

UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE CUENCA

CARRERA DE INGENIERÍA DE SISTEMAS

Trabajo de titulación previo a la
obtención del título de Ingeniera
de Sistemas

PROYECTO TÉCNICO

**“METODOLOGÍA PARA EL ANÁLISIS DE
MALWARE EN UN AMBIENTE CONTROLADO”**

AUTORA:

Tatiana María Jumbo Tene

TUTOR:

Ing. Pablo Leonidas Gallegos Segovia

Cuenca – Ecuador

Mayo 2017

DECLARATORIA DE RESPONSABILIDAD

Yo, Tatiana María Jumbo Tene, con número de cédula 0105909931 autora del trabajo de titulación “METODOLOGÍA PARA EL ANÁLISIS DE MALWARE EN UN AMBIENTE CONTROLADO” certifico que el total contenido del Proyecto Técnico es de mí exclusiva responsabilidad y autoría

Cuenca, Mayo del 2016



Tatiana María Jumbo Tene

0105909931

CESIÓN DE DERECHOS DE AUTOR

Yo, Tatiana María Jumbo Tene, con documento de identificación N° 0105909931, manifiesto mi voluntad y cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del trabajo de titulación: "METODOLOGÍA PARA EL ANÁLISIS DE MALWARE EN UN AMBIENTE CONTROLADO", mismo que ha sido desarrollado para optar por el título de: Ingeniera de Sistemas, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



Tatiana Jumbo

0105909931

Mayo del 2017

CERTIFICACIÓN

Yo declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: "METODOLOGÍA PARA EL ANÁLISIS DE MALWARE EN UN AMBIENTE CONTROLADO", realizado por, Tatiana María Jumbo Tene, obteniendo el "Proyecto Técnico", que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana.

Cuenca, Mayo del 2017



Pablo Leonidas Gallegos Segovia

0102593589

AGRADECIMIENTO

Agradezco a Dios por regalarme la vida y la salud, a toda mi familia de manera especial a mis padres Juan y Delia por su esfuerzo y apoyo incondicional, a todos mis profesores de la UPS de forma especial a mi tutor de proyecto Ing. Pablo Gallegos por su tiempo y compromiso para realizar con éxito el proyecto, a la gran familia Taurustech por la confianza entregada y de forma especial al Ing. Paul García por su paciencia al compartirme sus conocimientos sobre seguridad informática.

Tatiana Jumbo

DEDICATORIA

El presente proyecto va dedicado a mis padres Juan y Delia que son quienes me inspiran y motivan cada día de mi vida, a mis hermanos Diana, Leonel, Catherine, Niksa y Juan por todo el apoyo y compañía que me han regalado. A mis sobrinitos Carlitos y Sebitas quienes han terminado siendo ese motor de cada uno de mis proyectos.

Este logro es por ustedes y para ustedes.

Tatiana Jumbo

RESUMEN

“Cada segundo se producen hasta cuatro diferentes malware y es necesario saber cómo combatirlos” KasperskyLab

Muchas entidades financieras son víctimas de ataques dirigidos mediante software malicioso conocido como malware, estos ataques cibernéticos son realizados por hackers cuya finalidad es transferir miles de millones de dólares a nivel mundial hacia paraísos fiscales. Es necesario realizar una investigación sobre los efectos y procesos de estos malware con la finalidad de encontrar mecanismos de prevención, reacción y mitigación.

Dado que el malware es cada vez más avanzado, muchas soluciones de prevención, como el firewall, software antivirus y antispyware, se están viendo superadas, lo cual se debe a que el malware aprovecha las ventajas de la tecnología para ser nocivo, rápido y sutil en la forma de engañar a sus víctimas.

Nuestra investigación se enfoca al análisis de malware, generando un entorno controlado en el que se pueda realizar investigaciones sobre el comportamiento de estos códigos maliciosos, mediante el uso de la herramienta CUCKOO para agilizar el análisis.

ABSTRACT

"Every second occurs up to four different malware and you need to know how to combat them" KasperskyLab

Many financial institutions are victims of targeted attacks by malicious software known as malware, these cyber-attacks are performed by hackers whose purpose is to transfer billions of dollars worldwide towards tax havens. It is necessary to carry out an investigation about the effects and processes of these malware in order to find mechanisms of prevention, reaction and mitigation.

As malware is becoming more advanced, many prevention solutions, such as the firewall, antivirus software and antispymware, are being overcome, which is because the malware takes advantage of the technology to be harmful, quick and subtle in the way of deceiving their victims.

Our Research it focuses on the analysis of malware, generating a controlled environment where you can conduct research on the behavior of these malicious codes, using the tool CUCKOO to expedite the analysis

Contenido

RESUMEN	7
ABSTRACT	8
INTRODUCCIÓN	12
PROBLEMA.....	13
Antecedentes	13
Importancia y Alcances	13
Delimitación	13
Objetivo General:	14
Objetivos Específicos:.....	14
1. ESTADO DEL ARTE DE MALWARE.....	16
1.1. Malware	16
1.1.1. Concepto	16
1.1.2. Historia	16
1.1.3. Clasificación del malware.....	17
1.1.4. Ingeniería Social	18
1.1.5. Técnicas de Propagación de malware.....	18
1.1.6. Ciclo de un ataque de malware.....	19
1.2. Técnicas y Herramientas para el Análisis de Malware.....	20
1.2.1. Tipos de Análisis de Malware.....	20
1.2.2. Herramientas para el análisis de malware.....	20
2. MARCO METODOLÓGICO.....	23
2.1. Instalación de Cuckoo Sandbox.....	24
2.1.1. Requerimientos de Hardware	24
2.1.2. Instalación de paquetes	24
2.1.3. Instalación de Cuckoo	26
2.1.4. Instalación de los equipos invitados (víctima)	27
2.2. Configuración de Cuckoo Sandbox.....	30
2.3. Colocar muestra de malware	33
2.4. Información recolectada	36
3. RESULTADOS	42

4. POLÍTICAS DE PREVENCIÓN, REACCIÓN Y MITIGACIÓN DE MALWARE	51
4.1. Prevención ante ataque de malware	51
4.2. Reacción ante ataque de malware.....	52
4.3. Mitigación ante ataque de malware	53
CONCLUSIONES	54
RECOMENDACIONES.....	54
Referencias.....	55

Índice de Ilustraciones

Ilustración 1: Incremento del Malware en los últimos cinco años [4]	16
Ilustración 2: Ciclo de un ataque de malware [12]	19
Ilustración 3: Escenario Virtual Cuckoo Sandbox.....	23
Ilustración 4: Estadísticas de los Sistemas Operativos utilizados [18]	24
Ilustración 5: Resultado de construcción python.....	26
Ilustración 6: Deshabilitar UAC en victima.....	27
Ilustración 7: Puerto python ejecutándose.....	28
Ilustración 8: Snapshot maquina victima	29
Ilustración 9: snapshot por interface gráfica 1	29
Ilustración 10: snapshot por interface gráfica 2	29
Ilustración 11: Configurar archivo cuckoo.conf	30
Ilustración 12: Configurar archivo virtualbox.conf.....	31
Ilustración 13: Configurar archivo reporting.conf.....	31
Ilustración 14: Ejecutar Cuckoo.....	32
Ilustración 15: Colocar muestra de malware por línea de comandos	33
Ilustración 16: Iniciar servicio web.....	33
Ilustración 17: Colocar muestra de malware por interface grafica	34
Ilustración 18: Mensaje exitoso	34
Ilustración 19: Inicio de análisis por Cuckoo	35
Ilustración 20: Mensaje de análisis completado	35
Ilustración 21: Base de Datos de tareas	35
Ilustración 22: Resultado del Análisis de Cuckoo	36
Ilustración 23: Detalle INFO	36
Ilustración 24: Detalle FILE.....	37
Ilustración 25: Detalle SIGNATURES.....	37
Ilustración 26: Detalle SCREENSHOTS	38

Ilustración 27: Detalla STATIC ANALYSIS	39
Ilustración 28: Detalle DROPPED FILES	39
Ilustración 29: Detalle NETWORK ANALYSIS	40
Ilustración 30: Detalle BEHAVIOR SUMMARY	41
Ilustración 31: Detalle PROCESSES	41
Ilustración 32: Tipo de malware Caso1	42
Ilustración 33: Nombre del malware y su creador	42
Ilustración 34: Conexión sitios externos Caso 1	43
Ilustración 35: Validación de dirección IP	43
Ilustración 36: Filtro DNS wireshark.....	44
Ilustración 37: Filtro para visualizar peticiones wireshark.....	44
Ilustración 38: Archivos descargados	45
Ilustración 39: Instalación Wise Game Booster 1	46
Ilustración 40: Instalación Wise Game Booster 2	46
Ilustración 41: Instalación Wise Game Booster 3	47
Ilustración 42: Instalación Wise Game Booster 4	47
Ilustración 43: Modifica archivos del sistema	48
Ilustración 44: Modifica llaves en el registro	48
Ilustración 45: Tipo de malware Caso 2	49
Ilustración 46: Informe VirusTotal	49
Ilustración 47: Conexión Sitios externos Caso2	50
Ilustración 48: Solicitud del malware Caso2	50
Ilustración 49: Cumplimiento de Políticas [21]	51
Ilustración 50: Fases para atender un incidente de infección por malware [24]	52

Índice de Tablas

Tabla 1: Requerimientos de Hardware	24
---	----

INTRODUCCIÓN

El presente trabajo tiene la finalidad de presentar un laboratorio seguro para llevar a cabo un automatizado análisis de muestras de malware. Esperando reducir la cantidad de tiempo analizando un malware mediante la utilización de Cuckoo Sandbox.

Implementaremos pruebas para determinar el ciclo de vida del malware, características de propagación, infección y recolección de datos, cómo afecta a los sistemas comprometidos, comprender las motivaciones y los objetivos del ataque para generar políticas de seguridad que apoyen a las empresas a garantizar la confidencialidad, integridad y disponibilidad (CID).

Nuestro enfoque va hacia el análisis de malware dinámico que permite la detección de malware en función de su comportamiento esto implica la ejecución de las muestras y la observación de sus acciones en tiempo de ejecución. Se utilizarán muestras reales de la base de conocimiento de Cisco FireAMP, ejecutadas sobre entornos controlados, usando máquinas virtuales dentro de vmware, Además utilizaremos como nuestra víctima a una maquina con Windows 7, por la gran utilización de este sistema en el mundo empresarial.

PROBLEMA

Antecedentes

En la actualidad todos contamos con al menos un dispositivo conectado a internet, el internet forma parte de nuestra forma de vida directa o indirectamente, si bien es una herramienta de comunicación, que facilita nuestra forma de vida, también genera un gran problema debido a que somos tan dependientes de la tecnología, que si algún momento nos falla no podríamos vivir con normalidad.

El software malicioso conocido como malware aprovecha las ventajas ofrecidas por la tecnología para ser nocivo, rápido y sutil en la forma de engañar a cualquier usuario de internet.

Dado a la gran cantidad de usuarios en internet y a que el malware es cada vez más avanzado, muchas soluciones de prevención, como el firewall, software antivirus y antispyware, se están viendo superadas, y no son lo suficientemente seguras para la navegación en internet y utilización de la tecnología.

Importancia y Alcances

El Banco del Austro de Ecuador fue víctima de un ataque de malware que eludió los sistemas de seguridad del banco, generando un perjuicio al banco por 12 millones de dólares los que fueron transferidos a cuentas en Hong Kong. [1]

El día viernes 21 de Octubre de 2016 se registró un ataque masivo hacia DyN Managed DNS, fue un ataque desde una botnet, el ataque lo realizaron webcams, dvrs, impresoras, módems, etc. Se considera que un ataque de este tipo puede desestabilizar económicamente a cualquier país. [2] [3]

Los ataques de malware no necesariamente son realizados hacia grandes organizaciones cualquier persona con acceso a internet puede ser víctima de estos ataques.

El alcance de este proyecto es apoyar a cualquier persona, a validar si está siendo o va ser víctima de un ataque de malware, con ayuda de la información presentada por la herramienta Cuckoo. Permitiendo colocar la muestra de malware bajo un ambiente controlado, para el análisis de la misma.

Delimitación

Nuestra investigación se enfoca al análisis dinámico de malware, generando un entorno controlado en el que se pueda realizar investigaciones sobre el comportamiento de estos códigos maliciosos, mediante el uso de la herramienta CUCKOO.

Aplicada bajo una arquitectura virtual en vmware, contando con una maquina victima a la que se enviaran las muestras de malware y una maquina central a la que le llega la información de las actividades que realiza el malware durante su ejecución.

OBJETIVOS

Objetivo General:

Entender el ciclo de vida del malware, características de propagación, infección y recolección de datos, cómo afecta a los sistemas comprometidos, comprender el contexto, las motivaciones y los objetivos del ataque para apoyar a garantizar la confidencialidad, integridad y disponibilidad (CID).

Objetivos Específicos:

- Análisis del estado del arte del Malware
- Generar un escenario en sistemas virtuales, para realizar pruebas de contexto.
- Implementar un caso de estudio.
- Usar la herramienta CUCKOO para análisis Malware
- Generar políticas de prevención, reacción y mitigación de malware.

GLOSARIO

Amenaza: es todo elemento, acción o evento capaz de atentar contra la seguridad de la información, causando una alteración total y/o parcial a la información de la organización, generando un impacto negativo de tipo material, económico, informativo o prestigio de esta.

Víctima: son los afectados ante una amenaza; puede ser cualquier persona natural o jurídica, o puede ser una organización, incluso puede ser toda una nación.

Snapshot: del inglés foto instantánea, se pueden guardar determinados estados del sistema virtualizado como punto base para el análisis para correrla cuando sea necesario.

Yara: es una herramienta que sirve para identificar y clasificar muestras de malware

Vulnerabilidad: es una debilidad en el diseño o implementación de algún sistema informático, que puede ser utilizado por un atacante, violando la seguridad del mismo, para ocasionar algún daño

Clave de registro: de Windows es un elemento en el que se guardan las especificaciones de configuración del PC mediante claves. Estas claves cambiarán de valor y/o se crearán cuando se instalen nuevos programas o se altere la configuración del sistema.

Control y mando: de botnets son redes de ordenadores infectadas con malware. Los atacantes pueden controlarlas como un grupo y ordenarles que realicen una tarea concreta

1. ESTADO DEL ARTE DE MALWARE

Dentro de Cybercrimen¹ el campo del malware se ha incrementado de una manera acelerada, según los reportes del instituto AV-TEST [4] el malware ha crecido en un 500% en los últimos 5 años como se muestra en Ilustración 1: Incremento del Malware en los últimos cinco años Tomada el 25/08/2016

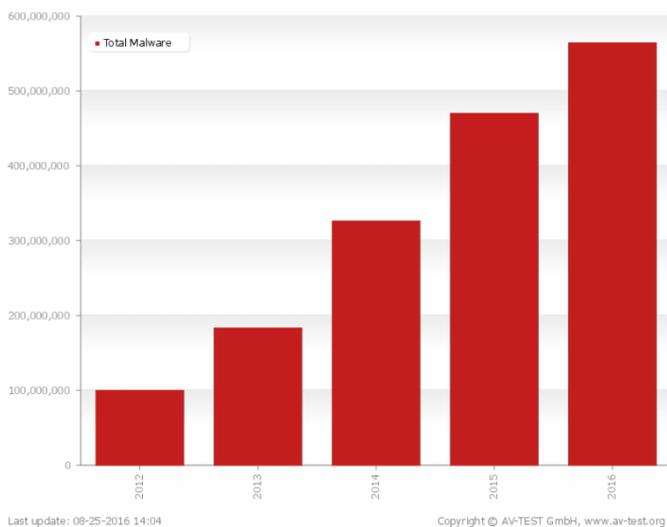


Ilustración 1: Incremento del Malware en los últimos cinco años [4]

1.1. Malware

1.1.1. Concepto

Malware, acrónimo en inglés de las palabras “malicious” y “software” y que traducido al español equivale a software malicioso [5]. Por definición hace referencia a software desarrollado con un fin malintencionado y engloba todos los tipos conocidos, entre ellos: Virus, Gusanos, Troyanos, Ransomware, Rootkit, Backdoor, Downloader y Spyware

1.1.2. Historia

Según SecureList de Kaspersky los programas maliciosos no son nuevos, estos se fueron desarrollando a la par de la tecnología. “Los primeros malware causaban temor en los usuarios dado que las computadoras se comportaban de formas inesperadas. Sin embargo, a partir de los años noventa, el malware se convirtió en una amenaza seria, capaz de robar información confidencial como detalles de las cuentas bancarias y contraseñas” [6]

Tanto ESET [7] como Kaspersky [6] hablan de un juego llamado “Darwin” como uno de los primeros malware desarrollados. En donde los programas podían atacar y destruir a los

¹ Cybercrimen: involucra el uso de equipos digitales como computadoras, Smartphone, tablets para promover o realizar prácticas ilegales como robo de información personal, difamaciones, etcétera [26]

programas del oponente y lo más importante, reproducirse, el objetivo del juego era borrar los programas del oponente y ganar control sobre el campo de batalla.

“Darwin” es una muestra de que los primeros malware fueron creados sin ningún objetivo de daño o destrucción, sin embargo, las operaciones modernas de malware son sofisticadas, están bien financiadas y son capaces de generar interrupciones importantes en las organizaciones debido a que son creadas con fines delictivos.

1.1.3. Clasificación del malware

KasperskyLab detalla los tipos de malware que existen, entre los que se encuentran, están los botnet que reúne un conjunto de equipos infectados y los ponen a realizar operaciones ilegales [8].

Virus

Es un código que se puede auto-replicar y se distribuye así entre varios sistemas informáticos, de ahí su nombre por la analogía con los virus biológicos, se instala sin el consentimiento del usuario infectando archivos existentes [5] [8].

Gusanos

Al igual que los virus se pueden auto-replicar, pero la diferencia es que no infecta archivos existentes, sino que simplemente se instalan o se ejecutan en la memoria ram [8].

Troyanos

Su cualidad es que aparentan ser programas legítimos, se camuflan dentro de software no malicioso, su funcionalidad de asemeja a la de un parásito, se difunden a través de internet [9].

Pueden ser clasificados según el daño que causan, de la siguiente manera:

- **Downloader:** descarga y ejecuta otros códigos maliciosos
- **Banker:** posee como objetivo el robo de credenciales de acceso financieras
- **Dropper:** se ejecuta en paralelo con un programa legítimo.
- **Clicker:** busca beneficio económico a través de clics en publicidad
- **Keylogger:** registra las actividades que se realizan en el sistema
- **Backdoor:** abre puertos en el sistema sin autorización
- **Bot:** convierte el sistema en zombi

Adware

Este tipo de malware se instala en el sistema sin que el usuario lo note. Su función es descargar y/o mostrar textos o imágenes de publicidad en la pantalla de la víctima [8] [9].

Spyware

Este tipo de malware es desarrollado exclusivamente para recolectar información generado por los usuarios en los sistemas informáticos [8] [9].

Rogue

Simula ser un programa de seguridad, indica falsas alertas sobre infecciones o problemas que pudiera tener el sistema [9].

Ransomware

Es un software desarrollado para extorsionar a sus víctimas, a las cuales se le muestra un mensaje informándoles sobre una falta cometida (Ver pornografía, usar software pirata, etc.) Pidiendo dinero a cambio de su Exoneración [5]

Rootkit

Su principal función es evadir la detección por parte del usuario o de herramientas automatizadas de seguridad. Cuando arranca antes del sistema operativo se lo conoce como bootkit [5] [8]

1.1.4. Ingeniería Social

La ingeniería social es una de las técnicas más efectivas utilizadas para realizar una infección. Su efectividad se debe a la falta de cultura en seguridad informática que tienen las personas [10]

La ingeniería social manipula a las personas ya sea con el fin de obtener información confidencial o hacerlas partícipes activos del ataque, provocando por ejemplo que ellas mismo infecten sus equipos informáticos.

Los vectores de ataque pueden ser muchos, como llamadas telefónicas, chat, email, etc. En estos casos resulta un tanto complejo la aplicación de controles de seguridad.

1.1.5. Técnicas de Propagación de malware

Se detallan los métodos más usadas por el malware para su propagación

Redes Sociales

Involucrando la técnica de ingeniería social para atraer a los usuarios, los delincuentes utilizan las redes sociales para propagarse, debido a que estas contienen una gran cantidad de información

Correos electrónicos

Utiliza la ingeniería social para engañar a los usuarios y lograr obtener información de ellos, con el fin de compartir correos basura, direcciones de páginas falsas [11]

Sistemas Operativos

Sin importar la plataforma donde se ejecute el malware, siempre trata de buscar las vulnerabilidades de los sistemas

1.1.6. Ciclo de un ataque de malware

Ramos [12] describe las etapas que un cibercriminal realiza con el objetivo de obtener algún beneficio, desde el inicio de un ataque hasta el robo de información



Ilustración 2: Ciclo de un ataque de malware [12]

El **análisis** de su víctima que puede ser una entidad o una persona específica, determina lo que el malware quiere conseguir de él, las acciones que realizará, respondiendo a la pregunta ¿Qué ataca?

A continuación, está el **desarrollo** en donde se crea el código malicioso y la forma de ataque a la víctima, utilizando sofisticadas herramientas el malware toma la forma de archivos, aplicaciones, sistemas y programas que aparentan ser serias, inofensivas y atractivas para los usuarios, respondiendo a la pregunta ¿Cómo lo hace?

Seguido está la **propagación**, mediante la interacción del usuario, más personas pueden ser víctimas del ataque, el malware utiliza técnicas de ingeniería social y el entorno de red local para la propagación en los otros equipos presentes en la red, respondiendo a la pregunta ¿cómo se propaga?

Luego está la **infección** que lleva a cabo las acciones para lo que fue creado, utiliza la explotación de vulnerabilidades. Puede causar daños en el sistema, como la eliminación de archivos, el cambio en el registro, modificación del archivo, recogida de datos/información confidencial, y así sucesivamente.

Finalmente, la **recolección de datos** una vez que los usuarios han interactuado con el ataque, los cibercriminales tienen acceso a la información recolectada. *Muchos malware se conectan con sitios en Internet para descargar archivos adicionales o para enviar información que roba del sistema* [13]

1.2. Técnicas y Herramientas para el Análisis de Malware

El análisis de malware para Oktavianto y Muhardianto es el proceso de identificar el comportamiento de las actividades del malware, lo que están haciendo, lo que quieren, y cuáles son sus principales objetivos. *El objetivo del análisis de malware es obtener una comprensión de cómo funciona un malware, por lo que podemos proteger nuestra organización mediante la prevención de los ataques de malware* [14]

1.2.1. Tipos de Análisis de Malware

Existen dos tipos comunes de técnicas para el proceso del análisis de malware usadas por los analistas de malware: análisis estático (se analiza el código) y análisis dinámico (se analiza el comportamiento). Estas dos técnicas permiten a los analistas entender rápidamente y de forma detallada, los riesgos y las intenciones de una muestra de malware.

Análisis Estático

Para realizar el análisis estático, se necesita una fuerte comprensión en programación y conceptos de lenguaje ensamblador x86, además de una gran cantidad de tiempo. Durante el proceso de análisis estático, no se ejecuta el software malicioso. Primero se realiza el desmontaje y descompilación, y después se puede analizar el código ensamblador de bajo nivel. El malware moderno implementa sistemas anti-depuración para evitar el análisis de fragmentos de código [10].

Análisis Dinámico

En el análisis dinámico, que es el que se utilizara en el presente trabajo, se realiza la ejecución del malware para observar su actividad y los cambios que se producen cuando se está ejecutando el malware. Al realizar este tipo de análisis, se necesita un ambiente seguro y la red no debe conectarse a la red de producción. La ventaja de realizar análisis dinámico es que se puede entender completamente cómo funciona un malware, pudiendo supervisar los cambios realizados en el sistema de archivos, registro, procesos, y su comunicación en la red [10].

Dentro de las tareas a realizar durante el análisis dinámico de un código malicioso se definen los siguientes:

- Análisis de los procesos en tiempo real
- Monitorización del registro
- Monitorización de la creación, eliminación o modificación de archivos
- Monitorización del tráfico de red

1.2.2. Herramientas para el análisis de malware

A medida que el malware se hizo más sofisticado, necesitamos más herramientas que permitan analizar el malware fácilmente sin comprometer nuestro sistema. Nombrando algunas de estas herramientas tenemos: IDA free, Ollydbg 2.0, Hiew demo, FAR Manager, **Sandbox**, Fiddler, Wireshark, Notepad++

Herramientas para el análisis estático

Algunas de las herramientas utilizadas para el tipo de análisis estático son las siguientes:

- IDA free: es un desensamblador y depurador de múltiples procesadores alojado en Windows, Linux o Mac OS X. Para su descarga visitar en www.hex-rays.com/products/ida/support/download_freeware.shtml
- OllyDbg: permiten realizar una inspección de la amenaza a un nivel avanzado, se requiere más práctica y comprensión acerca de la estructura del sistema operativo. Se lo puede obtener desde www.ollydbg.de/version2.html
- Notepad++: es un editor de código fuente gratuito, soporta varios idiomas. Se ejecuta en el entorno MS Windows. Se lo puede obtener desde notepad-plus-plus.org/
- Hiew demo: permite ver y editar archivos de cualquier longitud en los modos texto, hexadecimal y decodificación. Página oficial www.hiew.ru/
- FAR Manager: Es un programa para administrar archivos y carpetas en sistemas operativos Windows. Página oficial www.farmanager.com/download.php?l=en

Herramientas para el análisis dinámico

Entre las herramientas que utilizan los analistas de malware detallamos las siguientes:

- FAR Manager: Es un programa para administrar archivos y carpetas en sistemas operativos Windows. Para su descarga www.farmanager.com/download.php?l=en
- Sandbox: utiliza tecnología de aislamiento de programas para evitar que cambios no deseados ocurra a sus datos, programas y aplicaciones personales. Cuckoo Sandbox forma parte de esta herramienta. cuckoosandbox.org/2015-03-04-cuckoo-sandbox-12.html
- Fiddler: proxy de depuración web gratuito para cualquier navegador, sistema o plataforma www.telerik.com/fiddler
- Wireshark: un analizador de red que permite capturar todas las comunicaciones para su posterior análisis www.wireshark.org/
- Notepad++: es un editor de código fuente gratuito, soporta varios idiomas. Se ejecuta en el entorno MS Windows. Se lo puede obtener desde notepad-plus-plus.org/

Este proyecto se enfoca en la utilización de Cuckoo Sandbox como herramienta de análisis de malware, a continuación, una explicación más detallada de esta herramienta.

Sandboxing

Es una técnica para el aislamiento de un programa (en este caso, malware) proporcionando entornos de ejecución confinados, que se puede utilizar para ejecutar programas no fiables desde el entorno principal [14]. Provee un conjunto de recursos controlados para que las pruebas se ejecuten en los sistemas invitados, como el acceso a la red.

Existe una gran variedad de sistemas Sandbox, entre ellas esta Cuckoo Sandbox, del que trataremos en este proyecto

Cuckoo

Es un sistema automatizado de análisis de malware de código abierto. Se utiliza para ejecutar y analizar automáticamente los archivos y recoger los resultados completos de análisis, describen lo que hace el software malicioso mientras se ejecuta dentro de un sistema operativo aislado [14] [15].

CUCKOO SANDBOX

Cuckoo Sandbox es un sistema de análisis de malware. Significa que se puede lanzar cualquier archivo sospechoso en él y en cuestión de segundos Cuckoo va a dar vuelta algunos resultados detallados que describa lo que hizo de archivos cuando se ejecuten dentro de un entorno aislado [16].

Está diseñado para su uso en el análisis de los siguientes tipos de archivos [15]:

- Ejecutables de Windows genérico
- Archivos DLL
- Documentos PDF
- documentos de Microsoft Office
- URL
- scripts PHP
- Archivos ZIP

A continuación, se describen los principales archivos que contiene Cuckoo:

cuckoo.conf

Este archivo de configuración contiene información sobre el comportamiento general y las opciones de análisis en Cuckoo Sandbox.

virtualbox.conf

Este archivo contiene la información acerca de la configuración de su máquina virtual víctima.

reporting.conf:

Este archivo contiene información sobre metodologías de los reportes.

2. MARCO METODOLÓGICO

Construiremos un laboratorio seguro y controlado para analizar el malware con VMware Workstation, que es una de las herramientas de virtualización más potente en la actualidad.

Bajo este laboratorio podemos supervisar y determinar las actividades del malware. A pesar de que con frecuencia el malware utiliza técnicas de evasión, al sospechar que está siendo ejecutado bajo un ambiente virtual.

Para el análisis dinámico es necesario establecer un laboratorio complejo, ya que debemos ejecutar el malware para observar su comportamiento. El malware se comporta de manera diferente dependiendo del sistema operativo y el entorno en el que se está ejecutando [17]. La configuración de la red donde vamos a colocar nuestro escenario es un punto importante, debido a que algunos malware suelen ser auto-replicante, otros utilizan el internet para comunicarse con su servidor de malware autor.

Dentro de nuestro escenario virtual de Cuckoo Sandbox, contamos con el equipo principal que consiste en un software central de gestión bajo el sistema operativo Ubuntu, este se ocupa de las ejecuciones de muestras de malware y análisis, ya que es aquí donde podremos ver los resultados de las muestras, también se encuentran las maquinas victimas que se conectan en red con la maquina central Cuckoo, que consisten en un software cliente bajo el sistema operativo windows7, son las maquinas que serán infectadas con el malware.

El siguiente diagrama muestra la arquitectura del escenario virtual:

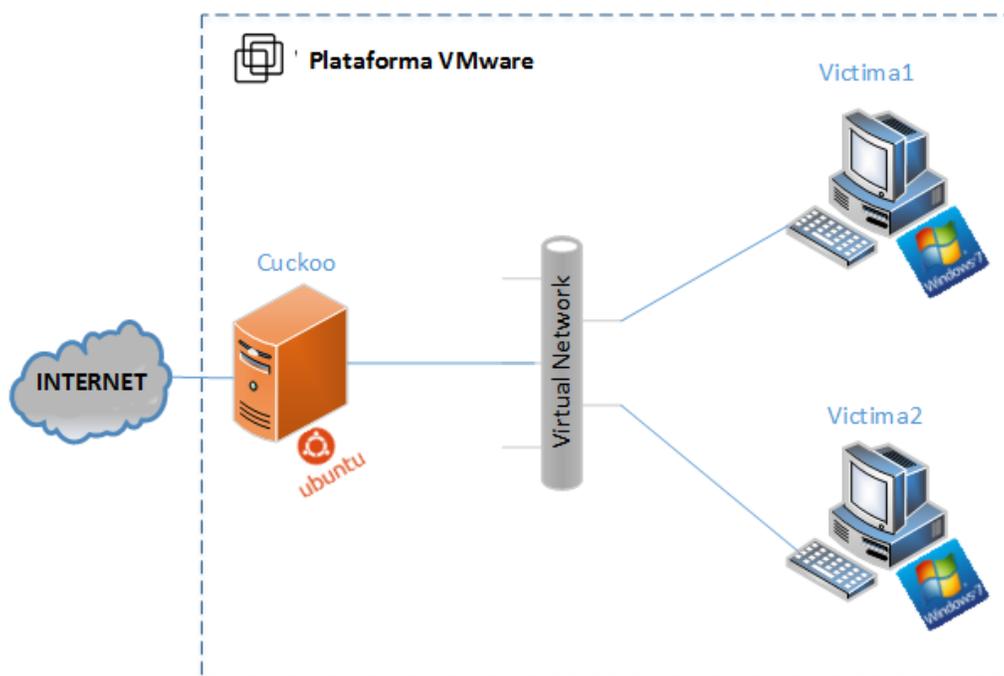


Ilustración 3: Escenario Virtual Cuckoo Sandbox

2.1. Instalación de Cuckoo Sandbox

La instalación y configuración de Cuckoo Sandbox se basa en las recomendaciones realizadas en la página oficial de Cuckoo Sandbox docs.cuckoosandbox.org.

2.1.1. Requerimientos de Hardware

La siguiente tabla muestra los requerimientos de los equipos, tanto para el equipo principal en donde se instalará Cuckoo, como en el equipo victima donde se correrán muestras de malware real.

	Equipo Principal (Cuckoo)	Equipo Invitado (Victima)
<i>Procesador</i>	4	1
<i>Memoria</i>	2 GB	512 MB
<i>Disco Duro</i>	80 GB	25 GB
<i>SO</i>	Ubuntu 16.04.1	Windows 7

Tabla 1: Requerimientos de Hardware

Es necesario que nuestra victima trabaje con Windows 7 debido a que según las estadísticas presentadas en netmarketshare [18] Windows 7 es el sistema operativo más utilizados en el mundo empresarial.

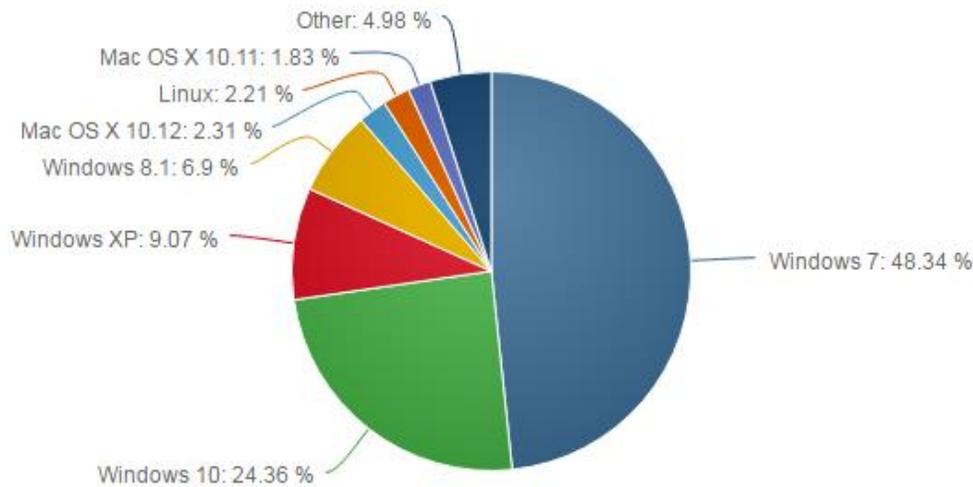


Ilustración 4: Estadísticas de los Sistemas Operativos utilizados [18]

2.1.2. Instalación de paquetes

Una vez instalada la maquina principal, instalaremos todos los paquetes necesarios para el correcto funcionamiento de la SandBox.

Dentro del equipo principal, inicialmente instalaremos los paquetes de Python, ejecutando los siguientes comandos.

```
# apt-get install python
```

Cuckoo necesita la aplicación sqlalchemy como el conjunto de herramientas de base de datos para Python.

```
# apt-get install python-sqlalchemy
```

Se instalan diversos paquetes Python,

```
# apt-get install python-dpkt python-jinja2 python-magic python-pymongo python-libvirt python-bottle python-pefile ssdeep
```

- dpkt: se utiliza para extraer información de los archivos PCAP
- jinja2: se utiliza para informes HTML e interfaz web
- magic: se utiliza para identificar formatos de archivos
- pymongo: se utiliza para almacenar los resultados en una base de datos MongoDB
- libvirt: usa el administrador de la máquina KVM
- bottlepy: utiliza las utilidades web.py y api.py
- pefile: se utiliza para el análisis estático de binarios PE32

Instalar archivos adicionales que necesita para ejecutar python

```
# apt-get install gcc python-socks  
# apt-get install libfuzzy-dev  
# apt install subversion  
# apt-get install python-dev  
# apt-get install python-requests  
# apt-get install python-pyrex
```

Instalar ssdeep que se utiliza para calcular Hash fuzzy o archivos

```
# apt-get install ssdeep
```

A continuación, se instalan algunas dependencias

```
# apt-get install build-essential git libpcre3 libpcre3-dev libpcre++-dev
```

- build-essential: contiene las herramientas necesarias para crear, compilar e instalar programas.
- git: permite administrar automáticamente la vinculación de archivos desde cualquier repositorio git.
- libpcre3, libpcre3-dev y libpcre++-dev: se requieren para la instalación de yara

Instalar pydeep para hash fuzzy ssdeep de muestras. Dentro de la carpeta `opt` clonar pydeep desde la fuente git.

```
$ cd /opt  
$ git clone https://github.com/kbandla/pydeep.git pydeep
```

Se compila e instala el paquete con Python. Dentro de la carpeta `opt/pydeep`

```
$ cd /opt/pydeep/  
# python setup.py build  
# python setup.py install
```

```

root@ubuntu:/opt/pydeep# python setup.py build
running build
running build_ext
root@ubuntu:/opt/pydeep# sudo python setup.py install

```

Ilustración 5: Resultado de construcción python

Se debe obtener el resultado de la Ilustración 5: Resultado de construcción python, para continuar con la instalación. Seguido necesitará instalar yara para clasificar las muestras de malware

```
# apt-get install automake -y
```

Dentro de la carpeta `opt`. Descargar, descomprimir e instalar el paquete Yara (`yara.tar.gz`).

```
$ cd /opt
# tar -xzf yara.tar.gz
```

Dentro de la carpeta `opt/yara`.

```
# cd /opt/yara
# ln -s /usr/bin/aclocal-1.11 /usr/bin/aclocal-1.12
# ./configure
# make
# make install
```

Pegar `yara-python` dentro de `yara` y darle permisos. `yara` y `yara-python` se utiliza para hacer coincidir las firmas Yara (use la versión `svn`)

```
$ cd yara-python
```

Se compila e instala el paquete con Python.

```
# python setup.py build
# python setup.py install
```

Instalar `tcpdump`, que es un analizador de paquetes que circulan a través de una red

```
# apt-get install tcpdump
```

Establecer capacidades específicas de Linux

```
# setcap cap_net_raw,cap_net_admin=eip /usr/sbin/tcpdump
```

Para verificar los resultados del comando anterior

```
$ getcap /usr/sbin/tcpdump /usr/sbin/tcpdump =cap_net_admin,cap_net_raw+eip
```

2.1.3. Instalación de Cuckoo

Instalados todos los paquetes y librerías de la sandbox, descargar Cuckoo, descomprimir e instalar

```
# git clone git://github.com/cuckoobox/cuckoo.git
# tar -zxvf cuckoo-current.tar.gz
```

2.1.4. Instalación de los equipos invitados (víctima)

La máquina víctima que se encuentra virtualizada dentro de la máquina principal con virtualbox, se configura de la siguiente manera

Inicialmente configurar la red del equipo invitado. Colocar el tipo de red en **Host-only Adapter**, de esta forma la muestra de malware tendrá acceso únicamente a lo permitido por la máquina principal

Compartir una carpeta entre el equipo principal y el equipo víctima, para compartir información entre estas.

Instalar Python, para este caso se utilizó Python 3.5.2, el equipo invitado al ser el cliente de la máquina principal, requiere tener las librerías de python.

Instalar PIL de acuerdo a la versión de Python instalada, PIL permite obtener capturas de pantallas mientras se realiza un análisis de malware

Deshabilitar los servicios de actualizaciones automáticas y el Firewall de Windows, para que las muestras de malware se puedan ejecutar bien.

Deshabilitar el Control de Cuentas de Usuario (UAC), para garantizar que el equipo no bloquee ninguna acción del malware.

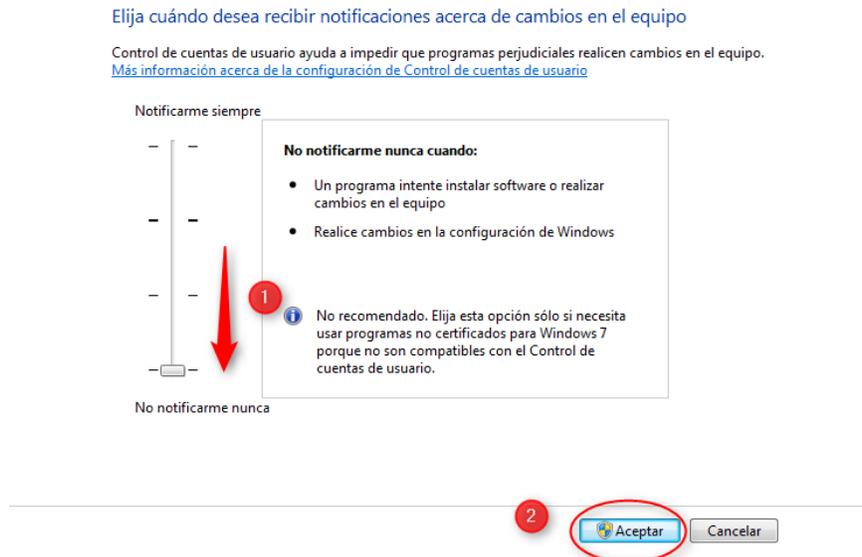


Ilustración 6: Deshabilitar UAC en víctima

Luego de configurar los atributos de la máquina. Se debe realizar lo siguiente para hacer que la máquina víctima sea un cliente de la máquina principal.

1. Copiar el agente Python desde el equipo principal a la carpeta compartida
\$ cp /opt/cuckoo/agent/agent.py /home/tatty/Documents/
2. Copiar el archivo agent.py a la carpeta C:\Python27 del equipo invitado

3. Cambiar el nombre del archivo `agent.py` a `agent.pyw` para evitar que la ventana de Python se muestre y el agente se ejecute en background.
4. Colocar el `agent.pyw` en la carpeta de Startup. Para cuando se encienda el equipo invitado, automáticamente se ejecute el agente de Python

C:\Users\victimaI\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup.

Python utiliza el puerto 8000, comprobamos que se esté ejecutando con el siguiente comando

```
$ netstat -aon
```

```

Administrator: C:\Windows\system32\cmd.exe
C:\Users\victimaI>netstat -aon

Conexiones activas

Proto  Dirección local      Dirección remota      Estado      PID
TCP    0.0.0.0:135          0.0.0.0:0             LISTENING   708
TCP    0.0.0.0:445          0.0.0.0:0             LISTENING   4
TCP    0.0.0.0:5357         0.0.0.0:0             LISTENING   4
TCP    0.0.0.0:8000         0.0.0.0:0             LISTENING   1988
TCP    0.0.0.0:49152        0.0.0.0:0             LISTENING   372
TCP    0.0.0.0:49153        0.0.0.0:0             LISTENING   756
TCP    0.0.0.0:49154        0.0.0.0:0             LISTENING   460
TCP    0.0.0.0:49155        0.0.0.0:0             LISTENING   928
TCP    0.0.0.0:49156        0.0.0.0:0             LISTENING   452
TCP    0.0.0.0:49157        0.0.0.0:0             LISTENING   1996
TCP    192.168.56.101:139   0.0.0.0:0             LISTENING   4
TCP    [::]:135             [::]:0                LISTENING   708
TCP    [::]:445             [::]:0                LISTENING   4
TCP    [::]:5357            [::]:0                LISTENING   4
TCP    [::]:49152           [::]:0                LISTENING   372
TCP    [::]:49153           [::]:0                LISTENING   756
TCP    [::]:49154           [::]:0                LISTENING   460
TCP    [::]:49155           [::]:0                LISTENING   928
TCP    [::]:49156           [::]:0                LISTENING   452
TCP    [::]:49157           [::]:0                LISTENING   1996
UDP    0.0.0.0:500          *:.*                  *:.*        928
UDP    0.0.0.0:3702         *:.*                  *:.*        1508
UDP    0.0.0.0:3702         *:.*                  *:.*        1508
UDP    0.0.0.0:4500        *:.*                  *:.*        928
  
```

Ilustración 7: Puerto python ejecutándose

Configurar las reglas de reenvío y filtrado de IP del equipo principal mediante iptables, para habilitar el acceso a internet:

```

# iptables -A FORWARD -o eth0 -i vboxnet0 -s 192.168.56.0/24 -m conntrack --ctstate NEW -j ACCEPT
# iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
# iptables -A POSTROUTING -t nat -j MASQUERADE
# systemctl -w net.ipv4.ip_forward=1
  
```

Se deben colocar después de un reinicio de la maquina principal, debido a que no son iptables persistentes

Crear un snapshot de la maquina víctima, para volver a un estado limpio luego de realizar un análisis. Este proceso se lo puede realizar mediante el terminal por líneas de comando o mediante la interface gráfica, para el caso de la terminal colocar los siguientes comandos:

```
$ vboxmanage snapshot "victimaI" take "cuckooSnap01" --pause
```

```
$ vboxmanage controlvm "victimaI" poweroff
$ vboxmanage snapshot "victimaI" restorecurrent
$ vboxheadless --startvm "victimaI"
```

```
tatty@ubuntu:~$ vboxmanage snapshot victima1 take cuckooSnap01 --pause
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
Snapshot taken. UUID: e9131896-8e9e-4282-a7b9-01de2483e0b2
tatty@ubuntu:~$ vboxmanage controlvm "victima1" poweroff
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
tatty@ubuntu:~$ vboxmanage snapshot "victima1" restorecurrent
Restoring snapshot e9131896-8e9e-4282-a7b9-01de2483e0b2
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
tatty@ubuntu:~$ vboxheadless --startvm "victima1"
Oracle VM VirtualBox Headless Interface 5.1.8
(C) 2008-2016 Oracle Corporation
All rights reserved.
```

Ilustración 8: Snapshot maquina victima

Por la interface gráfica, seguir los siguientes pasos.

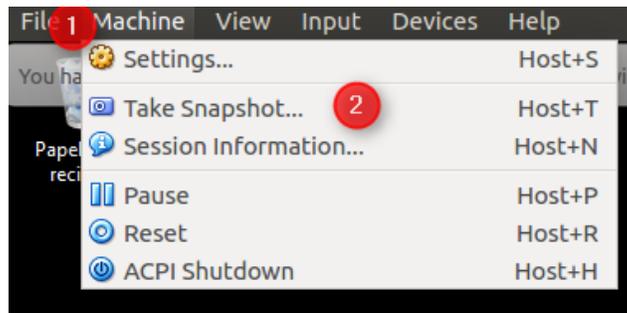


Ilustración 9: snapshot por interface gráfica 1

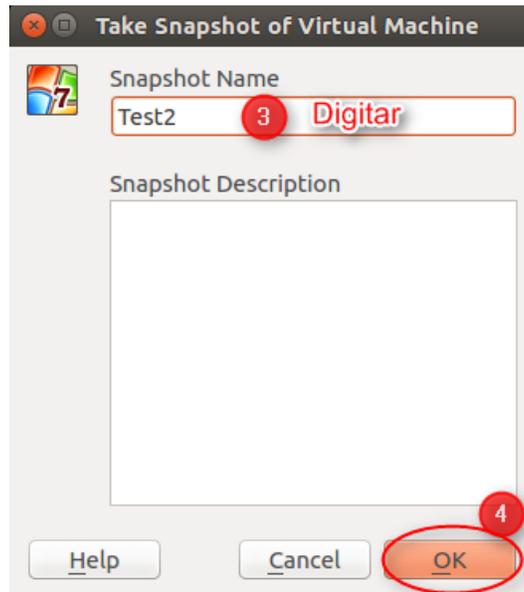


Ilustración 10: snapshot por interface gráfica 2

2.2. Configuración de Cuckoo Sandbox

Es necesario parametrizar los archivos de configuración de Cuckoo, que consisten en los siguientes archivos principales:

cuckoo.conf

Donde se puede pedir a Cuckoo que compruebe la versión más reciente al momento de ejecutarse. Puede describir su método de virtualización. También puede anotar la dirección IP del host y el número de puerto utilizado por Cuckoo Sandbox. Para el caso de nuestro proyecto se ha configurado de la siguiente manera

```
[cuckoo]
version_check = on
delete_original = off
delete_bin_copy = off

machinery = virtualbox

memory_dump = off
terminate_processes = off
reschedule = off
process_results = on

max_analysis_count = 0
max_machines_count = 0
max_vmstartup_count = 10

freespace = 64
tmppath = /tmp
rooter = /tmp/cuckoo-rooter

[routing]
route = none
internet = none
rt_table = main
auto_rt = yes

[resultserver]
ip = 192.168.56.1
port = 2042
force_port = no
upload_max_size = 10485760

[processing]
analysis_size_limit = 104857600
resolve_dns = on
sort_pcap = on

[database]
connection =
timeout =

[timeouts]
default = 120
critical = 60
vm_state = 60
```

Ilustración 11: Configurar archivo cuckoo.conf

virtualbox.conf

Donde se especifica la forma en que se ejecuta VirtualBox y la forma en que interactuará con Cuckoo, ya que puede ser por línea de comandos o con interfaz gráfica de usuario. Puede editar el nombre del sistema operativo invitado en esta configuración. Definir el sistema operativo utilizado en la maquina víctima y la dirección IP del sistema invitado. Para el caso de nuestro proyecto se ha configurado de la siguiente manera

```

[virtualbox]
mode = headless 1
path = /usr/bin/VBoxManage
interface = vboxnet0 2
machines = victima1

[victima1] 3
label = victima1
platform = windows 4
ip = 192.168.56.101 5
# snapshot = Snapshot1
# interface = vboxnet0
# resultserver_ip = 192.168.56.1
# resultserver_port = 2042
# tags = windows_xp_sp3,32_bit,acrobat_reader_6

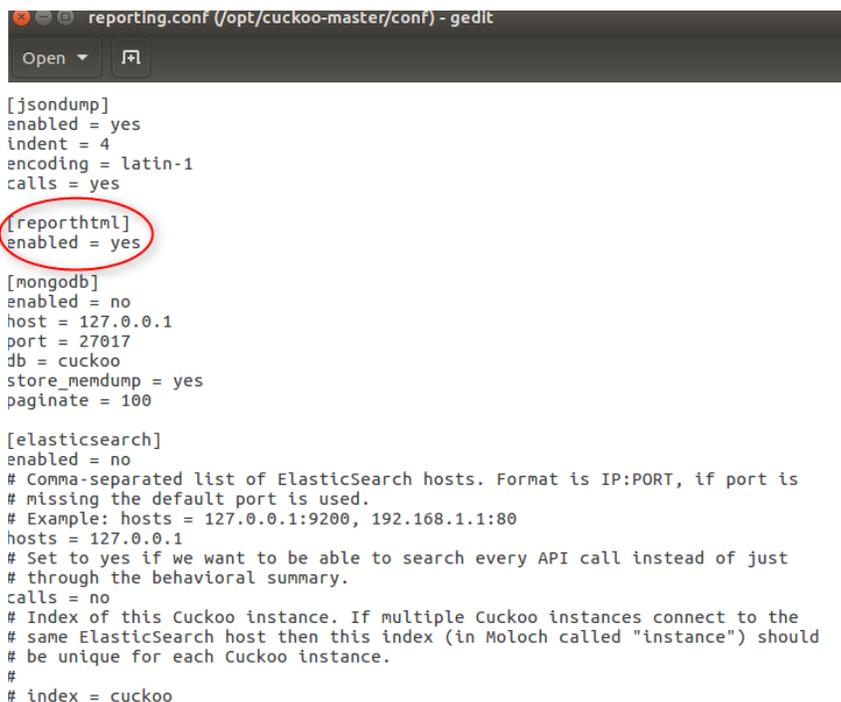
[honeyd]
label = honeyd
platform = linux
ip = 192.168.56.102
tags = service, honeyd
options = nictrace noagent

```

Ilustración 12: Configurar archivo virtualbox.conf

reporting.conf

Adicional configurar el archivo reporting, donde se habilitará el reporte de tipo html.



```

reporting.conf (/opt/cuckoo-master/conf) - gedit
Open [icon]

[jsondump]
enabled = yes
indent = 4
encoding = latin-1
calls = yes

[reporthtml]
enabled = yes

[mongodb]
enabled = no
host = 127.0.0.1
port = 27017
db = cuckoo
store_memdump = yes
paginate = 100

[elasticsearch]
enabled = no
# Comma-separated list of Elasticsearch hosts. Format is IP:PORT, if port is
# missing the default port is used.
# Example: hosts = 127.0.0.1:9200, 192.168.1.1:80
hosts = 127.0.0.1
# Set to yes if we want to be able to search every API call instead of just
# through the behavioral summary.
calls = no
# Index of this Cuckoo instance. If multiple Cuckoo instances connect to the
# same Elasticsearch host then this index (in Moloch called "instance") should
# be unique for each Cuckoo instance.
#
# index = cuckoo

```

Ilustración 13: Configurar archivo reporting.conf

Finalmente se ejecuta Cuckoo para el análisis de malware, con el comando

```
$ python ./cuckoo.py
```

Dentro de la ruta /opt/cuckoo-master

2.3. Colocar muestra de malware

Tener en cuenta las siguientes consideraciones

1. La máquina víctima¹ debe estar apagada, y sacada un snapshot, Cuckoo la encenderá para correr el malware sobre ella.
2. Las muestras de malware fueron tomadas de la consola de GitHub, AMP de Cisco y malc0de, estas muestras se encuentran dentro de la carpeta compartida Documents/Muestras la misma que fue creada durante la instalación y configuración de Cuckoo Sandbox

Podemos colocar las muestras para el análisis por la terminal o por interface web

- Por la terminal dentro de la carpeta `utils` de Cuckoo, se ejecuta en siguiente comando con la ruta de la muestra.

```
$ ./submit.py /home/tatty/Documents/Muestras/[nombre de la muestra]
```

Asegurarse que se obtuvo el resultado de éxito.

```
tatty@ubuntu:/opt/cuckoo-master/utils$ ./submit.py /home/tatty/Documents/Muestras/Samples/2a8820e59a92442347caa7cb4ad35f6e.7z
Success: File "/home/tatty/Documents/Muestras/Samples/2a8820e59a92442347caa7cb4ad35f6e.7z" added as task with ID 1
tatty@ubuntu:/opt/cuckoo-master/utils$
```

Ilustración 15: Colocar muestra de malware por línea de comandos

- Por la interface web, se debe iniciar el servicio web, que se encuentra dentro de `/opt/cuckoo-master/utils` con el siguiente comando

```
$ python web.py
```

```
tatty@ubuntu:/opt/cuckoo-master/utils$ python web.py
Bottle v0.12.7 server starting up (using WSGIRefServer())...
Listening on http://0.0.0.0:8080/
Hit Ctrl-C to quit.
```

Ilustración 16: Iniciar servicio web

Iniciará un servidor web en el localhost usando el puerto 8080. Dentro del navegador colocar `http://localhost:8080`, esto abre un formulario para cargar el malware, especificar algunas opciones y enviar:

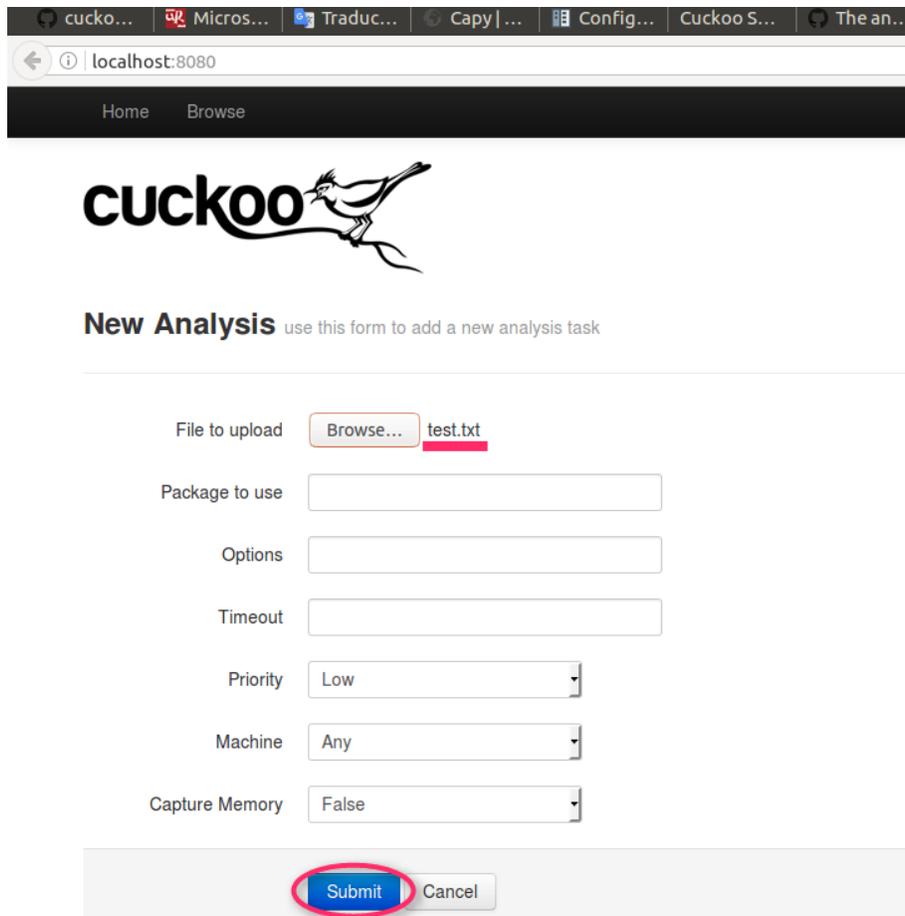


Ilustración 17: Colocar muestra de malware por interface grafica

Asegurarse de obtener el mensaje de éxito.



New Analysis

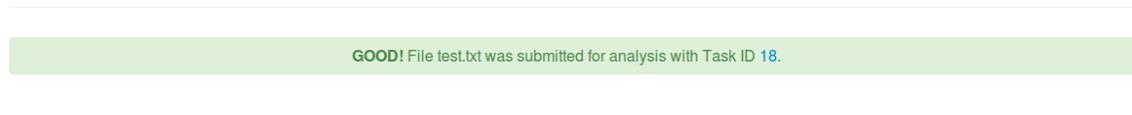


Ilustración 18: Mensaje exitoso

Una vez colocada la muestra de malware, podemos observar que Cuckoo inicia el proceso de análisis de la muestra.

file:///opt/cuckoo-master/storage/analyses/16/reports/report.html#screenshots

cuckoo

Info File Signatures Screenshots Static Dropped Network Behavior Volatility

Category	Started On	Completed On	Duration	Cuckoo Version
FILE	2016-11-29 06:12:33.106461	2016-11-29 06:15:29.695680	176 seconds	2.0-dev

Machine	Label	Manager	Started On	Shutdown On
victima1	victima1	VirtualBox	2016-11-29 06:12:33	2016-11-29 06:15:29

File Details

File name	WGBSetup.exe
File size	1859760 bytes
File type	PE32 executable (GUI) Intel 80386, for MS Windows
CRC32	5371E048
MD5	0ce637ba6fd1241fd715913493be520f
SHA1	2cfe52fa74f2685ba85bab2b237814aa970ee8e
SHA256	ec0d9d94029f5c76b77f33d251c283e8dda29d030252f605e4eed018a86c0b1c

Ilustración 22: Resultado del Análisis de Cuckoo

Antes de iniciar con el análisis, entenderemos la información presentada por Cuckoo

2.4. Información recolectada

La información que arroja Cuckoo de acuerdo a los módulos que tengamos configurados, se detalla a continuación:

Información (Info)

Muestra el tipo de objeto que fue analizado, la fecha en que se inició y terminó dicho análisis, la duración del análisis y la versión de Cuckoo con la que se realizó el análisis. Seguido de la información de la maquina víctima, nombre de la máquina, escenario, hora de encendido de la máquina y hora en la que se apagó.

file:///opt/cuckoo-master/storage/analyses/16/reports/report.html#network

cuckoo

Info File Signatures Screenshots Static Dropped Network Behavior Volatility

Category	Started On	Completed On	Duration	Cuckoo Version
FILE	2016-11-29 06:12:33.106461	2016-11-29 06:15:29.695680	176 seconds	2.0-dev

Machine	Label	Manager	Started On	Shutdown On
victima1	victima1	VirtualBox	2016-11-29 06:12:33	2016-11-29 06:15:29

Ilustración 23: Detalle INFO

Detalles del Archivo (File Details)

Muestra el nombre del archivo analizado, su tamaño en bytes, el tipo de archivo, el CRC32 y diversas funciones hash como MD5, SHA1, SHA256 y SHA512. Además, se muestra un algoritmo de hashing fuzzy (Ssdeep) y la información con PEiD, seguido de información acerca de Yara y finalmente un campo de Virus Total, el cual despliega el número de motores antivirus que detectaron la muestra como una amenaza.

File Details

File name	WGBSetup.exe
File size	1859760 bytes
File type	PE32 executable (GUI) Intel 80386, for MS Windows
CRC32	5371E048
MD5	0ce637ba6fd1241fd715913493be520f
SHA1	2cfe52fa74f2685ba85bab2b237814aa970ee8e
SHA256	ec0d9d94029f5c76b77f33d251c283e8dda29d030252f605e4eed018a86c0b1c
SHA512	32ce41bdbe4717b267f853bab9ae35b3714bc11251049494da0dccc53d5af29babf9c67dade3606caceb8d1e4193b47036dfaf9ff17b3ae87350798fc5d93e6
Ssdeep	49152:i0Tu+P+7hxXB8C7m9hq1uSYoxB32CQBITeZKv4rcVN0:Nxm7hPf7m9T8xBHQIy0v4U6
PEiD	None matched
Yara	None matched
VirusTotal	Permalink VirusTotal Scan Date: 2016-11-22 15:39:59 Detection Rate: 0/56 (Expand)

Ilustración 24: Detalle FILE

Firmas (Signatures)

Contiene las firmas de la Muestra de malware

Signatures

antivm_queries_computername details
antivm_memory_available details
pe_features details
network_http details
allocates_rwx details
antisandbox_foregroundwindows details
antivm_disk_size details
creates_exe details
dropper details
browser_security details
deletes_self details
modifies_files details

Ilustración 25: Detalle SIGNATURES

Capturas de Pantalla (Screenshots)

En esta sección se encuentran las capturas de pantalla que Cuckoo toma mientras se realiza el análisis de la muestra.

Screenshots

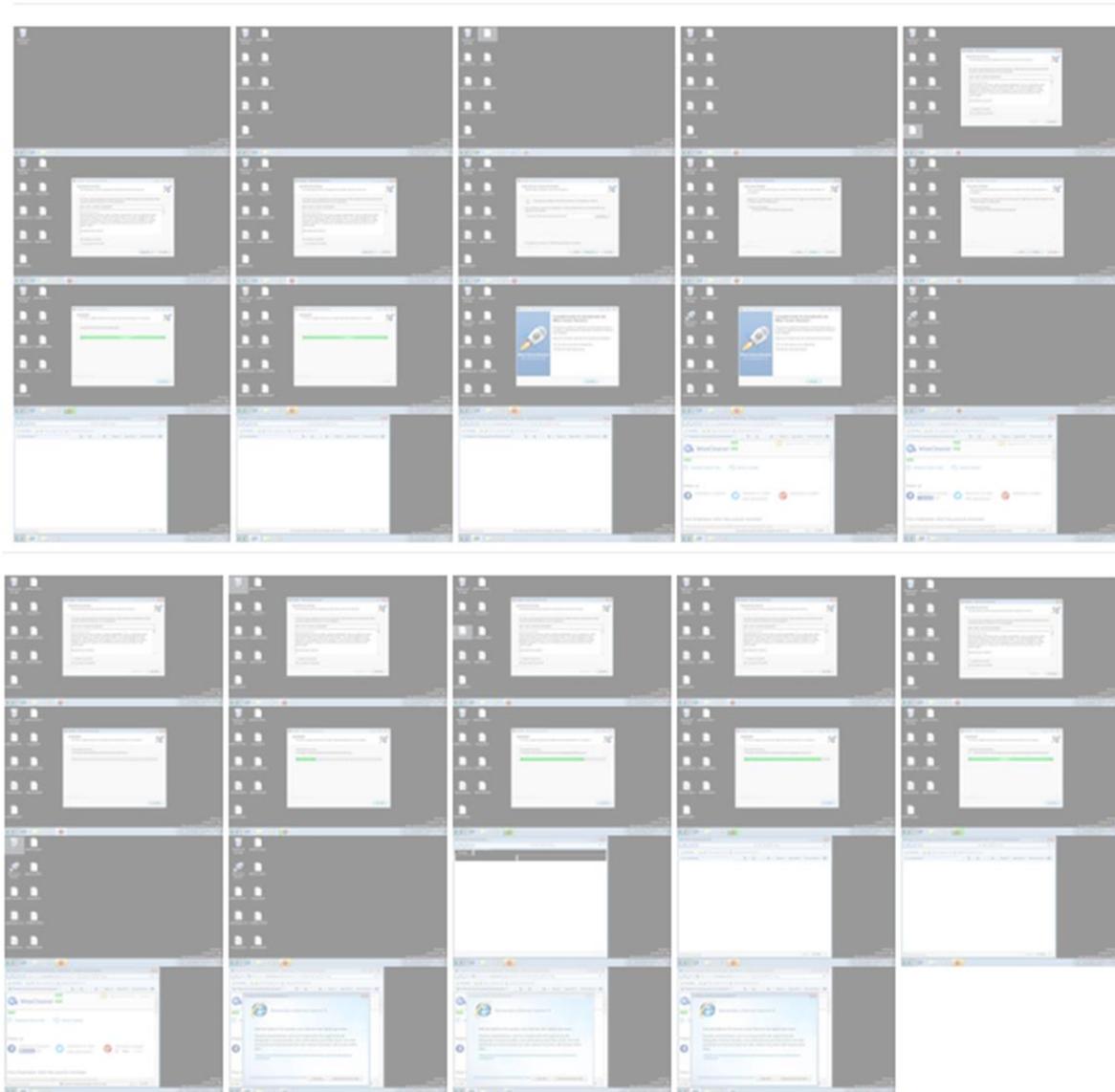


Ilustración 26: Detalle SCREENSHOTS

Análisis Estático (Static Analysis)

Podemos apreciar la información respecto a la estructura del archivo ejecutable. Se puede encontrar información sobre las secciones que conforman el archivo y las direcciones de memoria. Así mismo, es posible determinar si el archivo malicioso importa alguna librería durante su ejecución

Static Analysis

Version Infos

LegalCopyright:	WiseCleaner.com
FileVersion:	1.38
CompanyName:	WiseCleaner.com
Comments:	This installation was built with Inno Setup.
ProductName:	Wise Game Booster
ProductVersion:	1.38
FileDescription:	Wise Game Booster
Translation:	0x0000 0x04b0

Sections

Resources

Imports

Strings

Ilustración 27: Detalla STATIC ANALYSIS

Archivos Eliminados (Dropped Files)

Muestra información sobre los archivos eliminados por el malware y descargados por Cuckoo.

Dropped Files

```
e42dd813bcb43f8c_czech.ini  
567931ee80ea6f30_finnish.ini  
5ebe66c05bcd318a_norwegian(bokmal).ini  
6ac23c246b149d6c_arabic.ini  
b3837ed7d2ee86b2_chinese(traditional).ini  
8523c5d3ecfc56e6_estonian.ini  
e663b8ce44df7a7c_russian.ini  
4389f49870ac58fb_interdiction[1].js  
229fbb19a7f24d8e_djfcqyb.ini  
0eb2a0a6935c3f30_serbian(latin).ini  
4d2b335732bcfd3a_slovak.ini  
3006e231168136c9_indonesian.ini  
e13cd7dca8b7ec1c_korean.ini  
8b2dffe09eb708f_license.txt  
e996c2f1464046fe_kurdish(kurmanci).ini  
f0bb7b22d9e7e856_vietnamese.ini  
170fc43e8c8445bb_countdown[1].js  
a495e1684784f4a2_activity-google-analytics[1].js
```

Ilustración 28: Detalle DROPPED FILES

Análisis de Red (Network Analysis)

Analiza los archivos PCAP y extrae la información de red, como el tráfico de DNS, dominios, direcciones IP, solicitudes HTTP, IRC y tráfico SMTP

Network Analysis

Hosts Involved

IP Address
157.248.0.22
199.96.57.6
204.79.197.200
216.58.219.131
216.58.219.141
216.58.219.142
31.13.73.36
66.148.114.8
66.148.115.210
8.8.8.8

DNS Requests

HTTP Requests

Ilustración 29: Detalle NETWORK ANALYSIS

Resumen del Comportamiento (Behavior Summary)

Se encuentra un historial de aquellos archivos que fueron modificados, el detalle de las exclusiones mutuas (mutexes) como las llaves de registro

Behavior Summary

```
File-Read
• C:\Program Files\Wise\Wise Game Booster\Languages\Italian.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Bulgarian.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Spanish(Spain).ini
• C:\Users\victimal\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content
• C:\Program Files\Wise\Wise Game Booster\Languages\Belarusian.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Catalan.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\English.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Portuguese(Portugal).ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Dutch(Netherlands).ini
• C:\Users\victimal\Videos\desktop.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Polish.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Greek.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Portuguese(Brasil).ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Kurdish(Kurmanci).ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Chinese(Traditional).ini
• C:\Users\victimal\Pictures\desktop.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\French.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Serbian(Latin).ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Vietnamese.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Norwegian(Bokmal).ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Czech.ini
• C:\Program Files\Wise\Wise Game Booster\Languages\Romanian.ini
• C:\Users\victimal\Music\desktop.ini
```

Ilustración 30: Detalle BEHAVIOR SUMMARY

Procesos (Processes)

Se selecciona el nombre del archivo analizado, y desplegará una tabla con el resumen de las operaciones realizadas sobre el sistema. Estas operaciones se encuentran separadas por categorías, y se diferencian por un color distinto

Processes

```
registry filesystem process services network synchronization
Isass.exe PID: 460, Parent PID: 372
WGBSetup.exe PID: 2532, Parent PID: 2504
WGBSetup.tmp PID: 2676, Parent PID: 2532
explorer.exe PID: 1180, Parent PID: 1108
WiseGameBooster.exe PID: 3520, Parent PID: 2676
```

Ilustración 31: Detalle PROCESSES

3. RESULTADOS

La información recolectada por Cuckoo nos ayuda a realizar un análisis dinámico sobre lo que el malware estaba haciendo en el sistema infectado. Desde el principio, cuando el malware fue desplegado en el sistema, qué cambios hizo en el sistema, y así sucesivamente.

Análisis 1: Muestra Wise Game Booster descargada de malc0de [19]

Para el ejemplo, la muestra de malware es un ejecutable llamado WGBSetup.exe

File Details

File name	WGBSetup.exe
File size	1859760 bytes
File type	PE32 executable (GUI) Intel 80386, for MS Windows
CRC32	5371E048
MD5	0ce637ba6fd1241fd715913493be520f
SHA1	2cefe52fa74f2685ba85bab2b237814aa970ee8e
SHA256	ec0d9d94029f5c76b77f33d251c283e8dda29d030252f605e4eed018a86c0b1c
SHA512	32ce41bdbe4717b267f853bab9ae35b3714bc11251049494da0dccc53d5af29babf9c67dade3606caceb8d1e4193b47036dfaf9ff17b3ae87350798fc5d93e6
Ssdeep	49152:i0Tu+P+7hxXB8C7m9hq1uSYoxB32CQBIteZKv4rcVN0:Nxm7hPf7m9T8xBHQIy0v4U6
PEID	None matched
Yara	None matched
VirusTotal	Permalink VirusTotal Scan Date: 2016-11-22 15:39:59 Detection Rate: 0/56 (Expand)

Ilustración 32: Tipo de malware Caso1

Se trata de un juego llamado Wise Game Booster, creado por la compañía WiseCleaner.com

Static Analysis

Version Infos

LegalCopyright:	WiseCleaner.com
FileVersion:	1.38
CompanyName:	WiseCleaner.com Nombre del creador
Comments:	This installation was built with Inno Setup.
ProductName:	Wise Game Booster Juego
ProductVersion:	1.38
FileDescription:	Wise Game Booster
Translation:	0x0000 0x04b0

Sections

Resources

Imports

Strings

Ilustración 33: Nombre del malware y su creador

Se observa que realizan conexiones a sitios externos

Network Analysis

Hosts Involved

IP Address
157.240.0.22
199.96.57.6
204.79.197.200
216.58.219.131
216.58.219.141
216.58.219.142
31.13.73.36
66.148.114.8
66.148.115.210
8.8.8.8

Direcciones a las que se conecta la muestra

DNS Requests

HTTP Requests

Ilustración 34: Conexión sitios externos Caso 1

Las direcciones IP a las que se conectan son google, Facebook, y la página de los creadores del juego



Ilustración 35: Validación de dirección IP

Podemos revisar el tráfico que existió hacia las direcciones IP en el archivo dump.pcap, con ayuda de wireshark.

En la Ilustración 36: Filtro DNS wireshark podemos identificar los servidores a los que se conectó el malware a través de las peticiones DNS

No.	Time	Source	Destination	Protocol	Length	Info
173	109.694402	192.168.56.101	8.8.8.8	DNS	72	Standard query 0xdccb A www.bing.com
174	109.769729	8.8.8.8	192.168.56.101	DNS	164	Standard query response 0xdccb A www.bing.com CNAME ww-bing-...
183	111.790849	192.168.56.101	8.8.8.8	DNS	75	Standard query 0x156f A wisecleaner.com
184	111.959230	8.8.8.8	192.168.56.101	DNS	91	Standard query response 0x156f A wisecleaner.com A 66.148.114...
194	112.636134	192.168.56.101	8.8.8.8	DNS	79	Standard query 0x463f A www.wisecleaner.com
195	112.725228	8.8.8.8	192.168.56.101	DNS	109	Standard query response 0x463f A www.wisecleaner.com CNAME wi...
483	122.247439	192.168.56.101	8.8.8.8	DNS	76	Standard query 0xef0a A www.facebook.com
501	122.324032	8.8.8.8	192.168.56.101	DNS	121	Standard query response 0xef0a A www.facebook.com CNAME star-...
555	122.610859	192.168.56.101	8.8.8.8	DNS	80	Standard query 0x1118 A platform.twitter.com
570	122.640950	192.168.56.101	8.8.8.8	DNS	75	Standard query 0x10d9 A apis.google.com
592	122.698303	8.8.8.8	192.168.56.101	DNS	122	Standard query response 0x1118 A platform.twitter.com CNAME p...
599	122.730876	8.8.8.8	192.168.56.101	DNS	112	Standard query response 0x10d9 A apis.google.com CNAME plus.l...
696	123.182458	192.168.56.101	8.8.8.8	DNS	79	Standard query 0x6c4e A www.wisecleaner.net
703	123.256546	8.8.8.8	192.168.56.101	DNS	109	Standard query response 0x6c4e A www.wisecleaner.net CNAME wi...
796	126.386492	192.168.56.101	8.8.8.8	DNS	79	Standard query 0xb34b A static.xx.fbcdn.net
797	126.461112	8.8.8.8	192.168.56.101	DNS	118	Standard query response 0xb34b A static.xx.fbcdn.net CNAME sc...
925	129.180054	192.168.56.101	8.8.8.8	DNS	79	Standard query 0x5578 A accounts.google.com
1007	130.229577	192.168.56.101	8.8.8.8	DNS	79	Standard query 0x5578 A accounts.google.com
1037	130.398785	8.8.8.8	192.168.56.101	DNS	95	Standard query response 0x5578 A accounts.google.com A 216.58...
1115	131.582560	192.168.56.101	8.8.8.8	DNS	75	Standard query 0xdc17 A ssl.gstatic.com
1120	131.792791	8.8.8.8	192.168.56.101	DNS	91	Standard query response 0xdc17 A ssl.gstatic.com A 216.58.219...
1158	133.828439	192.168.56.101	8.8.8.8	DNS	79	Standard query 0x99d6 A www.wisecleaner.net

▶ Frame 1037: 95 bytes on wire (760 bits), 95 bytes captured (760 bits)
 ▼ Ethernet II, Src: 0a:00:27:00:00:00 (0a:00:27:00:00:00), Dst: CadmusCo_23:64:b1 (08:00:27:23:64:b1)
 ▶ Destination: CadmusCo_23:64:b1 (08:00:27:23:64:b1)
 ▶ Source: 0a:00:27:00:00:00 (0a:00:27:00:00:00)

```

0000 08 00 27 23 64 b1 0a 00 27 00 00 00 08 00 45 00  ..#d... '.....E.
0010 00 51 c1 5c 00 00 7f 11 71 22 08 08 08 08 c0 a8  .Q..... q".....
0020 38 65 00 35 cd c0 00 3d d7 2f 55 78 81 80 00 01  8e.5...=:/Ux....
0030 00 01 00 00 00 00 08 61 63 63 6f 75 6e 74 73 06  .....a counts....
0040 67 6f 6f 67 6c 65 03 63 6f 6d 00 00 01 00 01 c0  google.c om.....
0050 0c 00 01 00 01 00 00 01 2a 00 04 d8 3a db 8d  .....*.....
  
```

Ilustración 36: Filtro DNS wireshark

De igual forma en la Ilustración 37: Filtro para visualizar peticiones wireshark podemos observar las peticiones realizadas por el malware a servidores remotos

No.	Time	Source	Destination	Protocol	Length	Info
285	120.128610	192.168.56.101	66.148.114.8	HTTP	512	GET /templates/images/download_center.png HTTP/1.1
344	121.466585	192.168.56.101	66.148.114.8	HTTP	540	GET /templates/activity/black-friday-2016-2/images/sid...
355	121.643455	192.168.56.101	66.148.114.8	HTTP	498	GET /templates/js/search.js HTTP/1.1
357	121.644792	192.168.56.101	66.148.114.8	HTTP	533	GET /templates/activity/black-friday-2016-2/images/sup...
358	121.644861	192.168.56.101	66.148.114.8	HTTP	513	GET /templates/css/thanks-for-choosing.css HTTP/1.1
431	122.029876	192.168.56.101	66.148.114.8	HTTP	501	GET /templates/images/logo.png HTTP/1.1
459	122.102001	192.168.56.101	66.148.114.8	HTTP	502	GET /templates/images/flags.png HTTP/1.1
464	122.118379	192.168.56.101	66.148.114.8	HTTP	501	GET /templates/images/arrow.png HTTP/1.1
470	122.160813	192.168.56.101	66.148.114.8	HTTP	506	GET /templates/images/gray_arrow.png HTTP/1.1
472	122.162231	192.168.56.101	66.148.114.8	HTTP	510	GET /templates/images/new-subscript.png HTTP/1.1
495	122.284562	192.168.56.101	66.148.114.8	HTTP	510	GET /templates/images/hot-subscript.png HTTP/1.1
518	122.364968	192.168.56.101	66.148.114.8	HTTP	503	GET /templates/images/search.png HTTP/1.1
524	122.440970	192.168.56.101	66.148.114.8	HTTP	521	GET /templates/images/thanks-for-choosing/icon.png HTT...
525	122.441014	192.168.56.101	31.13.73.36	HTTP	704	GET /plugins/like.php?href=http%3A%2F%2Fwww.facebook.c...
536	122.507855	192.168.56.101	66.148.114.8	HTTP	507	GET /images/products_icon/wdc-60.png HTTP/1.1
540	122.510278	192.168.56.101	66.148.114.8	HTTP	507	GET /images/products_icon/365-60.png HTTP/1.1
544	122.517074	192.168.56.101	66.148.114.8	HTTP	507	GET /images/products_icon/wrc-60.png HTTP/1.1
567	122.630439	192.168.56.101	66.148.114.8	HTTP	506	GET /images/products_icon/wu-60.png HTTP/1.1
596	122.713663	192.168.56.101	66.148.114.8	HTTP	508	GET /images/products_icon/wpc-60.png HTTP/1.1
608	122.798662	192.168.56.101	199.96.57.6	HTTP	435	GET /widgets.js HTTP/1.1
609	122.798699	192.168.56.101	66.148.114.8	HTTP	503	GET /wisecleaner_analytics/wa.js HTTP/1.1
623	122.840360	192.168.56.101	66.148.114.8	HTTP	507	GET /templates/images/gray-arrow.png HTTP/1.1

▶ Frame 357: 533 bytes on wire (4264 bits), 533 bytes captured (4264 bits)
 ▼ Ethernet II, Src: CadmusCo_23:64:b1 (08:00:27:23:64:b1), Dst: 0a:00:27:00:00:00 (0a:00:27:00:00:00)
 ▶ Destination: 0a:00:27:00:00:00 (0a:00:27:00:00:00)
 ▶ Source: CadmusCo_23:64:b1 (08:00:27:23:64:b1)

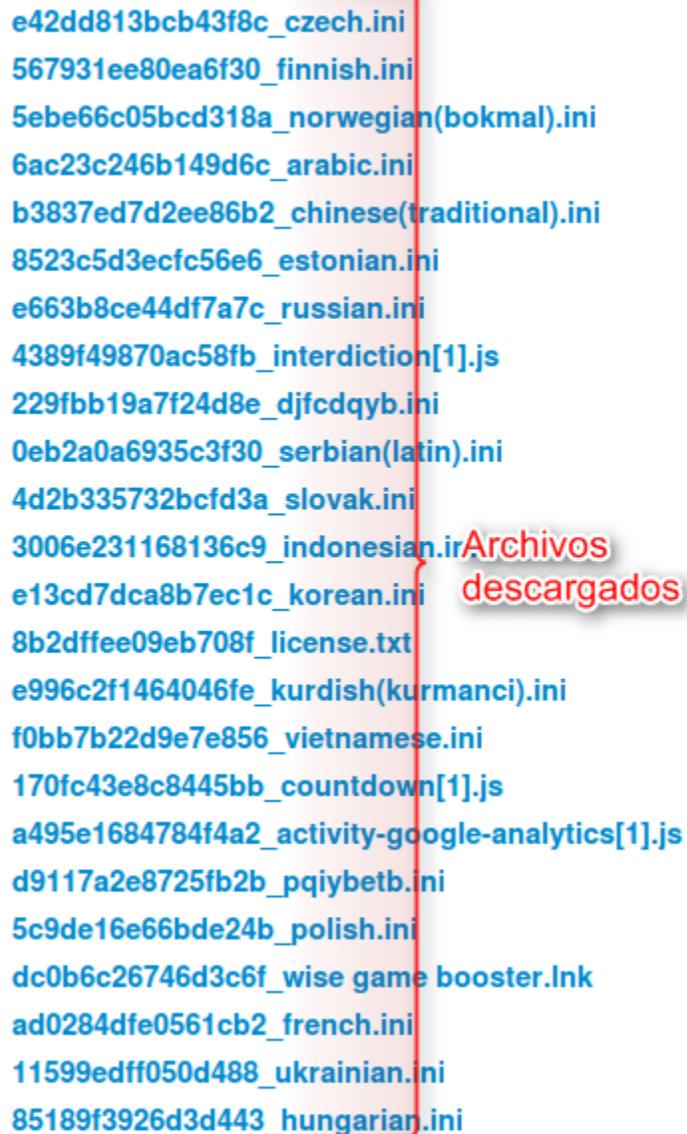
```

0000 0a 00 27 00 00 00 08 00 27 23 64 b1 08 00 45 00  ..'.....'#d...E.
0010 02 07 4d b4 40 00 80 06 fd 92 c0 a8 38 65 42 94  ..M@... ..8eB.
0020 72 08 c0 3e 00 50 0c ea 90 3b 0e 7e 48 c1 50 18  r...>.P. ;.-H.P.
0030 fa f0 c9 d5 00 00 47 45 54 20 2f 74 65 6d 70 6c  .....GE T /templ
0040 61 74 65 73 2f 61 63 74 69 76 69 74 79 2f 62 6c  ates/act ivity/bl
0050 61 63 6b 2d 66 72 69 64 61 79 2d 32 30 31 36 2d  ack-frid ay-2016-
  
```

Ilustración 37: Filtro para visualizar peticiones wireshark

Continuando con el reporte presentado por Cuckoo, vemos que el malware a descargado algunos archivos durante su ejecución.

Dropped Files



A screenshot of a list of files dropped by malware. The list is titled 'Dropped Files' and contains 20 entries. A red vertical box highlights the entire list. To the right of the list, the text 'Archivos descargados' is written in red with a white shadow.

- e42dd813bcb43f8c_czech.ini
- 567931ee80ea6f30_finnish.ini
- 5ebe66c05bcd318a_norwegian(bokmal).ini
- 6ac23c246b149d6c_arabic.ini
- b3837ed7d2ee86b2_chinese(traditional).ini
- 8523c5d3ecfc56e6_estonian.ini
- e663b8ce44df7a7c_russian.ini
- 4389f49870ac58fb_interdiction[1].js
- 229fbb19a7f24d8e_djfcqyb.ini
- 0eb2a0a6935c3f30_serbian(latin).ini
- 4d2b335732bcfd3a_slovak.ini
- 3006e231168136c9_indonesian.ir
- e13cd7dca8b7ec1c_korean.ini
- 8b2dffee09eb708f_license.txt
- e996c2f1464046fe_kurdish(kurmanci).ini
- f0bb7b22d9e7e856_vietnamese.ini
- 170fc43e8c8445bb_countdown[1].js
- a495e1684784f4a2_activity-google-analytics[1].js
- d9117a2e8725fb2b_pqiybetb.ini
- 5c9de16e66bde24b_polish.ini
- dc0b6c26746d3c6f_wise game booster.lnk
- ad0284dfe0561cb2_french.ini
- 11599edff050d488_ukrainian.ini
- 85189f3926d3d443_hungarian.ini

Ilustración 38: Archivos descargados

Las capturas de pantalla que toma Cuckoo, muestran la instalación de un juego llamado Wise Game Booster,

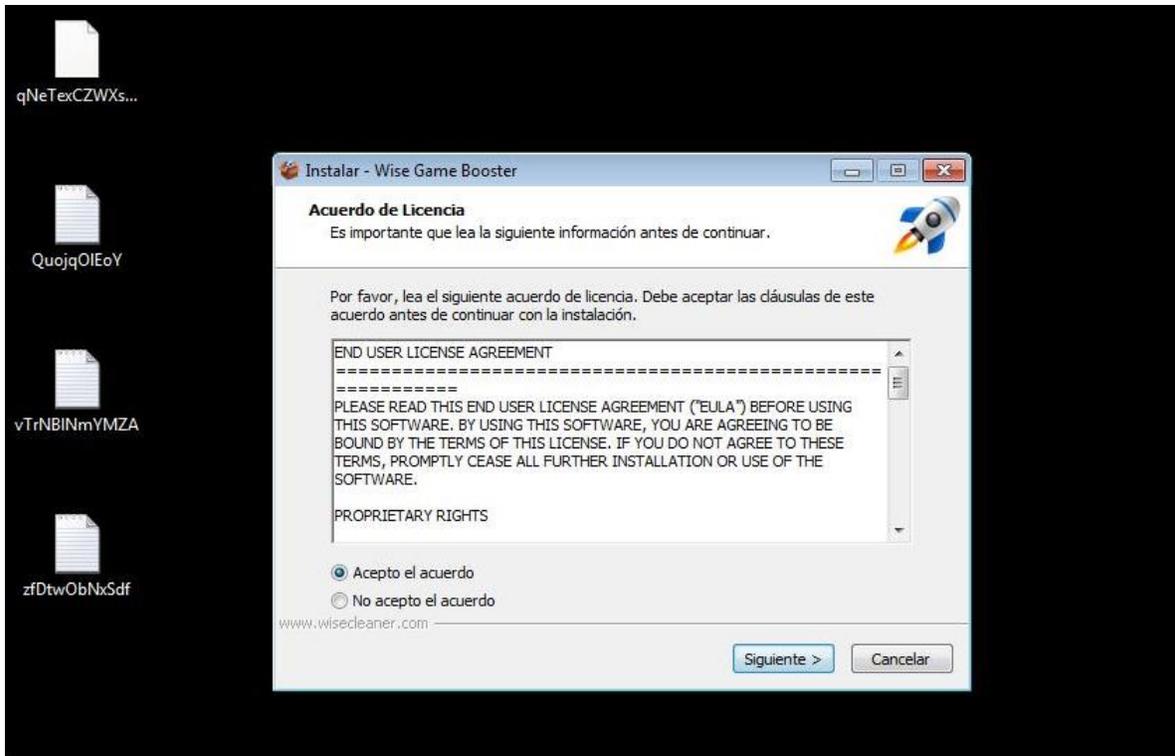


Ilustración 39: Instalación Wise Game Booster 1

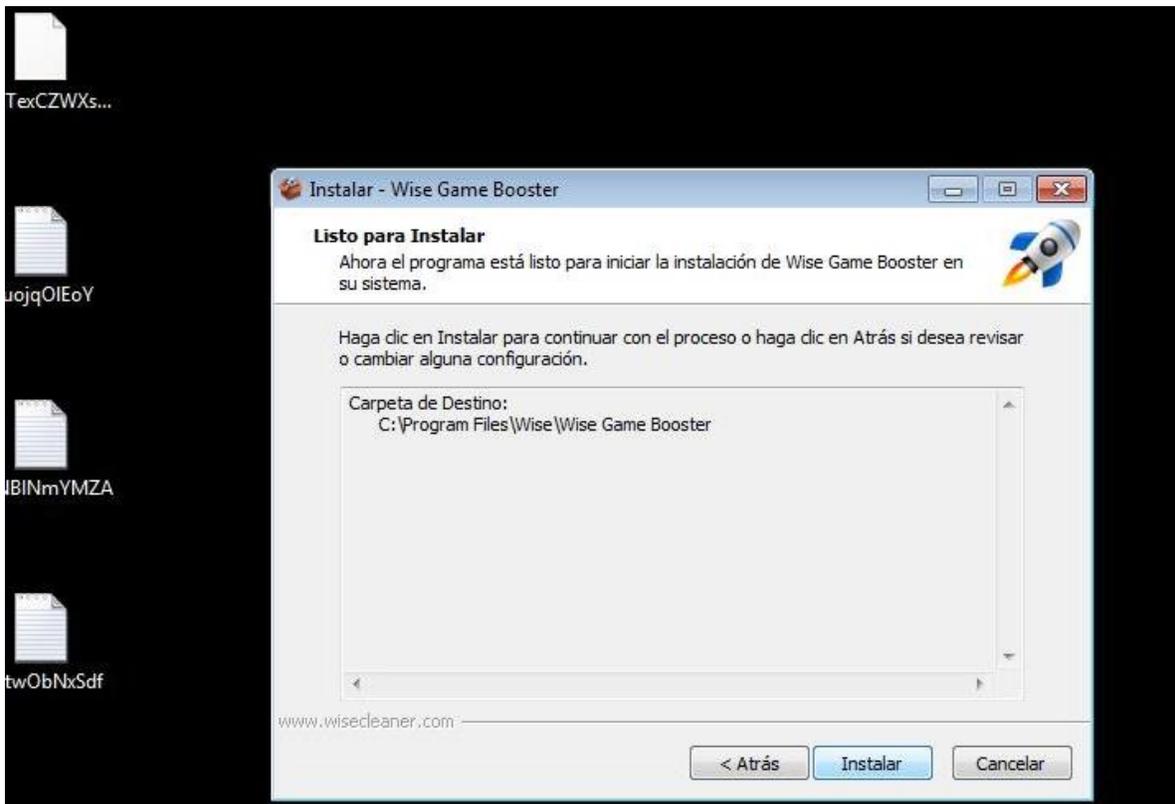


Ilustración 40: Instalación Wise Game Booster 2

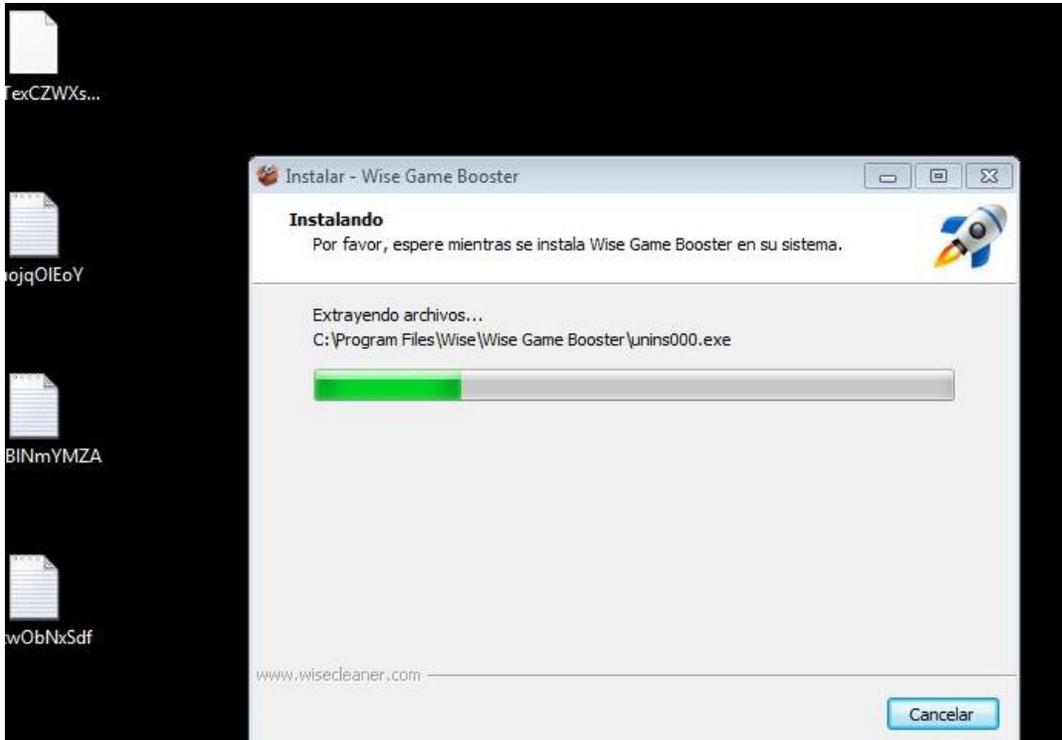


Ilustración 41: Instalación Wise Game Booster 3



Ilustración 42: Instalación Wise Game Booster 4

Dentro del análisis podemos observar que ha realizado la creación y alteración de archivos en el sistema

File-Deleted

- C:\Users\Victima\AppData\Local\Temp\dfjcdqyb.ini
- C:\Users\Victima\AppData\Local\Temp\is-C3IEA.tmp\WGBSetup.tmp
- C:\Users\Public\Desktop\Wise Game Booster.pif
- C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Wise Game Booster\Wise Game Booster.url
- C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Wise Game Booster\Wise Game Booster.lnk
- C:\Users\Public\Desktop\Wise Game Booster.url
- C:\Users\Victima\AppData\Local\Temp\is-F5R33.tmp\introduce.lnk
- C:\Users\Public\Desktop\Wise Game Booster.lnk
- C:\Users\Victima\AppData\Local\Temp\is-F5R33.tmp\introduce.url
- C:\Users\Victima\AppData\Local\Temp\is-F5R33.tmp\introduce.pif
- C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Wise Game Booster\Wise Game Booster.pif

File-Opened

- C:\Program Files\Wise\Wise Game Booster\Languages\Italian.ini
- C:\Program Files\Wise\Wise Game Booster\Languages\Bulgarian.ini
- C:\Program Files\Wise\Wise Game Booster\Languages\Spanish(Spain).ini
- C:\Users\Victima\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\DBSAY0P
- C:\
- C:\Program Files\Wise\Wise Game Booster\Languages\Belarusian.ini
- C:\Program Files\Wise\Wise Game Booster\Languages\Catalan.ini
- C:\Program Files\Wise\Wise Game Booster\Languages\English.ini

Lee, escribe, elimina, abre y mueve archivos

Ilustración 43: Modifica archivos del sistema

Realiza creación y alteración de llaves en el registro. Las llaves más aprovechadas por el malware es la Run, cuyo objetivo es lograr el inicio del malware al arrancar el sistema

Registry Key-Deleted

- HKEY_CURRENT_USER\Software\Microsoft\RestartManager\Session0000\SessionHash
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Uninstall\Wise Game Booster_is1
- HKEY_CURRENT_USER\Software\Microsoft\RestartManager\Session0000\Sequence
- HKEY_CURRENT_USER\Software\Microsoft\RestartManager\Session0000
- HKEY_CURRENT_USER\Software\Microsoft\RestartManager\Session0000\Owner
- HKEY_CURRENT_USER\Software\Microsoft\RestartManager\Session0000\RegFilesHash
- HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Uninstall\Wise Game Booster_is1
- HKEY_CURRENT_USER\Software\Microsoft\RestartManager\Session0000\RegFiles0000

Registry Key-Read

- HKEY_CURRENT_USER\Keyboard Layout\Toggle\Language Hotkey
- HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\SmartDithering
- HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{871C5380-42A0-1069-A2EA-08002B30309D}\ShellFolder\CallForAttributes
- HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{7007ACC7-3202-11D1-AAD2-00805FC1270E}\ShellFolder\HideOnDesktopPerL
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Zones\0\Icon
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Zones\1\Icon
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Rpc\MaxRpcSize
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\Print_Background
- HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\BrowserEmulation\IntranetCompatibilityMode
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_XSSFILTER*
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\NonEnum\{7007ACC7-3202-11D1-AAD2-00805FC
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\SendTimeOut
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Zones\1\CurrentLevel
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_PROTOCOL_LOCKDOWN*

Abre, elimina, lee y escribe llaves en el registro

Ilustración 44: Modifica llaves en el registro

Análisis 2: Muestra IoT descargada de GitHub [20]

Una vez colocada la muestra en nuestro entorno de pruebas, y finalizado su análisis contamos con los siguientes resultados

Es un archivo zip con el nombre de IoT.Mirai.zip

File Details

File name	IoT.Mirai.zip
File size	99484 bytes
File type	Zip archive data, at least v1.0 to extract
CRC32	D2F4DFB9
MD5	8597de8b76dc09045ce6e7a6415e4952
SHA1	f9d5fd184d15a1d61e7c3a13f3773cb9dcceac0f
SHA256	d59212d8eec533ecfebafda9f3b129799f8b56ea5fc11c54005bc4f359c7d2cf
SHA512	cf8bb2dad0d5f4bb3d88dc41a06adfa82b7e29ec6e56108625bb630dee971b0530984b84ec1fde6a71bd3c3997baef7cbc93dd0c2e0b39154a294d956dfb696
Ssdeep	1536:JnRFt0C0Bld71gj9gXkMQXAKDbMLBd4kkRjLC4CcwBP6g4GdSE2lh8Lqy40g7:x8TDgBhAKDIz1kRC4KSgx2lh0qy4F7
PEID	None matched
Yara	None matched
VirusTotal	Permalink VirusTotal Scan Date: 2016-11-03 05:10:58 Detection Rate: 1/55 (Expand)

Ilustración 45: Tipo de malware Caso 2

Dentro de los antivirus que lo ven como malware esta NANO-Antivirus, que lo clasifica como un trojano

Antivirus	Version	Result
Ad-Aware	3.0.3.794	Clean
AegisLab	4.2	Clean
AhniLab-V3	3.8.1.10943	Clean
Alibaba	1.0	Clean
ALYac	1.0.1.9	Clean
Antiy-AVL	1.0.0.1	Clean
Arcabit	1.0.0.788	Clean
Avast	8.0.1489.320	Clean
AVG	16.0.0.4064	Clean
Avira	8.3.3.4	Clean
AVware	1.3.0.42	Clean
Comodo	26050	Clean
Cyren	5.4.16.7	Clean
DWeb	7.0.23.8290	Clean
Emsisoft	3.5.0.658	Clean
ESET-NOD32	14380	Clean
F-Prot	4.7.1.166	Clean
F-Secure	11.0.19100.45	Clean
Fortinet	5.4.233.0	Clean
GData	25	Clean
Ikarus	73.2.1.16.0	Clean
Jiangmin	16.0.100	Clean
K7AntiVirus	9.244.21389	Clean
K7GW	9.244.21390	Clean
Kaspersky	15.0.1.13	Clean
Kingsoft	2013.9.14.323	Clean
NANO-Antivirus	1.0.46.12879	Trojan.Mirai.ehgpyrn
NProtect	2016-11-01-01	Clean
Panda	4.6.4.2	Clean

Ilustración 46: Informe VirusTotal

Se puede apreciar que la muestra intenta conectarse hacia direcciones IP externas

Network Analysis

Hosts Involved

IP Address
168.143.241.146
8.8.8.8

DNS Requests
HTTP Requests
ICMP requests

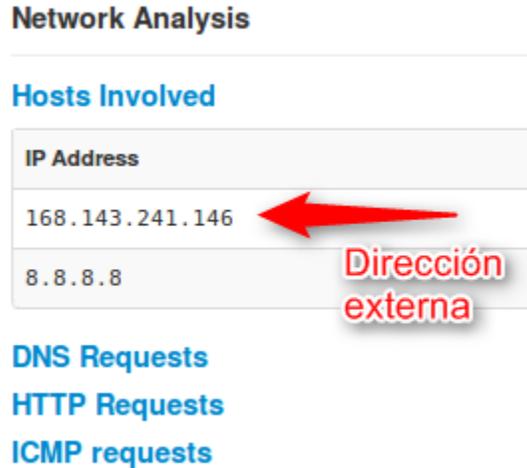


Ilustración 47: Conexión Sitios externos Caso2

Hace una petición al sitio web que muestra la Ilustración 48: Solicitud del malware Caso2

HTTP Requests

URL	Data
http://www.msftncsi.com/ncsi.txt	GET /ncsi.txt HTTP/1.1 Connection: Close User-Agent: Microsoft NCSI Host: www.msftncsi.com

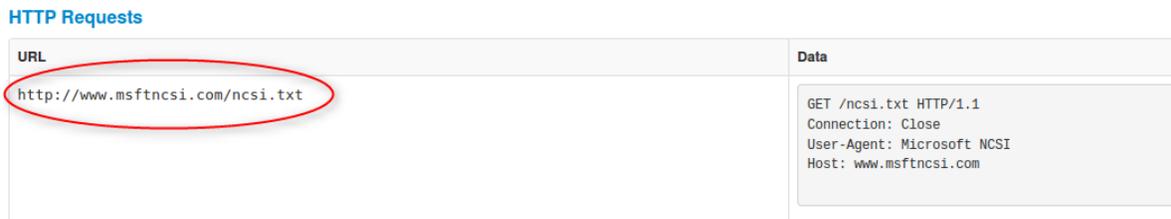


Ilustración 48: Solicitud del malware Caso2

NOTA: Debido a que la muestra es un bootnet, no realiza ninguna acción sin que se conecte al servidor que la creo, actualmente el servidor central de la bootnet se encuentra fuera de servicio.

4. POLÍTICAS DE PREVENCIÓN, REACCIÓN Y MITIGACIÓN DE MALWARE

Las políticas de seguridad de datos se desarrollaron para fomentar y mejorar la seguridad de los datos y facilitar la adopción de medidas de seguridad uniformes a nivel mundial.



Ilustración 49: Cumplimiento de Políticas [21]

4.1. Prevención ante ataque de malware

Los expertos en seguridad informática coinciden en que se debe tener en cuenta los siguientes aspectos para mantenerse prevenido ante un ataque de malware

Deben estar al tanto de lo que hay en ella (dispositivos, sistemas operativos, servicios, aplicaciones, usuarios y demás) [22]. Identificar cuáles son las infraestructuras críticas, la información sensible y las vulnerabilidades, y con base en eso tomar las medidas adecuadas.

Adicionalmente, deben implementar controles de acceso, reforzar políticas de seguridad, y bloquear el acceso a los activos críticos [22] [21]. La PCI DSS proporciona una referencia de requisitos técnicos y operativos desarrollados para proteger los datos. Dentro de los requisitos para cumplimiento de PCI DSS se señala lo siguiente:

Requisito 1: Instale y mantenga una configuración de firewall para proteger los datos [23].

Requisito 5: Proteger todos los sistemas contra malware y actualizar los programas o software antivirus regularmente [23].

Otro de los requisitos a tomar en cuenta como política de seguridad ante la prevención de malware es el requisito 2 de PCI DSS.

Requisito 2: No usar los valores predeterminados suministrados por el proveedor para las contraseñas del sistema y otros parámetros de seguridad [23].

Para finalizar se debe tener en cuenta el siguiente requisito de PCI DSS

Requisito 11: Probar periódicamente los sistemas y procesos de seguridad [23]

4.2. Reacción ante ataque de malware

Durante el ataque de malware se debe seguir actividades que contribuyan a una efectiva respuesta y rápida recuperación, tener en consideración los siguientes aspectos y recomendaciones

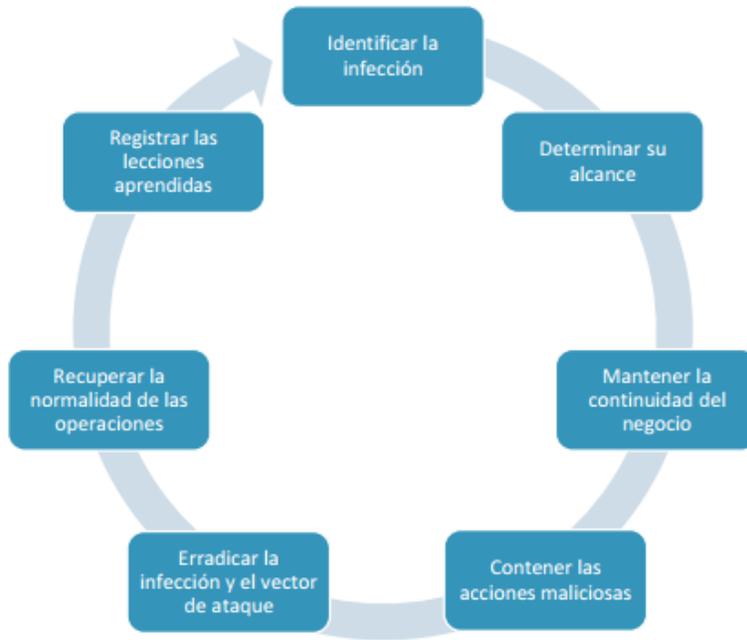


Ilustración 50: Fases para atender un incidente de infección por malware [24]

Identificar la infección: utilizar herramientas de detección automatizadas (antivirus, programas dedicados), la detección es el paso inicial para atender un incidente por malware

Determinar el alcance de la infección: determinar la cantidad de sistemas que han sido comprometidos y de qué manera, para estimar las consecuencias negativas.

Mantener la continuidad del negocio: es fundamental mantener la continuidad de las operaciones críticas de las organizaciones.

Contener las acciones maliciosas: A partir del comportamiento del código malicioso se pueden determinar los pasos a seguir para la contención (aislamiento, suspensión de los segmentos de red, entre otras actividades)

Erradicar la infección y eliminar el vector de ataque: aislar la falla que les permitió el ingreso, para luego eliminarla del sistema.

Recuperar la normalidad en las operaciones: confirmando que los sistemas se encuentran funcionando de manera normal.

Registrar las lecciones aprendidas: mejorando las medidas de seguridad y los procesos de atención de incidentes.

4.3. Mitigación ante ataque de malware

Actualmente las técnicas de propagación e infección del malware son muy avanzadas y puede eludir los controles de IPD/IDS, Antivirus, Honey P0d, etc, y en especial cuando el eslabón más débil de la cadena de seguridad es el usuario, y este es propenso a ingeniería social, uso de dispositivos móviles, Usb, etc.

Es por eso que es difícil determinar el estado de un ataque por lo que Cisco [25] recomienda un análisis continuo de los archivos y el tráfico de la red en busca de amenazas.

Estos análisis se los debe realizar con un software especializado como Kaspersky, ESET, etc.

Otra de las medidas que recomendamos es la aplicación de políticas mínimo privilegio a los usuarios, es decir, que en la red los usuarios por aplicación de políticas únicamente cuenten con permisos de lectura.

Asociados a este tipo de permisos tenemos:

- Restringir el acceso a los programas y archivos.

- Asegurar que se estén utilizados los datos, archivos y programas autorizados por el departamento de sistemas.

- Política de actualización frecuente de los parches de seguridad de sistemas operativos y aplicaciones.

- Restringir el acceso a los puertos USB.

- No uso de Crack de activación de sistemas operativos o aplicaciones.

De la misma forma en el requisito 11 de PCI DSS [23] indica que los componentes, procesos y software personalizado del sistema se deben probar con frecuencia para garantizar que los controles de seguridad continúen reflejando un entorno dinámico.

CONCLUSIONES

El análisis dinámico de Malware en ambientes virtualizados permite conocer de manera rápida y efectiva que acciones realiza el malware dentro de nuestro sistema. De esta forma se puede obtener información acerca de los archivos creados, conexiones de red, modificaciones en el registro, etc.

Las muestras de malware se comportan de manera diferente dependiendo del sistema operativo y el entorno en el que se está ejecutando, esto se debe a que la arquitectura de los sistemas operativos es diferente.

Los sistemas operativos más atacados según nuestras investigaciones es Windows en todas sus versiones, Android ocupa el segundo lugar.

Los resultados de los análisis el Cuckoo Sandbox no son definitivos, se requiere contrastar estos con otras herramientas, para evitar los falsos positivos. Algunas ocasiones podríamos encontrarnos con códigos maliciosos que traten de protegerse del análisis en entornos virtualizados, por lo cual es necesario aplicar algunas técnicas que nos permitan realizar el análisis de malware que evita la virtualización.

La implementación de varios perímetros y controles de seguridad no garantizan la seguridad de la información, sin embargo, la aplicación de políticas permite mitigar o reducir las superficies de ataque, y el impacto del malware dentro de una red.

Cuckoo Sandbox se convierte en una herramienta que apoya la gestión de la seguridad de la información, es importante remarcar que un enfoque dinámico para el análisis de malware permite conocer de manera rápida cuáles son las acciones que este realiza en el sistema. Una vez que se conoce esta información se puede elegir cuál es el curso siguiente a tomar en la investigación.

RECOMENDACIONES

Es necesario el uso de un laboratorio virtualizados y controlado en el manejo de las muestras de malware, debido a que permite tener el control de las acciones realizadas por el malware, pudiendo realizar snapshot de las máquinas para regresar a un punto en el que las máquinas se encuentran limpias.

Los usuarios que utilizamos cualquier equipo electrónico, debemos tener conciencia en las páginas en las que navegamos, lo que descargamos, lo que instalamos, revisar los certificados firmas digitales, que las fuentes sean oficiales, no conectar cualquier equipo extraíble a nuestra máquina.

Referencias

- [1] S. Khandelwal, «The hacker news,» 20 Mayo 2016. [En línea]. Available: <http://thehackernews.com/2016/05/swift-banking-hack.html>.
- [2] Staff tpx, «TPX,» octubre 2016. [En línea]. Available: <http://tpx.mx/blog/2016/iot-la-herramienta-de-ataques-distribuidos-mas-grande-conocida.html>.
- [3] S. Cobb, «WeLiveSecurity,» 26 octubre 2016. [En línea]. Available: <http://www.welivesecurity.com/la-es/2016/10/26/ataques-ddos-a-la-iot-octubre/>.
- [4] AVTEST, «AVTest,» 25 Agosto 2016. [En línea]. Available: <https://www.av-test.org/en/statistics/malware/>.
- [5] M. Rivero, «infospyware,» 01 octubre 2016. [En línea]. Available: <https://www.infospyware.com/articulos/que-son-los-malwares/>.
- [6] Kaspersky Lab, «SecureList,» 2015. [En línea]. Available: <https://securelist.lat/threats/historia-de-los-programas-maliciosos/>.
- [7] Laboratorio de ESET Latinoamerica, «ESET,» Febrero 2012. [En línea]. Available: http://www.eset-la.com/pdf/prensa/informe/cronologia_virus_informaticos.pdf.
- [8] Kaspersky Lab, «Seguridad 101: Los tipos de malware,» 2016. [En línea]. Available: <http://support.kaspersky.com/sp/viruses/general/614>.
- [9] ESET, «Definición de virus, códigos maliciosos y ataques remotos,» 9 septiembre 2016. [En línea]. Available: http://soporte.eset-la.com/kb186/?locale=es_ES.
- [10] O. S. Adebayo, «Techniques for Analysing Android Malware,» *bibliotecavirtual*, p. 7, 2016.
- [11] S. BORTNIK, «welivesecurity,» 2009. [En línea]. Available: <http://www.welivesecurity.com/la-es/2009/06/23/propagacion-malware-correo-electronico/>.
- [12] P. Ramos, «welivesecurity,» 02 noviembre 2011. [En línea]. Available: <http://www.welivesecurity.com/la-es/2011/11/02/el-ciclo-de-un-ataque-de-malware/>.
- [13] C. Gutiérrez, «welivesecurity,» 07 enero 2014. [En línea]. Available: www.welivesecurity.com/la-es/2014/01/07/analizando-muestras-cuckoo-entendiendo-reportes-3/.
- [14] D. Oktavianto y I. Muhandianto, «Getting Started with Automated Malware Analysis using Cuckoo Sandbox,» de *Cuckoo Malware Analysis*, 2013.
- [15] Cuckoo Foundation, «Cuckoo Sandbox,» 2015. [En línea]. Available: <http://docs.cuckoosandbox.org/en/latest/introduction/what/>.
- [16] Github, «github,» 2016. [En línea]. Available: <https://github.com/cuckoosandbox/cuckoo>.

- [17] Hard2bit Dr., «Análisis de malware, enfoque y caso práctico,» 10 Septiembre 2013. [En línea]. Available: <https://hard2bit.com/blog/analisis-de-malware-enfoque-y-caso-practico/>.
- [18] netmarketshare, «netmarketshare,» Diciembre 2016. [En línea]. Available: <https://www.netmarketshare.com/>.
- [19] malc0de, «malc0de,» 2017. [En línea]. Available: <http://malc0de.com/database/index.php?search=0ce637ba6fd1241fd715913493be520f>.
- [20] github, «github,» 2016. [En línea]. Available: <https://github.com/ytisf/theZoo/blob/master/malwares/Source/Original/loT.Mirai/loT.Mirai.zip>.
- [21] Cisco, «Abordar toda la continuidad del ataque,» 2014. [En línea]. Available: https://www.cisco.com/c/dam/r/es/la/internet-of-everything-ioe/assets/pdfs/sec_bda_wp_cte_pte_etmg_es-xl_39724.pdf.
- [22] S. PAGNOTTA, «Welivesecurity,» 28 abril 2014. [En línea]. Available: <https://www.welivesecurity.com/la-es/2014/04/28/grandes-companias-alojan-malware-segun-cisco/>.
- [23] PCI Security Standards Council, «PCI Security Standards Council,» Noviembre 2013. [En línea]. Available: https://es.pcisecuritystandards.org/_onelink_/pcisecurity/en2es/minisite/en/docs/PCI_DSS_v3.pdf.
- [24] ESET, «welivesecurity,» 2015. [En línea]. Available: https://www.welivesecurity.com/wp-content/uploads/2015/11/Guia_respuesta_infeccion_malware_ESET.pdf.
- [25] Cisco, «Protección frente a malware avanzado,» [En línea]. Available: http://www.cisco.com/c/es_es/solutions/enterprise-networks/advanced-malware-protection/index.html.
- [26] R. V. Cadena, «3C TIC,» 2016. [En línea]. Available: <http://www.3ciencias.com/wp-content/uploads/2016/06/art%C3%ADculo1.pdf>.