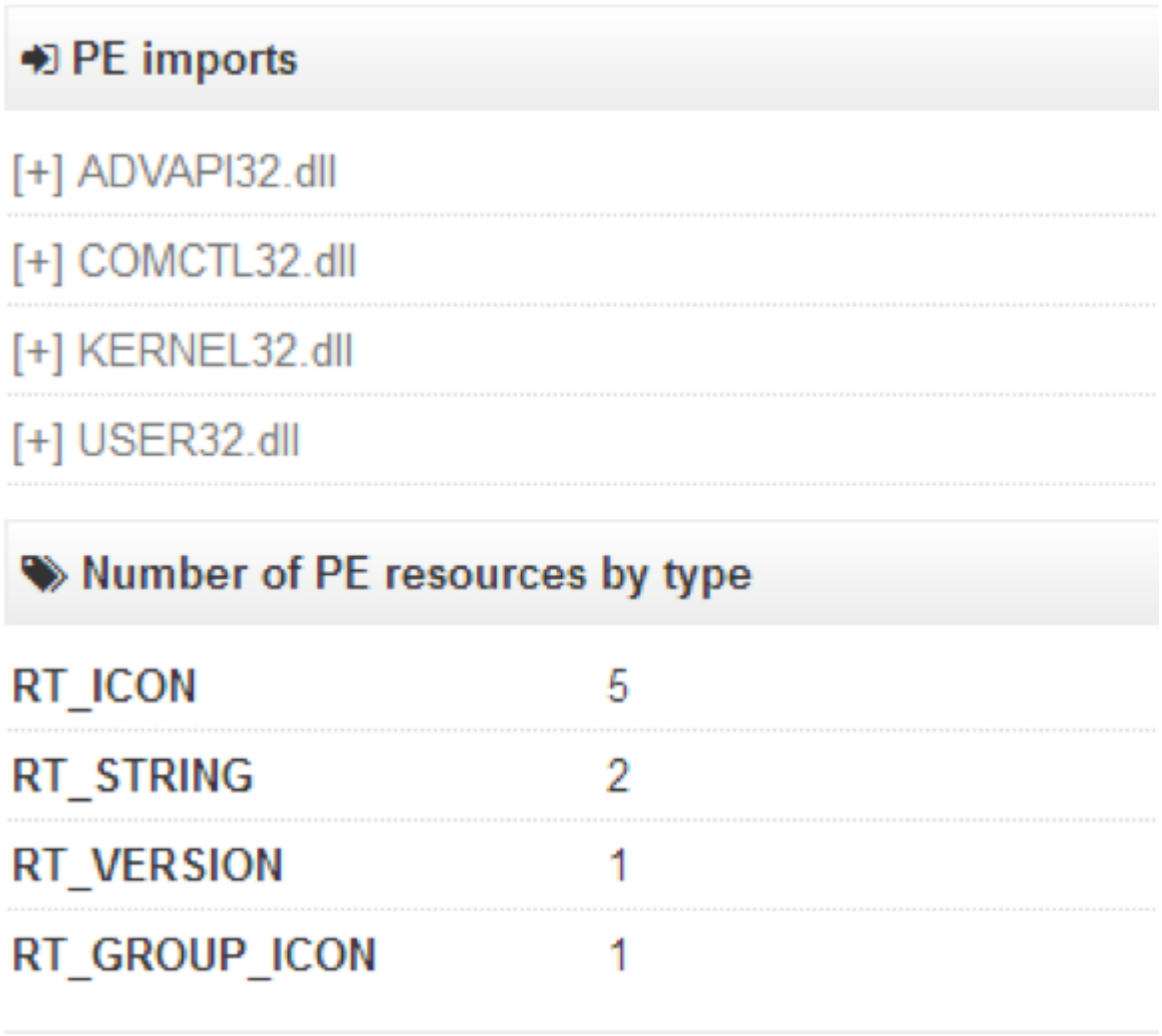


Imagen 9: DLL's usadas por el paquete y ejecutables portables por tipo

ExifTool file metadata	
SpecialBuild	5.6.6, FINAL_2013_1213_022844
SubsystemVersion	4.0
LinkerVersion	2.23
ImageVersion	1.0
FileSubtype	0
FileVersionNumber	5.6.6.3516
LanguageCode	English (U.S.)
FileFlagsMask	0x003f
FileDescription	Winamp Age
CharacterSet	Windows, Latin1
InitializedDataSize	299008
PrivateBuild	Release|Win32
EntryPoint	0x2a90
OriginalFileName	winampa.exe
MIMEType	application/octet-stream
LegalCopyright	Copyright 1997 Nullsoft, Inc.
FileVersion	5.6.6.3516
TimeStamp	2016:02:25 17:59:15+01:00
FileType	Win32 EXE
PEType	PE32
InternalName	winampa.exe

ExifTool file metadata	
SpecialBuild	5.6.6, FINAL_2013_1213_022844
SubsystemVersion	4.0
LinkerVersion	2.23
ImageVersion	1.0
FileSubtype	0
FileVersionNumber	5.6.6.3516
LanguageCode	English (U.S.)
FileFlagsMask	0x003f
FileDescription	Winamp Age
CharacterSet	Windows, Latin1
InitializedDataSize	299008
PrivateBuild	Release|Win32
EntryPoint	0x2a90
OriginalFileName	winampa.exe
MIMEType	application/octet-stream
LegalCopyright	Copyright 1997 Nullsoft, Inc.
FileVersion	5.6.6.3516
TimeStamp	2016:02:25 17:59:15+01:00
FileType	Win32 EXE
PEType	PE32
InternalName	winampa.exe

Imagen 10: Metadatos usados por el paquete malicioso

File identification	
MD5	9f1dbe4d91e8bfc321ab3f1b832a071d
SHA1	6db584b371352fc8af7e5499c07e753793a46caf
SHA256	fa096cfd9b1a9e9b09b360c74e07f6870d399873f2d19b283de098f3b35b7535
ssdeep	6144:G6Lv0bMDkh8PECXOk3rsywcXrCsk75Xw3h8Mqqbveuuh:G6Lv0bVh8MCNrstf1XwR8Xyvly
authentihash	5ce3165bbbb9f4e2eeb8b8e3cd62bca3c5058eb982d2185c789e3decd293ecab
imphash	70b6da0c0906fbe0b93de320ed32d0e9
Tamaño del fichero	355.5 KB (363983 bytes)
Tipo	Win32 EXE
Magic literal	PE32 executable for MS Windows (GUI) Intel 80386 32-bit
TrID	Win32 Executable MS Visual C++ (generic) (42.1%) Win64 Executable (generic) (37.3%) Win32 Dynamic Link Library (generic) (8.8%) Win32 Executable (generic) (6.0%) Generic Win/DOS Executable (2.7%)
Tags	peexe overlay

Imagen 11: Identificación del archivo con firmas de HASH

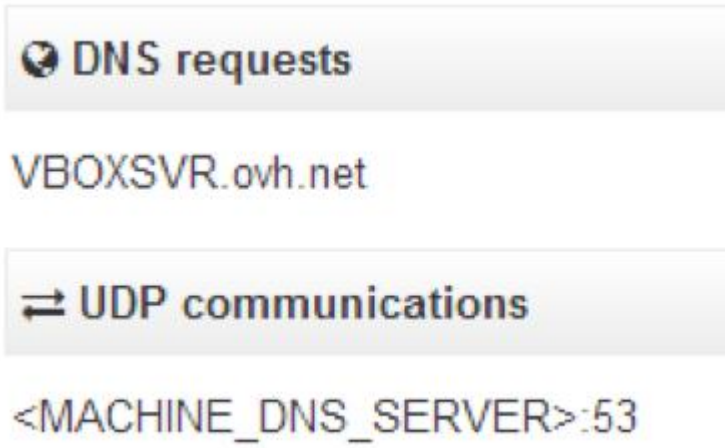


Imagen 12: Comunicaciones maliciosas al exterior

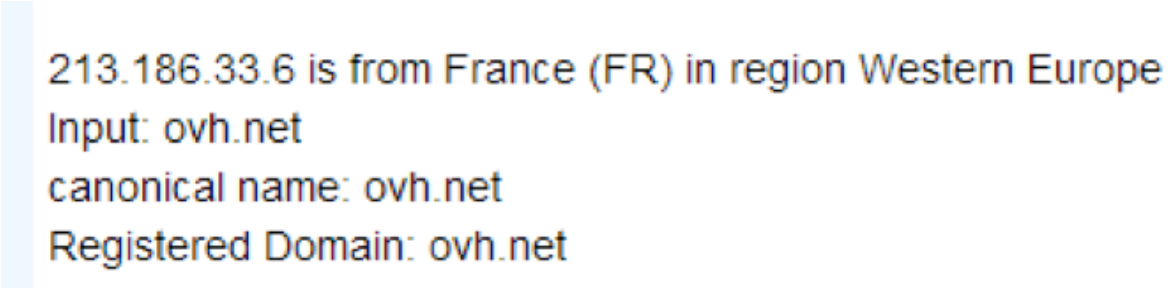


Imagen 13: Información del dominio ovh.net

En este caso las conexiones del malware se dirigen hacia la IP pública 213.186.33.6 ubicada en Francia, algo bastante curioso ya que actualmente muchos tipos de malware se dirigen hacia países asiáticos como China o Corea del Norte, en el caso de Ransomware se dirigen hacia dominios .onion de la red TOR.

6.2.2 Fase de Examen

A continuación procederemos a realizar el análisis del paquete malicioso evaluando sus DLL's. PE's, desensamblando el paquete y viendo cada uno de sus registros, para ello ubicamos una serie de herramientas en la máquina de análisis y evaluaremos en detalle cada una de las evidencias presentadas.

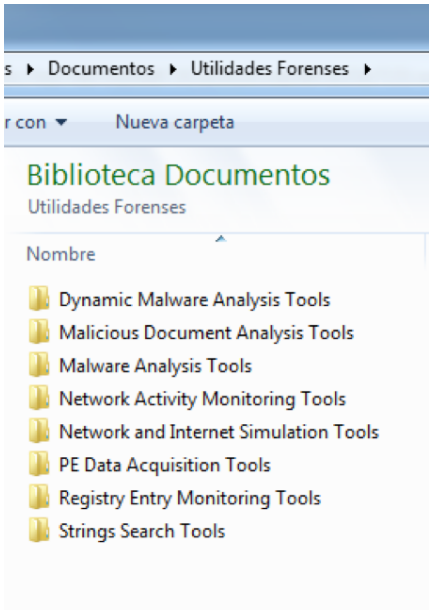


Imagen 14: Herramientas para el análisis de malware

La herramienta PEiD[19] nos ofrece información sobre empaquetadores comunes, compiladores y firmas en archivos PE, actualmente el software ha sido descontinuado y es por ello que posiblemente la herramienta no realizó match en su base de datos.

[19] Tomado de https://www.aldeid.com/wiki/PEiD

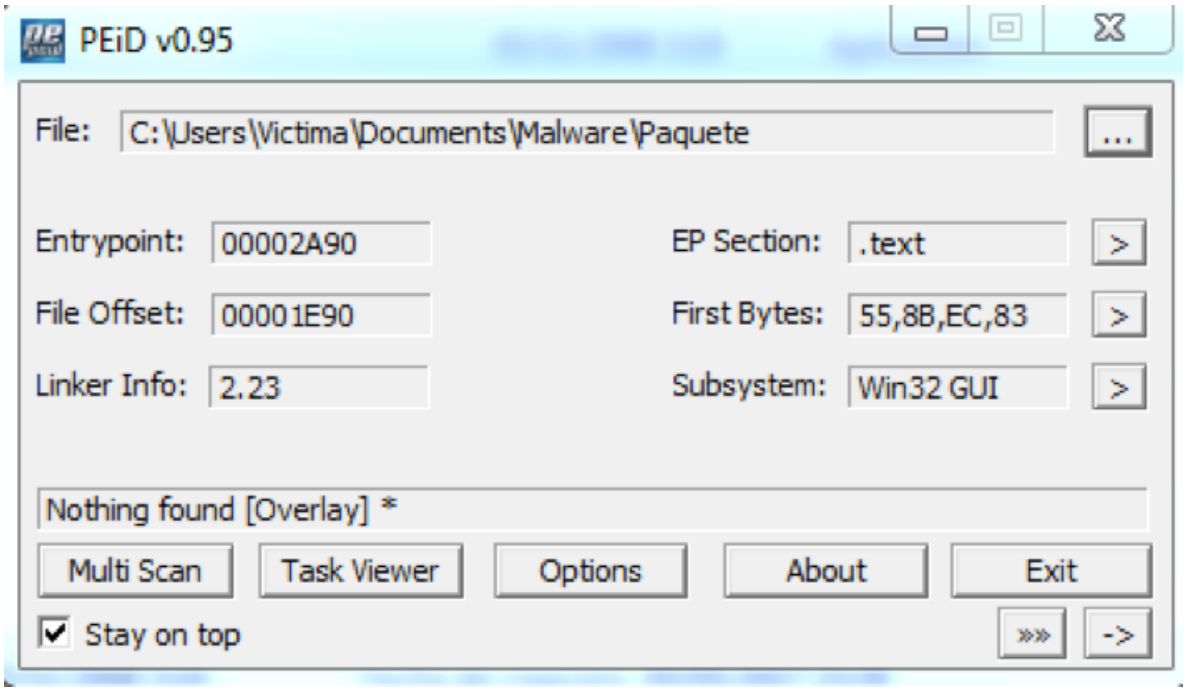


Imagen 15: Verificación con herramienta PEiD

A continuación vamos a utilizar la herramienta PEView[20] en la cual vamos a revisar las cabeceras de los PE relevantes, en ellos podemos encontrar el tamaño de los datos en crudo, la fecha que registra el malware y es allí que podemos afirmar que es un troyano que pertenece a una familia y tiene una variante, adicionalmente observamos que el troyano se presenta como una GUI en Windows y aparentemente lo presenta como una versión de Winamp, un viejo reproductor de música en sistemas operativos Windows.

[20] Tomado de https://www.aldeid.com/wiki/PEView

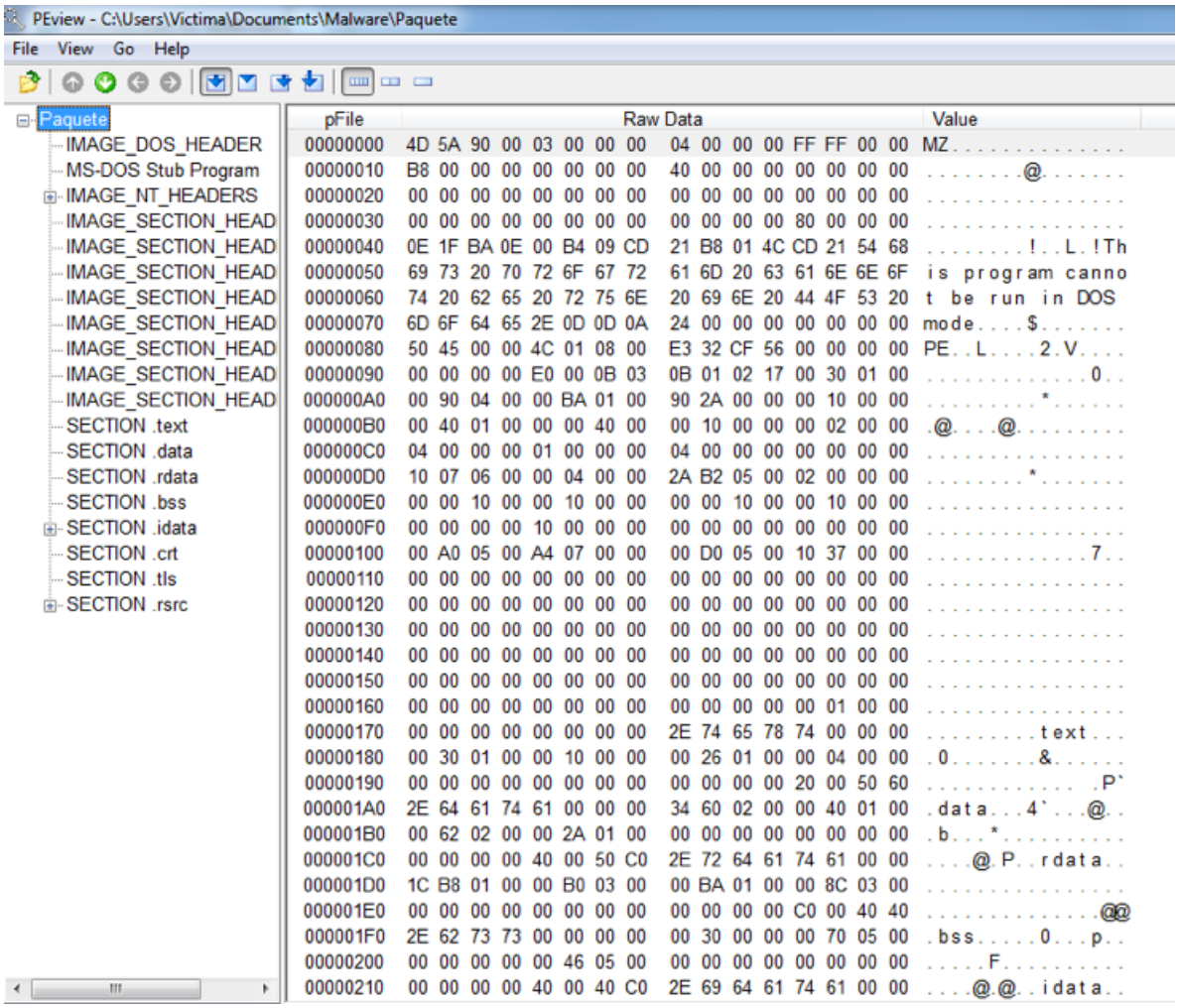


Imagen 16: Verificación de cabeceras de los ejecutables portables con herramienta PEview

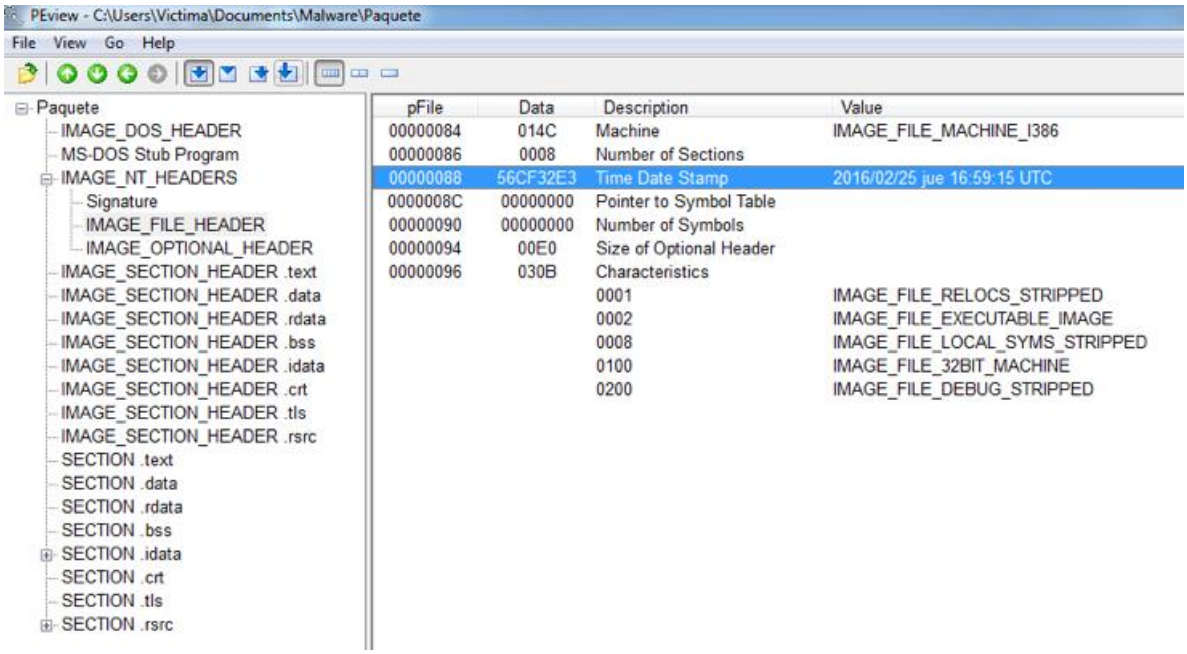


Imagen 17: TimeStamp del paquete malicioso

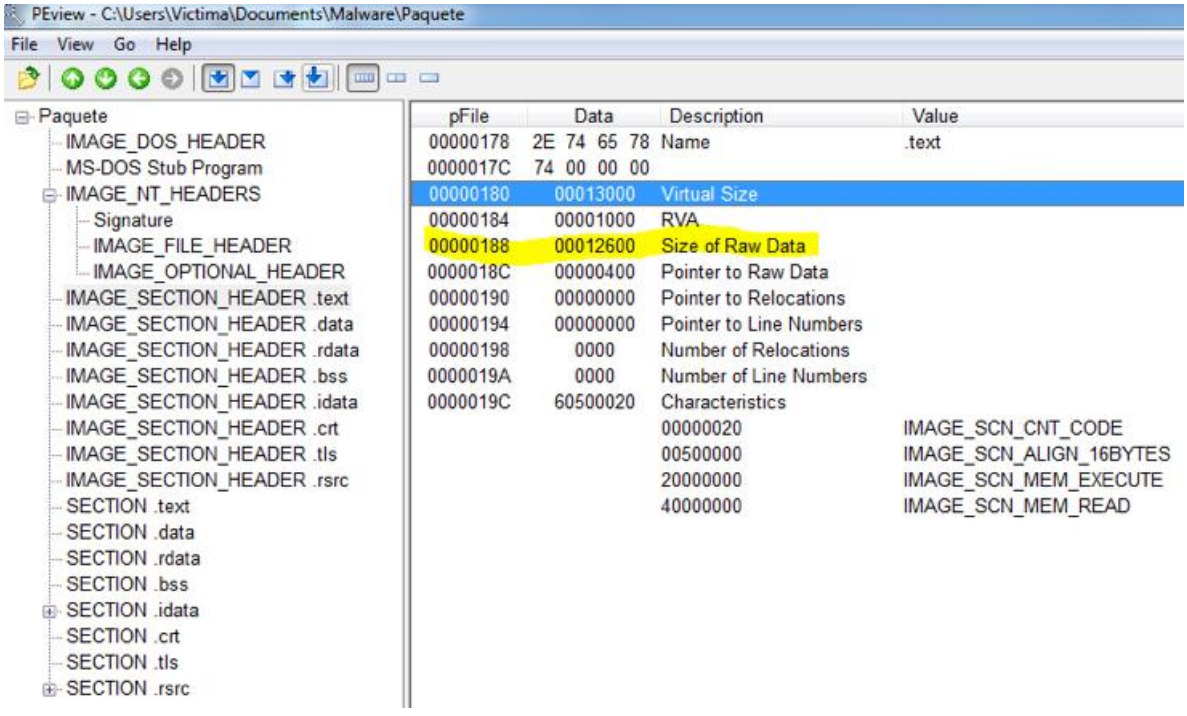


Imagen 18: Tamaño del dato en crudo

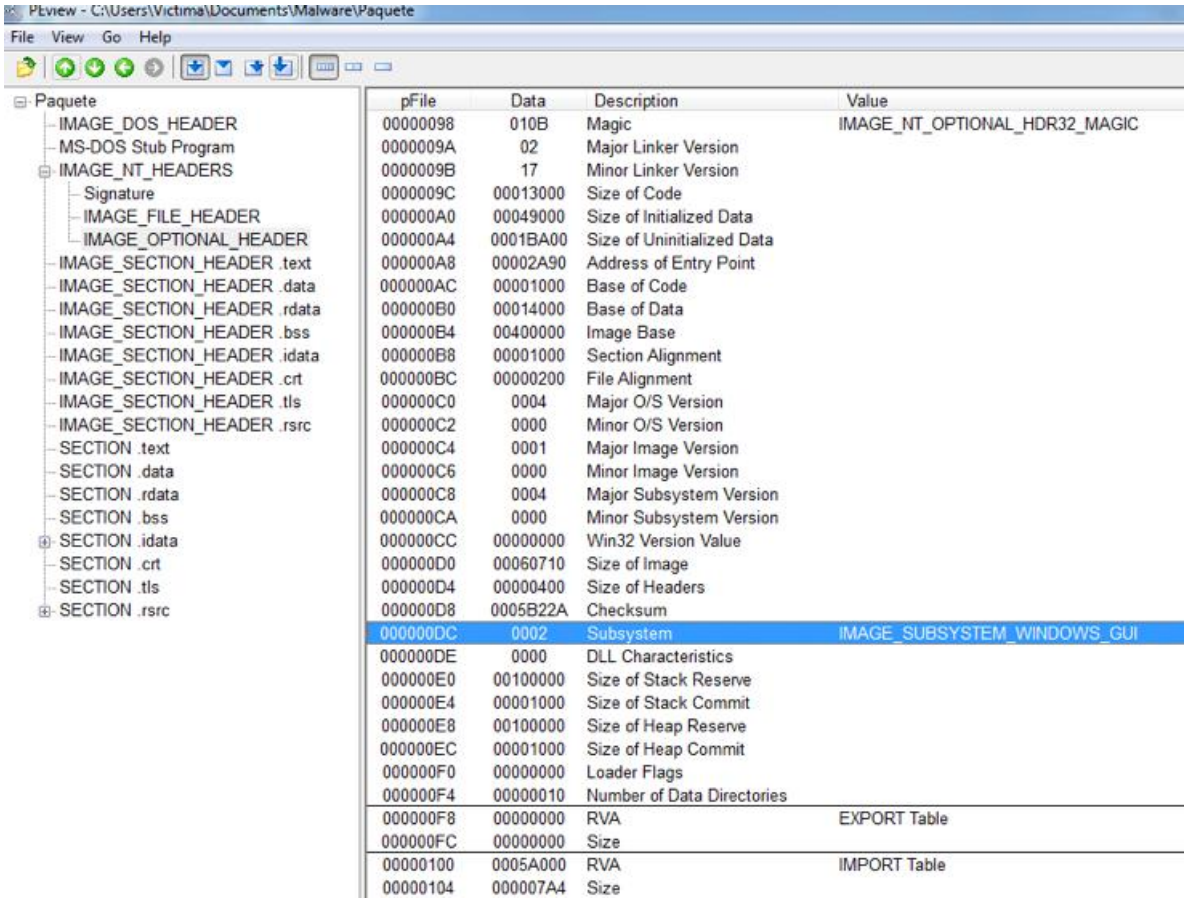


Imagen 19: El paquete malicioso presenta interfaz gráfica

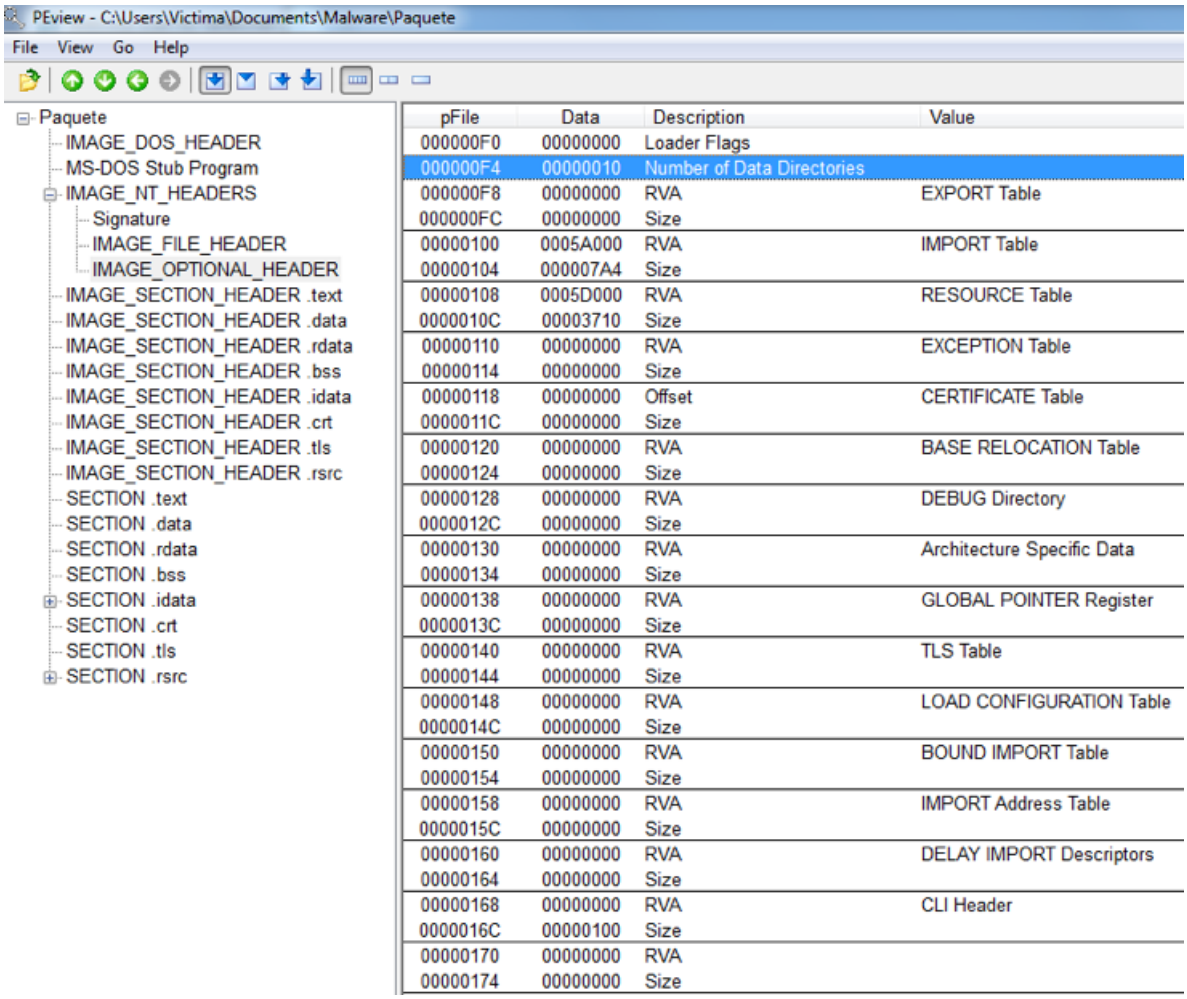


Imagen 20: Tamaño de las cabeceras

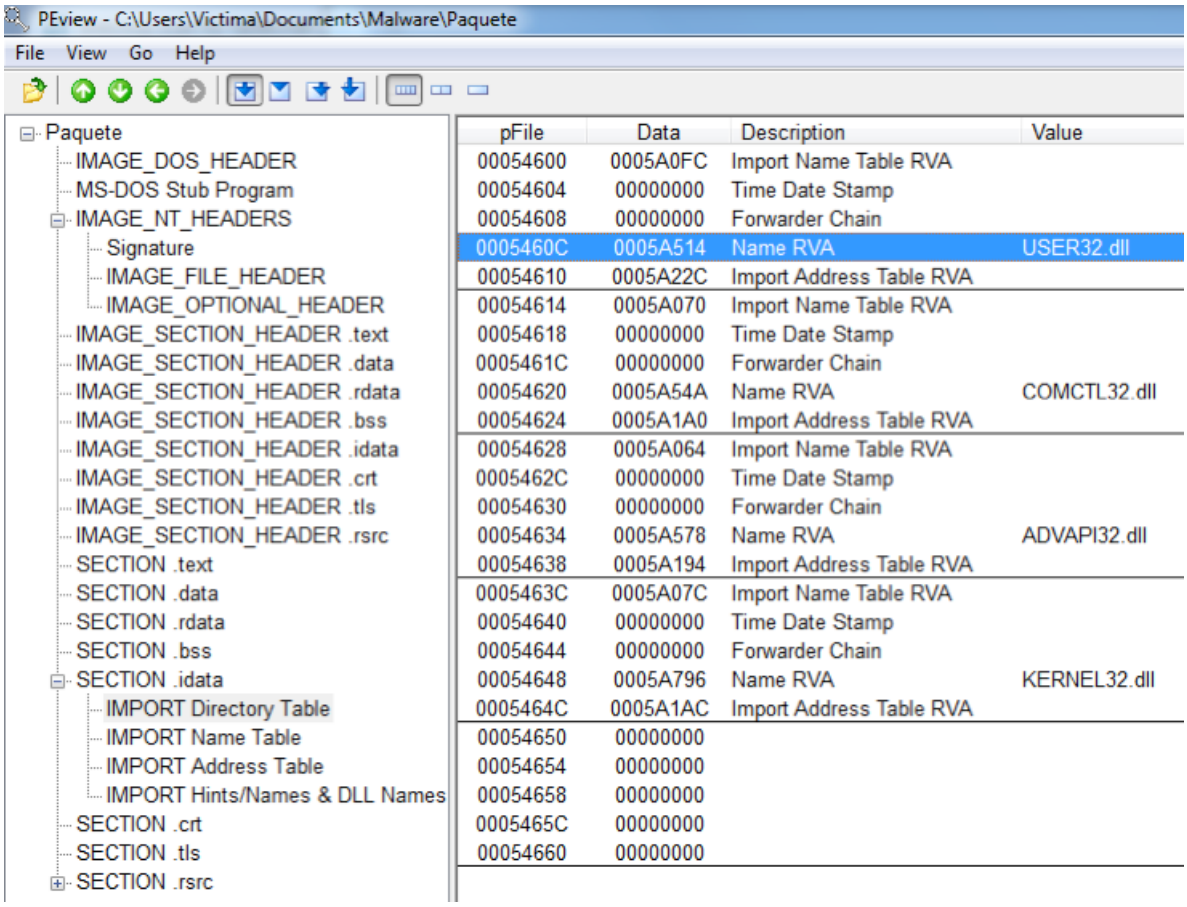


Imagen 21: Llamados de DLL's

A continuación vamos a utilizar la herramienta Strings[21] la cual nos va a permitir extraer las cadenas ASCII o Unicode del paquete malicioso, y nos da muchisima evidencia sobre funciones Get que implica que es un troyano que toma información relevante sobre sus objetivos, tambien muestra implicaciones sobre

[21] Tomado de: https://technet.microsoft.com/en-us/sysinternals/bb897439.aspx?ranMID=24542&ranEAID=TnL5HPStwNw&ranSiteID=TnL5HPStwNw-vuLCNLQfreqSHdyINFFeNw&tduid=(260922475e641520617f158ce647f028)(256380)(2459594)(TnL5HPStwNw-vuLCNLQfreqSHdyINFFeNw)()

cambios en los registros del sistema operativo y funciones Set para escribir sobre archivos, se realizó una escritura sobre un archivo de texto para mejor análisis.

```
C:\Users\Victima\Documents\Utilidades Forenses\Strings Search Tools\Strings>stri
ngs "C:\Users\Victima\Documents\Malware\paquete"
```

Imagen 22: Sentencia para la herramienta Strings

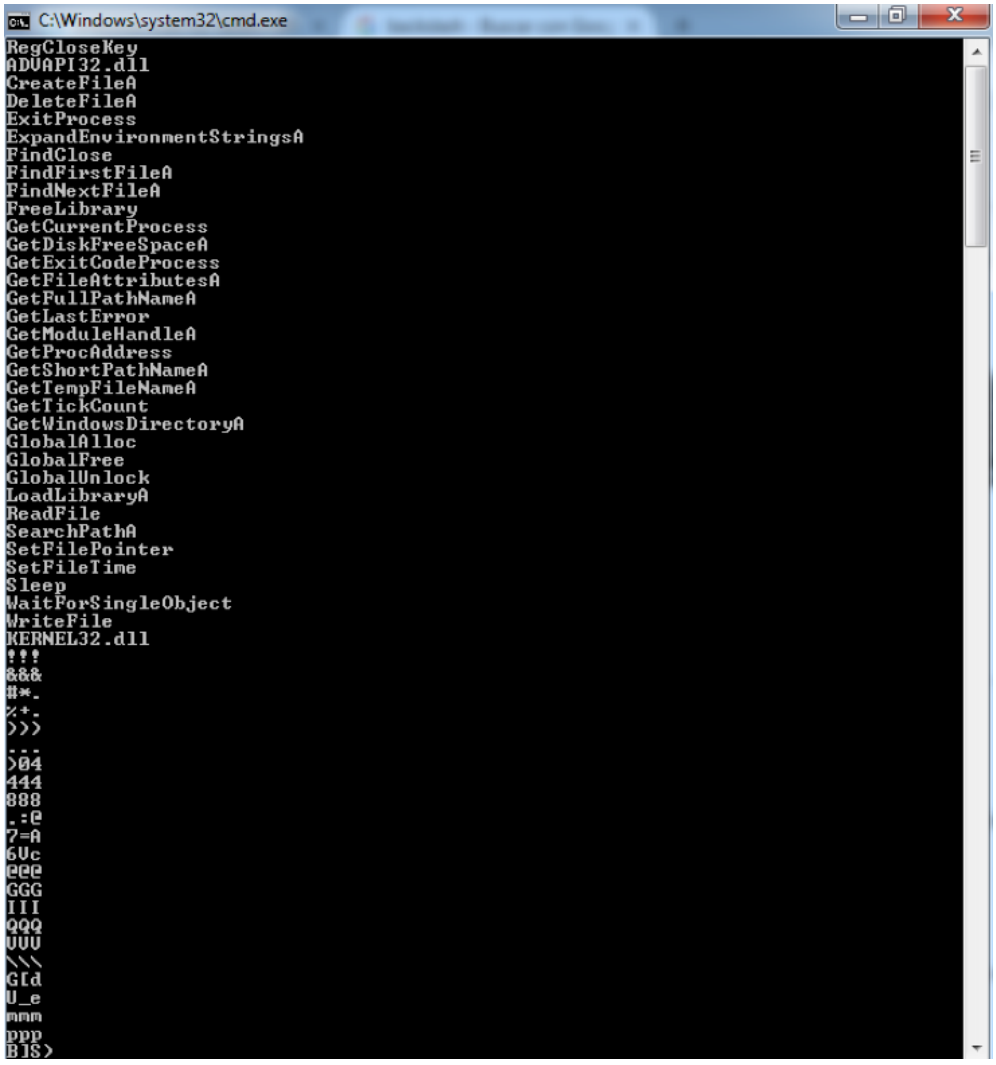
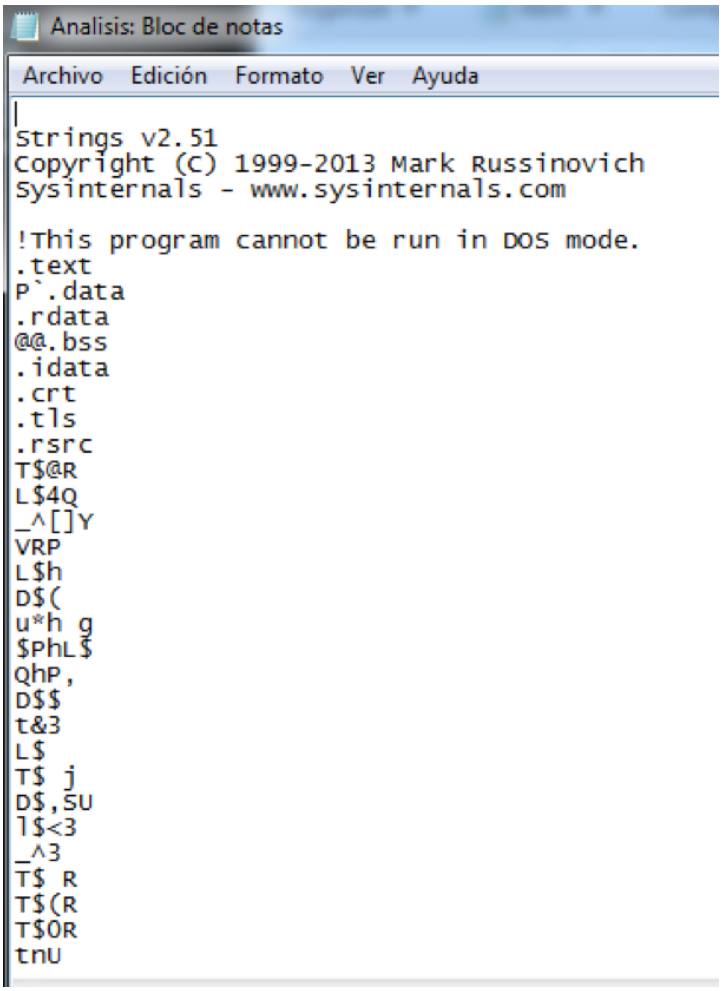


Imagen 23: Strings embebidos en el paquete malicioso

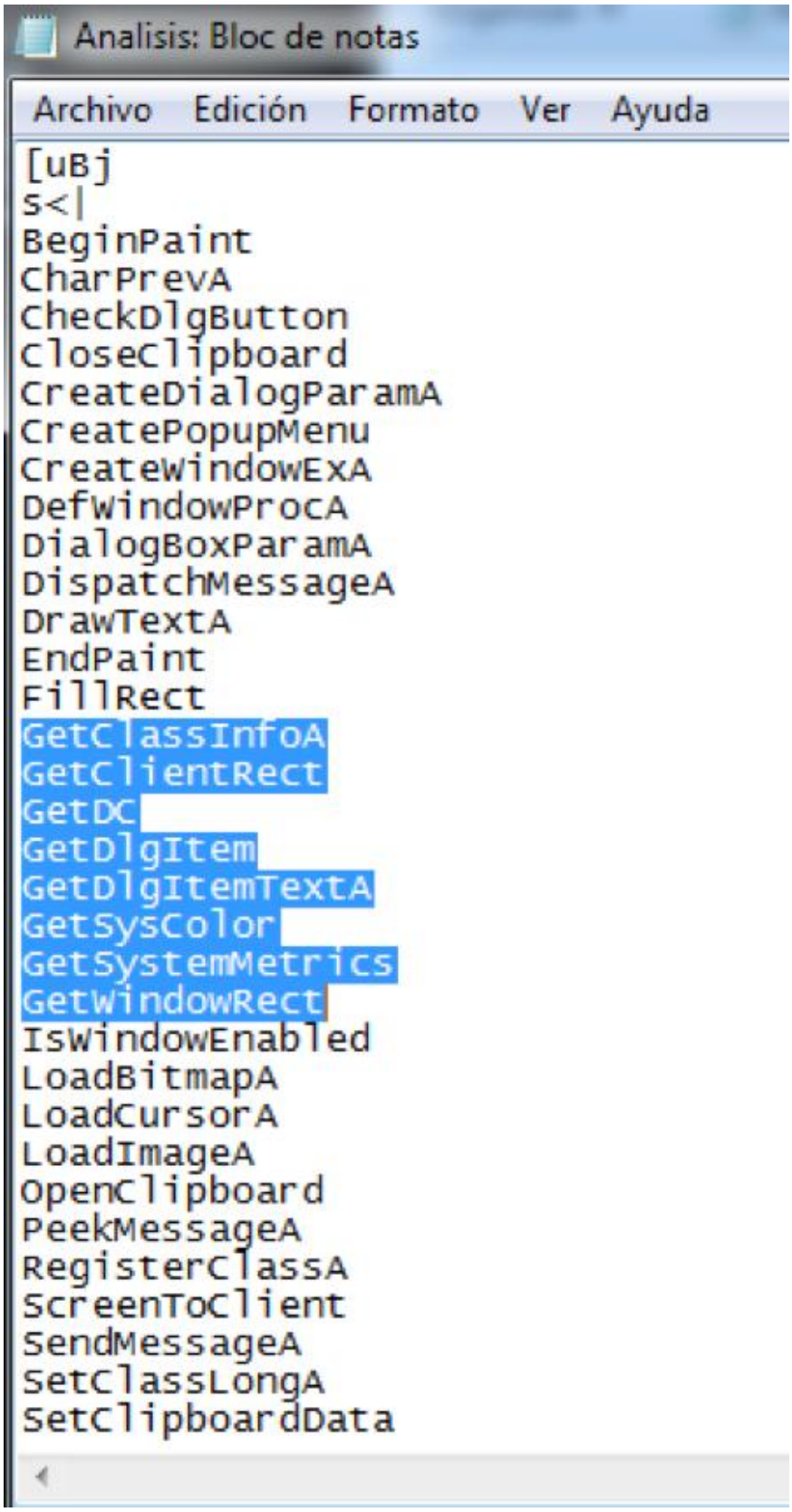
```
C:\Users\Victima\Documents\Utilidades Forenses\Strings Search Tools\Strings>stri
ngs "C:\Users\Victima\Documents\Malware\paquete">C:\Users\Victima\Documents\Malw
are\Analisis.txt

C:\Users\Victima\Documents\Utilidades Forenses\Strings Search Tools\Strings>
```

Imagen 24: Sentencia para exportar los Strings a .txt para mejor análisis




Analisis: Bloc de notas
Archivo Edición Formato Ver Ayuda
VariableA
InitializeCriticalSection
DeleteCriticalSe
Py_BuildValue
PyCObject_FromVoidPtrAndDesc
PyDict_SetItemString
PyFloat_FromDouble
PyArg_
ect
GetModuleFileNameW
GetFileAttributesW
LocalFree
LocalAlloc
ExitProcess
QueryPerformanceCount
wvsprintfW
USER32.dll
GDI32.dll
RegQueryValueExW
RegOpenKeyExW
RegCloseKey
ADV
DumpMsg
PCM
appname
..\startup.ini
TF8
BDROMIP_BACK_AV_OP_ERR_
PyExc_
intf
_dumpfil
GetCImmapi->VideoDe
etPi
m_pImmapi->Videset


Analisis: Bloc de notas
Archivo Edición Formato Ver Ayuda
[uBj
s<|
BeginPaint
CharPrevA
CheckDlgButton
CloseClipboard
CreateDialogParamA
CreatePopupMenu
CreateWindowExA
DefWindowProcA
DialogBoxParamA
DispatchMessageA
DrawTextA
EndPaint
FillRect
GetClassInfoA
GetClientRect
GetDC
GetDlgItem
GetDlgItemTextA
GetSysColor
GetSystemMetrics
GetWindowRect
IsWindowEnabled
LoadBitmapA
LoadCursorA
LoadImageA
OpenClipboard
PeekMessageA
RegisterClassA
ScreenToClient
SendMessageA
SetClassLongA
SetClipboardData


Analisis: Bloc de notas
Archivo Edición Formato Ver Ayuda
LoadBitmapA
LoadCursorA
LoadImageA
OpenClipboard
PeekMessageA
RegisterClassA
ScreenToClient
SendMessageA
SetClassLongA
SetClipboardData
SetCursor
SetDlgItemTextA
SetWindowPos
SetWindowTextA
ShowWindow
USER32.dll
ImageList_GetIconSize
ImageList_Merge
COMCTL32.dll
LsaLookupNames
RegCloseKey
ADVAPI32.dll
CreateFileA
DeleteFileA
ExitProcess
ExpandEnvironmentStringsA
FindClose
FindFirstFileA
FindNextFileA
FreeLibrary
GetCurrentProcess
GetDiskFreeSpaceA
GetExitCodeProcess
GetFileAttributesA

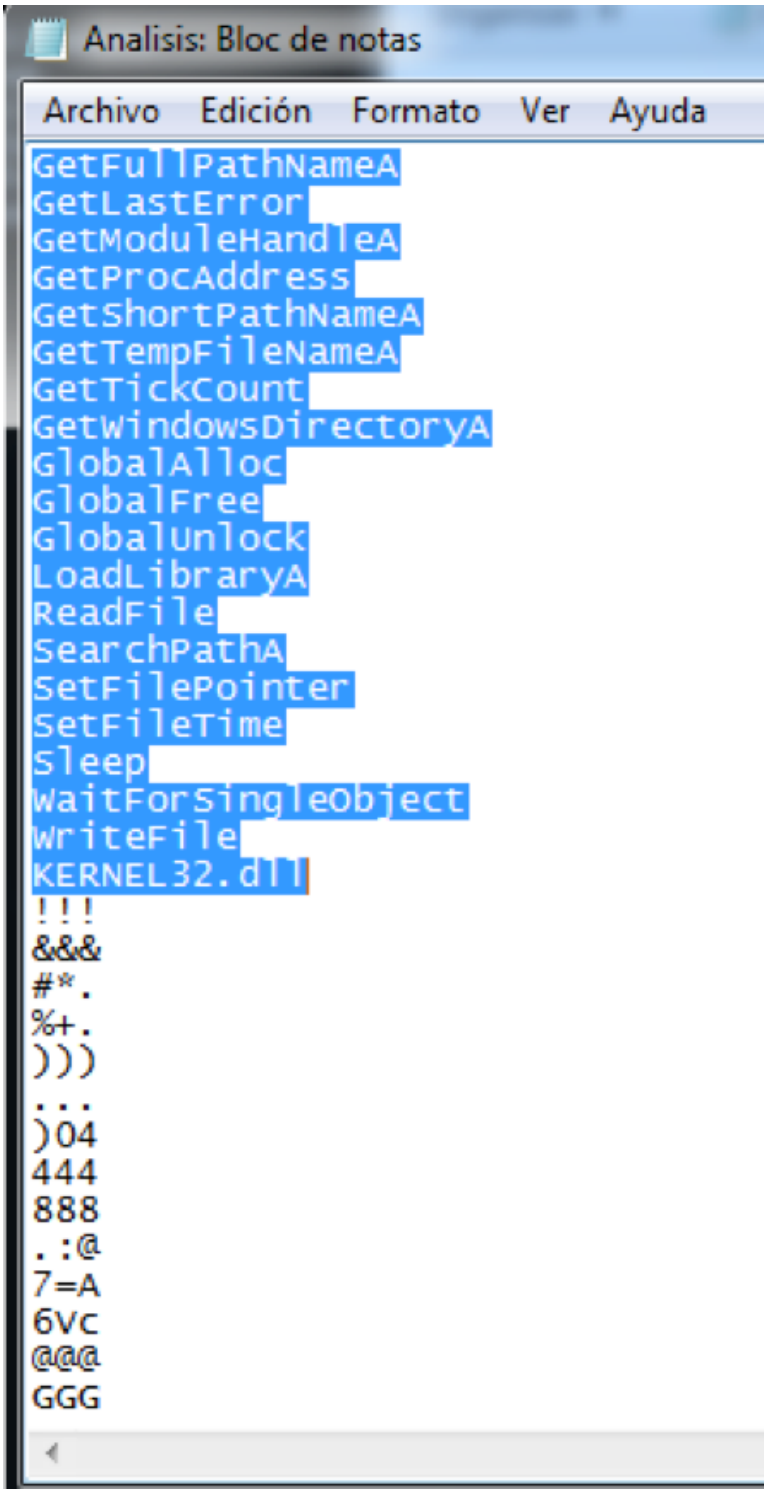


Imagen 25: Palabras con contenido malicioso de procesos y funciones Get del malware

A continuación vamos a desensamblar completamente el paquete malicioso con la herramienta IDA[22], con el vamos a poder ver completamente las funciones del paquete malicioso y las subrutinas de las funciones.

La herramienta nos corrobora lo invasivo que puede llegar a ser este troyano dado que obtiene información de la víctima incluso de la memoria y la sobreescribe a archivos de texto para luego ser enviados, tambien tiene funciones post quizá para ejecutar instrucciones remotamente.

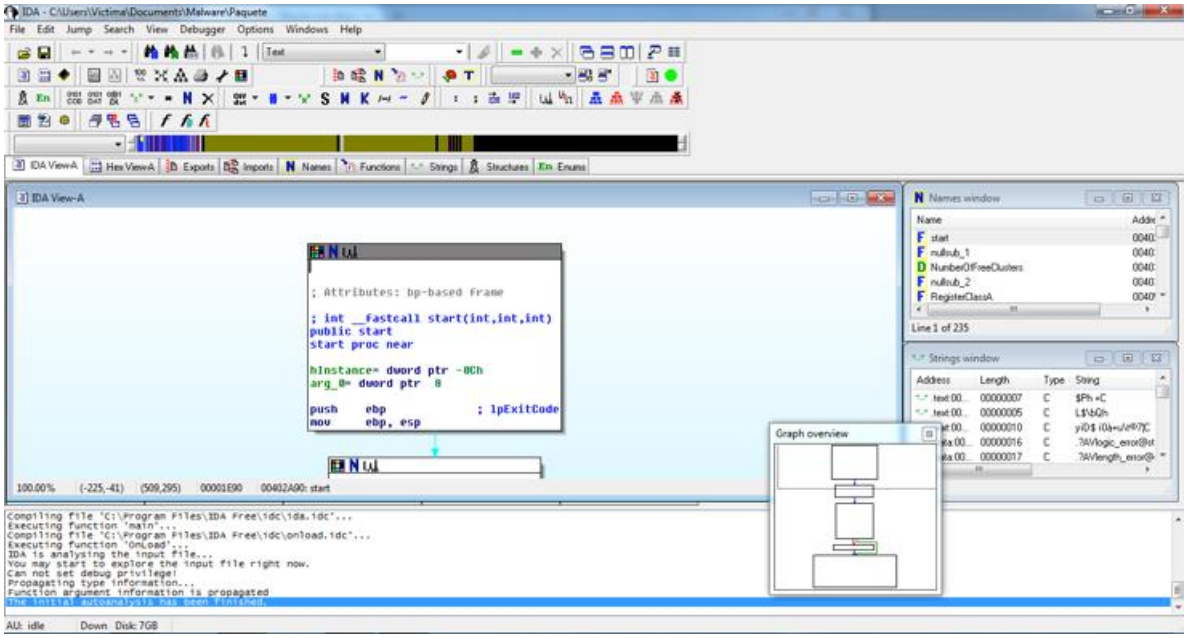


Imagen 26: Paquete malicioso desensamblado con IDA

[22] Tomado De https://www.hex-rays.com/products/ida/

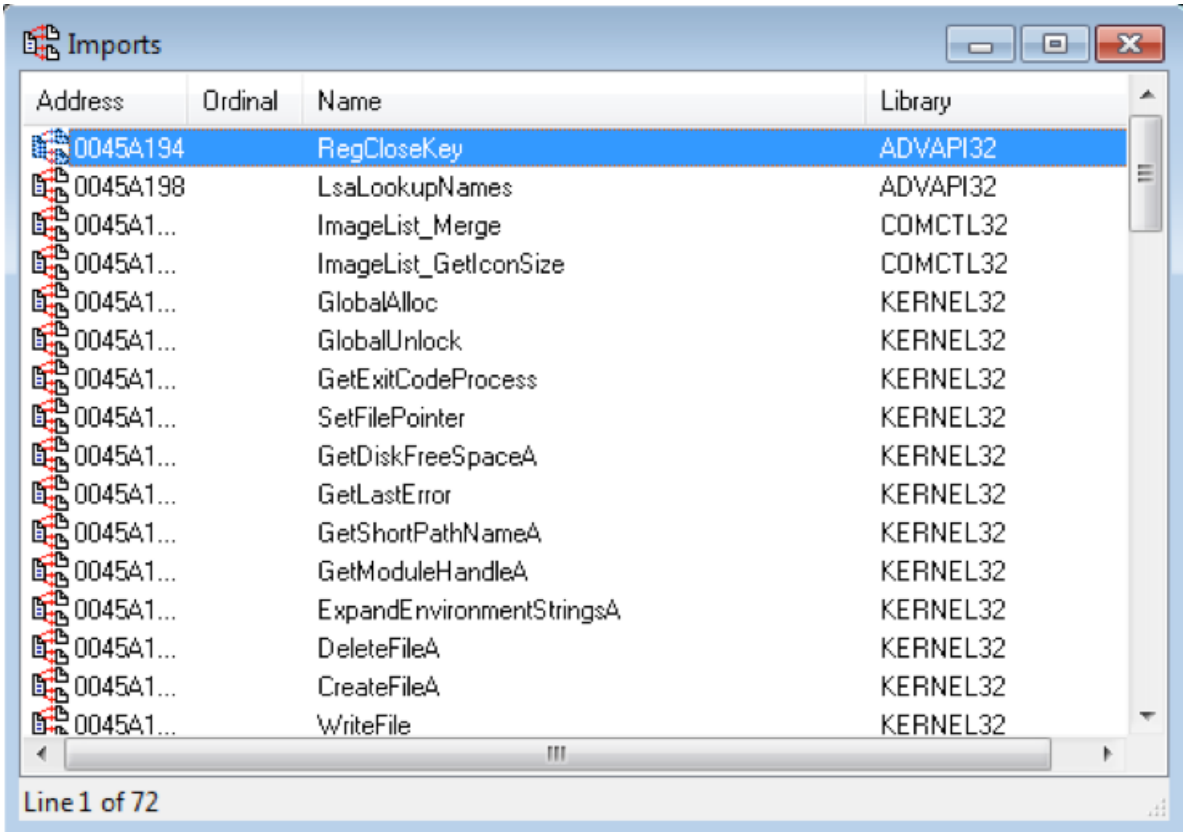


Imagen 27: Evaluación de los imports del paquete malicioso

Imagen 28: Examen de la subrutina de función Get del paquete malicioso

A continuación vamos a realizar el análisis con la herramienta Dependency Walker[23], la herramienta nos presenta un arbol jerárquico de todos los módulos, allí vamos a encontrar funciones que directamente indican la funcionalidad del troyano, lo invasivo y peligroso que puede llegar a ser porque toma demasiada

[23] Tomado de https://www.aldeid.com/wiki/Dependency-Walker

información de la máquina, escribe sobre registros, lee sobre la memoria y la escribe en archivos en otra ubicación, toma la información de la máquina y podría comunicarse remotamente con el exterior.

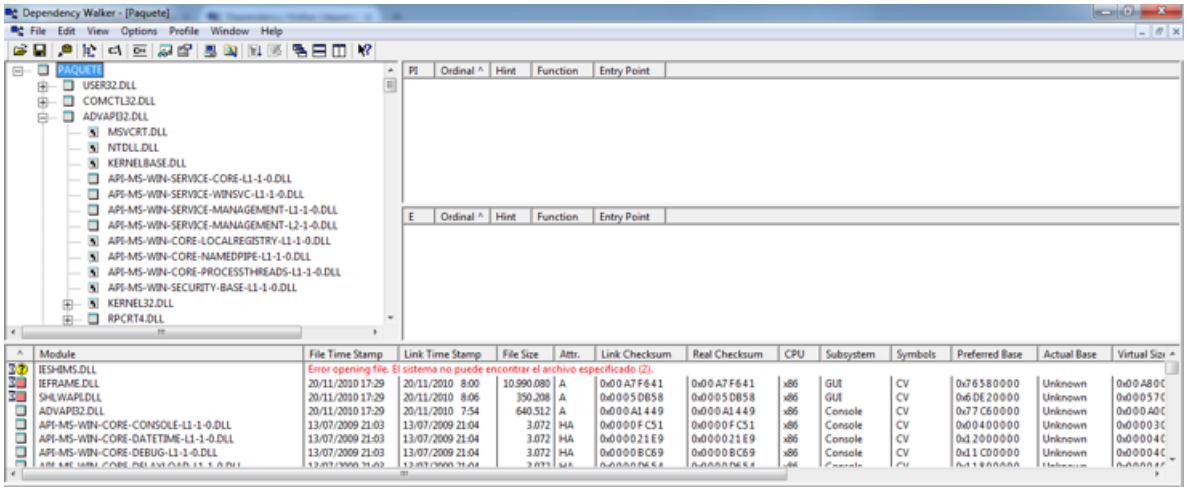


Imagen 29: Evaluación de DLL's con la herramienta Dependency Walker

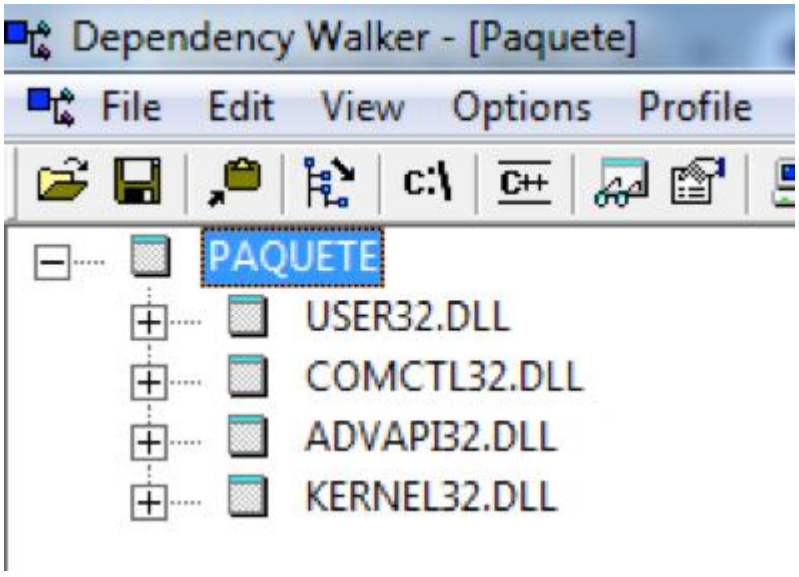


Imagen 30: DLL's identificadas por la herramienta Dependency Walker

PI	Ordinal ^	Hint	Function	Entry Point
C	N/A	98 (0x0062)	LdrAccessResource	0x77F2E7E9
C	N/A	106 (0x006A)	LdrFindResourceEx_U	0x77F2EA82
C	N/A	1890 (0x0762)	floor	0x77EF3EF0
C	N/A	1913 (0x0779)	memcpy	0x77EF40F0
C	N/A	1917 (0x077D)	memset	0x77EF4780

memcpy, wmemcpy

memcpy copies count bytes from src to dest; wmemcpy copies count wide characters (two bytes). If the source and destination overlap, the behavior of memcpy is undefined. Use memmove to handle overlapping regions.

Library: memory.h

See Also:

memcmp, wmemcmp Buffer Manipulation

strcpy_s, wcscpy_s, _mbscpy_s

Signature

```
void *memcpy(void *dest, const void *src, size_t count);
```

Imagen 31: Identificación de función memcpy

PI	Ordinal ^	Hint	Function	Entry Point
C	N/A	11 (0x000B)	GetSystemTimeAsFileTime	0x0DCE6B9D
C	N/A	14 (0x000E)	GetTickCount	0x0DCE6740

GetSystemTimeAsFileTime function

Retrieves the current system date and time. The information is in Coordinated Universal Time (UTC) format.

Syntax

C++

```
void WINAPI GetSystemTimeAsFileTime(
  _Out_ LPFILETIME lpSystemTimeAsFileTime
);
```

Imagen 32: Identificación de función GetSystemTimeAsFileTime

PI	Ordinal ^	Hint	Function	Entry Point
C	N/A	956 (0x03BC)	ReadFile	0x77E296FB
C	N/A	961 (0x03C1)	RegCloseKey	0x77E2CB4F
C	N/A	963 (0x03C3)	RegCreateKeyExW	0x77E20D25
C	N/A	965 (0x03C5)	RegDeleteKeyExW	0x77E16644
C	N/A	974 (0x03CE)	RegEnumValueW	0x77E25233
C	N/A	988 (0x03DC)	RegOpenKeyExW	0x77E2C189
C	N/A	991 (0x03DF)	RegQueryInfoKeyW	0x77E24D75
C	N/A	993 (0x03E1)	RegQueryValueExW	0x77E33218
C	N/A	1000 (0x03E8)	RegSetValueExW	0x77E294B0

RegCreateKeyEx function (Windows) - **msdn.microsoft.com**

The **RegCreateKeyEx** function creates all missing keys in the specified path. An application can take advantage of this behavior to create several keys at once.

https://**msdn.microsoft.com**/en-us/library/windows/desktop/ms724844...

Imagen 33: Identificación de función RegCreateKeyEx

PI	Ordinal ^	Hint	Function	Entry Point
C	N/A	N/A	CM_MapCrToWin32Err	Not Bound
C	N/A	N/A	CMP_RegisterNotification	Not Bound
C	N/A	N/A	CMP_UnregisterNotification	Not Bound

CMP_RegisterNotification

Results **1-1** of **1** for: **CMP_RegisterNotification**

CM_Register_Notification function - Windows 10 hardware dev

CM_Register_Notification function. Use RegisterDeviceNotification instead of CM_Register_Notification if your code targets Windows 7 or earlier versions of Windows.

https://**msdn.microsoft.com**/en-us/library/windows/hardware/hh780224...

Imagen 34: Identificación de función CMP_RegisterNotification

PI	Ordinal ^	Hint	Function	Entry Point
C	N/A	N/A	WinStationGetLoggedOnCount	Not Bound
C	N/A	N/A	WinStationSendMessageW	Not Bound
C	N/A	N/A	WinStationQueryInformationW	Not Bound

[MS-TSTS]: **RpcWinStationSendMessage** (Opnum 7)

The **RpcWinStationSendMessage** method displays a message box on a given terminal server session and, optionally, waits for a reply. The caller MUST have WINSTATION_MSG ...

https://**msdn.microsoft.com**/en-us/library/cc248836.aspx

Imagen 35: Identificación de función WinStationSendMessageW

E	Ordinal ^	Hint	Function	Entry Point
C	1002 (0x03EA)	0 (0x0000)	A_SHAFinal	NTDLL.A_SHAFinal
C	1003 (0x03EB)	1 (0x0001)	A_SHAInit	NTDLL.A_SHAInit
C	1004 (0x03EC)	2 (0x0002)	A_SHAUpdate	NTDLL.A_SHAUpdate
C	1005 (0x03ED)	3 (0x0003)	AbortSystemShutdownA	0x0005DDB4
C	1006 (0x03EE)	4 (0x0004)	AbortSystemShutdownW	0x0005DD60
C	1007 (0x03EF)	5 (0x0005)	AccessCheck	0x0000CA3C
C	1008 (0x03F0)	6 (0x0006)	AccessCheckAndAuditAlarmA	0x000412F9
C	1009 (0x03F1)	7 (0x0007)	AccessCheckAndAuditAlarmW	0x00042FF8

AccessCheck function (Windows) - **msdn.microsoft.com**

The **AccessCheck function** determines whether a security descriptor grants a specified set of access rights to the client identified by an access token.

https://**msdn.microsoft.com**/en-us/library/windows/desktop/aa374815...

Imagen 36: Identificación de función AccessCheck

E	Ordinal ^	Hint	Function	Entry Point
C	4 (0x0004)	1 (0x0001)	AcquireSRWLockShared	NTDLL.RtlAcquireSRWLockShared
C	5 (0x0005)	2 (0x0002)	ActivateActCtx	0x00045911
C	6 (0x0006)	3 (0x0003)	AddAtomA	0x000370DF
C	7 (0x0007)	4 (0x0004)	AddAtomW	0x000425F5
C	8 (0x0008)	5 (0x0005)	AddConsoleAliasA	0x000AB2CA
C	9 (0x0009)	6 (0x0006)	AddConsoleAliasW	0x000AB260
C	10 (0x000A)	7 (0x0007)	AddIntegrityLabelToBoundaryDescriptor	0x0008F7DC
C	11 (0x000B)	8 (0x0008)	AddLocalAlternateComputerNameA	0x00084D7B

AddConsoleAlias function (Windows) - **msdn.microsoft.com**

AddConsoleAlias function. Defines a console alias for the specified executable. Syntax. C++. Copy. BOOL WINAPI **AddConsoleAlias**(_In_ LPCTSTR Source, _In_ LPCTSTR ...

https://**msdn.microsoft.com**/en-us/library/windows/desktop/ms681935...

Imagen 37: Identificación de función AddConsoleAlias

PI	Ordinal ^	Hint	Function	Entry Point
C	N/A	0 (0x0000)	CreateFileMappingW	0x0 DCE DD3 A
C	N/A	1 (0x0001)	FlushViewOfFile	0x0 DCE F8 F F
C	N/A	2 (0x0002)	MapViewOfFile	0x0 DCE DCB6
C	N/A	3 (0x0003)	MapViewOfFileEx	0x0 DCE 8 C2 A
C	N/A	4 (0x0004)	OpenFileMappingW	0x0 DCE 8 BB5
C	N/A	5 (0x0005)	ReadProcessMemory	0x0 DCE 9 A0 A
C	N/A	6 (0x0006)	UnmapViewOfFile	0x0 DCE 6 BF D
C	N/A	7 (0x0007)	VirtualAlloc	0x0 DCE 79 F F
C	N/A	8 (0x0008)	VirtualAllocEx	0x0 DCE 79 B8

Imagen 38: Identificación de función ReadProcessMemory

PI	Ordinal ^	Hint	Function	Entry Point
C	N/A	0 (0x0000)	GetComputerNameExA	0x0 DD1 516 E
C	N/A	1 (0x0001)	GetComputerNameExW	0x0 DCE F A46
C	N/A	2 (0x0002)	GetDynamicTimeZoneInformation	0x0 DCF E AB9
C	N/A	3 (0x0003)	GetLocalTime	0x0 DCE9 236
C	N/A	4 (0x0004)	GetLogicalProcessorInformation	0x0 DCF C86 2
C	N/A	5 (0x0005)	GetLogicalProcessorInformationEx	0x0 DCF F731
C	N/A	8 (0x0008)	GetSystemInfo	0x0 DCE9 5 A3
C	N/A	9 (0x0009)	GetSystemTime	0x0 DCE E20 A
C	N/A	10 (0x000A)	GetSystemTimeAdjustment	0x0 DCF F6 E C

Imagen 39: Identificación de función GetSystemInfo

6.2.3 Análisis de Resultados

El anterior análisis de malware se puede determinar que estabamos enfrentados a un troyano genérico escrito en Febrero del año 2015, el paquete malicioso se presenta como un inofensivo Winamp e incluso tiene funciones de esta herramienta, pero detrás de el se esconde un paquete malicioso de instrucciones que toma información de toda la máquina, toma información de la memoria y la escribe en archivos, puede tomar instrucciones remotamente para ejecutar en la máquina.

Luego de ser instalado este malware cambia los registros del sistema operativo, tiene una librería que maneja el paquete de instrucciones inofensivos (winamp) como los ofensivos (control sobre la máquina), tiene múltiples funciones Get y Set y adicionalmente se evidencia que tiene una rutina para avisar cuando está conectado o desconectado el cliente, presenta una rutina para ejecutarse cada vez que se inicia la máquina.

Quizás el método de propagación es por medio de Phishing ya que el paquete se muestra como un software de Winamp y por medio de alguna página web puede ser descargado e instalado desplegando las funciones maliciosas que puede realizar el malware.

Para la construcción de este malware es muy probable que se utilizó un Joiner el cual es un software que permite que a un software saludable se le pueda añadir un componente de malware, el cual una vez se ejecute pueda realizar las dos funciones, en este caso el malware permitía ejecutar el winamp pero también realizaba las funciones de troyano.

Conforme a la investigación podemos determinar que el troyano era de tipo RAT (Remote Access Trojan) dado que le permitía al atacante informático recolectar una amplia información de la máquina, recolectar datos personales, ejecutar instrucciones remotas desde un command control, entre otro tipo de funciones.

6.3 Herramientas Para Análisis Forense de Malware

Las herramientas forenses son la base esencial de los análisis de evidencias digitales, es imprescindible que el investigador forense tenga un conocimiento total sobre el manejo de las herramientas, y basado en la experiencia y formación hace que estos dos elementos sean vitales para la práctica de análisis forense.

Herramienta	Función	Tipo De Análisis	Licencia	Descripción
Vmware	Elaboración de máquinas virtuales	Estático y Dinámico	Comercial	Vmware es una herramienta que nos va a permitir crear máquinas virtuales para realizar los análisis de malware pertinentes ya bien sea estático o dinámico
IDA Pro	Desensamblador de binarios	Estático y Dinámico	Comercial	IDA Pro es una herramienta que nos va a permitir desensamblar cualquier binario para analizar su naturaleza y su propósito
OllyDbg	Debugger	Dinámico	Free	OllyDbg es una herramienta que nos va a permitir realizar dinámicamente análisis de funcionamiento y comportamiento del paquete malicioso
Virus Total	Analizador de Binarios	Estático	Free	Virus Total es una página web que nos va a permitir analizar un binario confrontándolo contra bases de datos de diversos fabricantes y nos dará un resultado de si el paquete puede ser potencialmente peligroso

Herramienta	Función	Tipo De Análisis	Licencia	Descripción
Payload Security	Analizador de Binarios	Estático	Free	Payload Security es un repositorio de malware y evaluador de comportamiento de diversos binarios, con ellos se puede obtener muestras de malware para su respectivo análisis
Strings	Extracción de cadenas de un binario	Estático	Free	Strings es una herramienta de Sysinternals que nos va a permitir extraer las cadenas de un binario para su respectivo análisis
PEiD	Identificación de binarios	Estático	Free	La herramienta PEiD nos va a permitir revisar la identificación de un binario, y obtener metadata de los mismos
PE Exec	Información de PE	Estático	Free	La herramienta PE Exec nos va a permitir evaluar la información de los ejecutables portables de los binarios y entender la lógica de funcionamiento de las DLL's
Dependency Walker	Indentificación de dependencias de binarios	Estático	Free	Con Dependency Walker podemos ver funciones de librerías y conexiones a librerias del host huesped, podemos evaluar las funciones para entender el comportamiento de los mismos.
Process Monitor Tool	Indentificación de procesos en tiempo real	Dinámico	Free	Process Monitor es una herramienta de Sysinternals que nos va a permitir ver el funcionamiento de los procesos luego de que el malware ha sido implantado en el host

Existen algunas suite de herramientas forenses comerciales que utilizan grandes corporaciones e instituciones de análisis forense a nivel mundial como las que vamos a detallar a continuación:

- **ENCASE**

Encase Forensic es una herramienta de Guidance Software la cual es quizás la herramienta forense comercial más utilizada en el mercado, posee una amplia serie de herramientas para la fase de adquisición y de análisis entre las cuales se destacan:

- ✓ Granularidad de adquisición
- ✓ Reinicio de adquisición
- ✓ Archivos de evidencia lógica
- ✓ CRC: Imagen verificada por comprobación de redundancia cíclica (CRC) y MD5.
- ✓ Adquisición de evidencia en RAM
- ✓ Adquisición de evidencia a través del disco de inicio
- ✓ Analizador de registros
- ✓ Analizador de documentos
- ✓ Analizador de HASH
- ✓ Buscador de archivos en espacio no asignado[24]

- **Access Data Forensic ToolKit (FTK)**

Forensic Toolkit de AccessData® (FTK™) ofrece a los profesionales encargados de controlar el cumplimiento de la ley y a los profesionales de seguridad la capacidad de realizar exámenes forenses informatizados completos y exhaustivos. FTK posee funciones eficaces de filtro y búsqueda de archivos. Los filtros personalizables de FTK permiten buscar en miles de archivos para encontrar rápidamente la prueba que necesita. FTK ha sido reconocida como la mejor

[24] Tomado de http://www.ondata.es/recuperar/encase/spanish_webready_encaseforensicfeaturesheet.pdf

herramienta forense para realizar análisis de correo electrónico. Sus principales funciones son:

- Fácil de usar
- Opciones de búsqueda avanzadas
- Registry viewer
- Análisis de correo electrónico y de archivos zip
- Diseño de capa de base de datos
- Montaje seguro de dispositivos remotos[25]

- **CAINE**

CAINE (Medio ambiente de investigación asistido por computador) es una versión italiana de GNU / Linux, fue creado como un proyecto forense digital CAINE ofrece un completo entorno forense que está organizado para integrar herramientas de software existentes como módulos de software y proporcionar una interfaz gráfica amigable.

Los objetivos de diseño principales de CAINE son los siguientes:

-Un entorno interoperable que admite el investigador digital durante las fases de la investigación digital

- Una interfaz gráfica de usuario amigable
- una recopilación semiautomática del informe final[26]

[25] Tomado de http://informaticaforenseunadcd.blogspot.com.co/p/herramientas-de-software-utilizadas-en.html

[26] Tomado de http://informaticaforenseunadcd.blogspot.com.co/p/herramientas-de-software-utilizadas-en.html

7 CONCLUSIONES

- El análisis forense es una disciplina para el manejo de la evidencia digital, aplica técnicas y procedimientos para el análisis de un crimen informático, o el manejo apropiado de evidencia digital para su investigación.

- El malware es un software que altera el correcto funcionamiento de un PC, laptop, server entre otros, tiene diferentes comportamientos de acuerdo a su naturaleza.

- El análisis forense de malware reúne técnicas y procedimientos para el análisis de código malicioso, identificar la funcionalidad de los binarios y tomar medidas preventivas contra este tipo de amenaza informática.

- Los caballos de troya son un tipo de malware que se presenta ante un usuario como un software benigno y auténtico pero puede ser una suplantación o realiza actividades maliciosas sobre la máquina.

- El análisis forense estático revisa el binario sin ejecutarlo con el fin de analizar sus rutinas, funciones y comportamiento.

- El análisis forense dinámico realiza un análisis de comportamiento del malware cuando este ya ha sido infectado al huésped.

- Las empresas en Colombia actualmente no cuentan con los suficientes profesionales preparados para atender un incidente informático o no cuentan con los procedimientos necesarios para atender un incidente de

seguridad relacionado con malware, así mismo dependen mucho del antivirus para mitigar ataques informáticos relacionados con malware.

- Actualmente se cuenta con una alta cantidad de herramientas libres y comerciales que apoyan el estudio forense de malware.

BIBLIOGRAFÌA

SMITH, Brad A storm (worm) is breawing. IEEExplore computer society, 2008. ISSN 0018-9162.

BORGUELLO, Cristian. Botnets, redes organizadas para el crimen. Colorado: Eset Educational, 2007.

GEER, David. Malicious bots threaten network security. Computer magazine. Ashtabula, 2005. V.38, p. 18-20

GOTH, Greg. The politics of DDoS attacks. IEEE Distributed systems online, 2007. V.8, p. 3.

KRISHNOMOORTHY, Srinivasan. y DASGUPTA, Partha. Tacking congestion to address distributed denial of service: A push – forward mechanism. Globecom'04. 2004. V.4, p. 2055-2060.

BLACKERT, W. GREGG, D. CASTNER, A. KYLE, E. HOM, R. y JOCKERST, R. Analyzing interaction between distributed denial of service attacks and mitigation technologies. Darpa information survivability conference and exposition, 2003. V.1, p. 26-36.

CHONKA, A. WANLEI, Zhou. SINGH, J. y YANG, Xiang. Detecting and tracing DDoS attacks by intelligent decisión prototype. Percom'08, 2008. P. 578-583.

LAURENS, V. SADDIK, A. DHAR, P. y VINEET, Srivastava. Detecting distributed denial of service attack traffic at the agent machines. CCECE'06. p. 2.369-2372.

MACÍA FERNANDEZ, Gabriel. Ataques de denegación de servicio a baja tasa contra servidores. ICICS'06. p. 282-291.

BENSON, CRISTOPHER, Security strategies. Microsoft Technet.

GRAVES, KIMBERLY. Official Certified Ethical Hacker Review Guide. Indianapolis: Wiley Publishing Inc.

BACIK, SANDY. Building an effective information security policy architecture, USA: Taylor & Francis Group.

HAFNER, Michael, y BREU, Ruth. Security engineering for service-oriented architectures. Berlin: Springer.

CANO, Jeimy. Computación forense, Alfaomega

Security By Default (Online). Disponible en http://www.securitybydefault.com/2015/05/recursos-para-analisis-de-malware.html

NIST SP 800-86 (Online). Disponible en http://csrc.nist.gov/publications/nistpubs/800-86/SP800-86.pdf

NIST SP 800-83 (Online). Disponible en http://csrc.nist.gov/publications/nistpubs/800-83/SP800-83.pdf