UNIVERSIDAD DON BOSCO
VICERRECTORÍA ACADÉMICA
FACULTAD DE INGENIERÍA

TRABAJO DE GRADUACIÓN PARA OPTAR AL GRADO DE

Maestro(a) en Seguridad y Gestión de Riesgos Informáticos

PROYECTO

Desarrollo de una Metodología Avanzada para el Análisis de Malware en Entornos Controlados

PRESENTADO POR

Ing. Luis Osmin Hernández Martínez

Ing. Edwin Jonathan González Mejía

ASESOR

Mg. Salvador Alcides Franco Sánchez

Antiguo Cuscatlán, La Libertad, El Salvador, Centro América

Junio 2024

Índice de contenido

Índice de tablas

Índice de imágenes

1. Introducción

En el panorama actual de la seguridad informática, el análisis dinámico de malware surge como una herramienta fundamental para enfrentar la creciente sofisticación y diversidad de amenazas cibernéticas. Desde sus primeros días, este ha evolucionado constantemente, presentando nuevos desafíos tanto para usuarios como para instituciones públicas y corporativas. El surgimiento de amenazas como el ransomware[1] y los cryptominers[2] ha resaltado la urgente necesidad de detectar y prevenir ataques maliciosos de manera proactiva.

Este documento aborda la importancia del análisis dinámico de malware en el contexto actual, destacando su papel crucial en la identificación temprana y mitigación de amenazas. Se explora cómo el análisis estático y dinámico, proporciona una mayor comprensión su comportamiento, permitiendo una mejor detección y respuesta ante nuevas y desconocidas variantes de amenazas.

Además, se examinan las herramientas desarrolladas para analizar ejecutables desconocidos, subrayando su capacidad para reducir la carga de trabajo de los analistas humanos y mejorar la eficiencia en la identificación de amenazas. Se exploran las tendencias actuales en el análisis de malware, incluyendo el impacto de nuevos tipos, técnicas de evasión y el papel de la inteligencia artificial en la mejora de las capacidades de detección.

En última instancia, este documento busca proporcionar una visión integral y actualizada del análisis dinámico, con el objetivo de ayudar a los profesionales de la seguridad informática con los conocimientos, herramientas necesarias y guías de buenas prácticas para hacer el análisis seguro de malware y así hacer frente a las cambiantes amenazas cibernéticas en el mundo actual.

2. Marco teórico

El análisis de malware en entornos controlados es una práctica esencial en la ciberseguridad moderna, permitiendo a los investigadores comprender la naturaleza y el comportamiento de las amenazas digitales sin exponer sistemas reales a riesgos. El desarrollo de una metodología

[1] El **ransomware** es un malware capas de bloquear el acceso a archivos y sistemas, exigiendo un rescate a cambio de restaurar el acceso (NIST)
[2] Los **cryptominers** maliciosos aprovechan los recursos informáticos de las víctimas para llevar a cabo la minería de criptomonedas de forma no autorizada. (NIST)

avanzada para este análisis requiere una comprensión profunda de los principios fundamentales de la ciberseguridad[3], así como un enfoque sistemático y exhaustivo para abordar las complejidades del malware moderno.

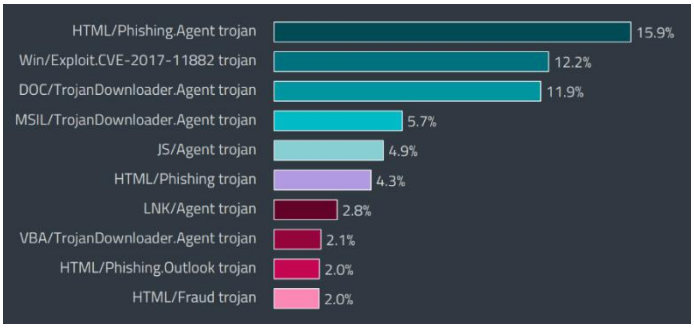
2.1. Definición de malware

El malware o software malicioso, se refiere a un programa que se inserta en un sistema, generalmente de forma encubierta, con la intención de comprometer la confidencialidad, integridad o disponibilidad de los datos, las aplicaciones o el sistema operativo de la víctima o de molestar o perturbar de otro modo a la víctima (NIST, 2019).

2.2. Incidentes relacionados con malware en los últimos años

El software[4] malicioso, constituye una de las principales amenazas en el ámbito de la seguridad informática. Desde los primeros días de la informática, este ha evolucionado significativamente en términos de complejidad y sofisticación, adaptándose constantemente para eludir las medidas de seguridad y causar daño a sistemas informáticos y usuarios en todo el mundo. Su impacto se ha hecho especialmente evidente en los últimos años, con ataques cada vez más frecuentes y destructivos. Desde el robo de datos confidenciales hasta la interrupción de servicios críticos y extorsión financiera, el malware representa una amenaza constante para individuos, empresas e instituciones gubernamentales (ESET, Security Report Latinoamérica, 2023). En respuesta a esta creciente amenaza, el campo del análisis de malware ha experimentado un rápido desarrollo, con enfoques dinámicos que buscan comprender y combatir activamente las nuevas variantes de malware que emergen constantemente en el panorama digital.

Ilustración 1: Detecciones de malware en el segundo semestre de 2022



Fuente: ESET Threat Report T1 2022

[3] La **ciberseguridad** es el proceso de proteger la información mediante la prevención, detección y respuesta a ataques. (NIST)
[4] El **software** son programas de computadora y datos asociados que pueden escribirse o modificarse dinámicamente durante la ejecución. (NIST)

Tabla 1: Descripción de detecciones de malware en el segundo semestre de 2022

No	Detección de malware	Descripción
1	**HTML/Phishing.Agent**	Este troyano malicioso se disfraza de correo electrónico legítimo con archivos adjuntos HTML[5]. Al abrir el adjunto, redirige al usuario a sitios web falsos que intentan robar credenciales y datos confidenciales.
2	**DOC/TrojanDownloader.Agent**	Documentos de Word maliciosos que descargan malware adicional, a menudo disfrazados de facturas, formularios o documentos legales importantes.
3	**Win/Exploit.CVE-2017-11882**	Explota una vulnerabilidad en Microsoft Equation Editor para ejecutar código malicioso y descargar más malware en el sistema comprometido.
4	**JS/Agent**	Archivos JavaScript[6] maliciosos y ofuscados que se alojan en sitios web legítimos comprometidos, con el objetivo de infectar a los visitantes.
5	**MSIL/TrojanDownloader.Agent**	Malware basado en .NET que descarga cargas adicionales, actuando como la primera capa de un paquete más complejo.
6	**HTML/Phishing**	Detección genérica de malware en URLs y archivos adjuntos de correo electrónico maliciosos.
7	**LNK/Agent**	Utiliza archivos de acceso directo de Windows para ejecutar otros archivos maliciosos y lograr persistencia en el sistema.
8	**VBA/TrojanDownloader.Agent**	Documentos de Office con macros maliciosas que descargan y ejecutan malware adicional cuando se habilitan las macros.
9	**HTML/Fraud**	Contenido HTML fraudulento, como sitios web de estafas, correos electrónicos y archivos adjuntos diseñados para engañar a las víctimas y obtener ganancias.
10	**MSIL/Spy.AgentTesla**	Troyano espía de código .NET que roba datos confidenciales, registra pulsaciones de teclas y accede a la cámara y micrófono.

Fuente: ESET Threat Report T1 2022

A mediados del año 2023 varios grupos de amenazas persistentes avanzadas (APT) intensificaron sus actividades maliciosas, demostrando una creciente sofisticación y diversificación en sus métodos de ataque. Según un nuevo informe de ESET, grupos norcoreanos como Lazarus y Kimsuky, así como los grupos rusos Sandworm y SturgeonPhisher, han estado muy activos, empleando nuevas técnicas y herramientas. (ESET, 2023).

A continuación, se describen algunos grupos e incidentes de seguridad relevantes en el panorama actual. Estos grupos pueden representar amenazas significativas para la seguridad cibernética y

[5] **HTML** (Lenguaje de Marcas de Hipertexto, del inglés HyperText Markup Language) es el componente más básico de la Web. (Mozilla)
[6] **JavaScript** es un lenguaje de programación ligero, interpretado, o compilado justo-a-tiempo (just-in-time) con funciones de primera clase. (Mozilla)

pueden estar involucrados en una variedad de actividades maliciosas, desde ataques de espionaje cibernético hasta operaciones de ransomware.

- El notorio grupo Lazarus, respaldado por Corea del Norte, ha continuado su enfoque en objetivos relacionados con criptomonedas[7]. En abril de 2023, se descubrió un nuevo malware Linux llamado OdicLoader y SimplexTea, vinculado al infame ataque a la cadena de suministro de 3CX. Posteriormente, se determinó que el código de SimplexTea formaba parte de una base de código común utilizada por Lazarus en todas las plataformas principales: Windows, Linux y macOS. En septiembre, se encontró una nueva variante de OdicLoader simulando ser una plataforma de trading de criptomonedas llamada MultiLayerSwap.
- Por su parte, el grupo Kimsuky ha ajustado sus enfoques, adoptando herramientas como OneNote, archivos CHM y accesos directos de Windows en sus campañas. Además, han reescrito parte de su malware en Go para evadir las detecciones. Una de sus principales campañas se ha centrado en enviar correos electrónicos de spearphishing[8] de alta calidad a analistas, académicos, investigadores y periodistas que se enfocan en asuntos relacionados con Corea del Norte.
- En cuanto al grupo ruso Sandworm, ha llevado a cabo varios ataques de borrado de datos en Ucrania utilizando variantes del malware RoarBat y NikoWiper, así como un nuevo wiper llamado SharpNikoWiper escrito en C#. Además, Sandworm ha utilizado un canal de Telegram pro-ruso para promover información sobre sus operaciones de cibersabotaje.

En al ámbito del comercio en línea el malware IIStealer representa una amenaza única que pone en peligro la relación de confianza entre vendedores y compradores en línea. Este malicioso complemento para el software de servidor web de Microsoft, Internet Information Services (IIS), permite a los atacantes acceder a toda la comunicación de red que fluye a través del servidor comprometido, incluyendo contraseñas, nombres de usuario e información de pago de transacciones de comercio electrónico (Hromcová, 2021).

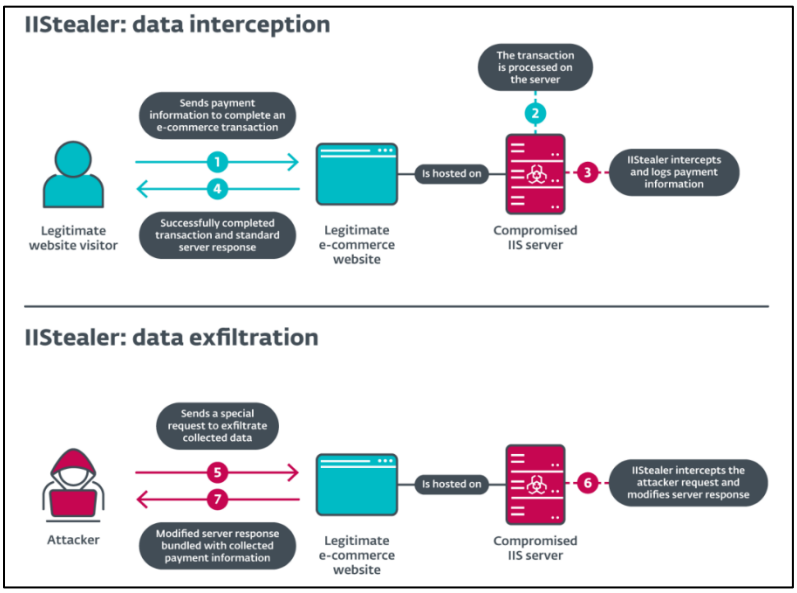
Incluso si un sitio web de comercio electrónico es de confianza y la comunicación está encriptada con SSL/TLS, los visitantes no tienen forma de saber el estado de seguridad de los servidores que alojan esos sitios web. Es allí donde se procesan sus datos y, sin que lo sepan, son robados

[7] Las **criptomendas** son un activo/crédito/unidad digital dentro del sistema, que se envía criptográficamente de un usuario de la red blockchain a otro. (NIST)
[8] El **spearphishing** consiste en una modalidad phishing dirigida contra un objetivo específico, en el que los atacantes intentan, mediante un correo electrónico, conseguir información confidencial de la víctima. (INCIBE)

por IIStealer. Este caso destaca el delicado equilibrio que debe mantenerse entre facilitar las ventas en línea y garantizar la seguridad cibernética.

Aunque los usuarios tomen precauciones, están expuestos a amenazas en los servidores que alojan los sitios web que visitan, sobre los cuales no tienen control ni visibilidad. (ESET, 2022)

Ilustración 2: interceptación y extracción de datos median IIStealer



Fuente: Zuzana Hromcová, ESET Industry Report on Retail

Según el reciente Reporte de Seguridad de ESET 2023, el malware continúa representando un grave peligro para las empresas y usuarios en Latinoamérica. A pesar de los avances en ciberseguridad, los ciberdelincuentes encuentran nuevas formas de propagar códigos maliciosos en la región.

Uno de los principales vectores de infección son las campañas de phishing[9], especialmente en países como Ecuador, Costa Rica, Colombia, Guatemala y El Salvador, que lideran las detecciones de este tipo de amenaza. Mediante ingeniería social, los atacantes engañan a las víctimas para que abran archivos adjuntos o hagan clic en enlaces maliciosos que descargan malware en sus sistemas.

El ransomware sigue siendo un protagonista, con más de 400 familias diferentes detectadas en 2022. Aunque solo el 21% de las empresas encuestadas admitió haber sufrido un ataque de

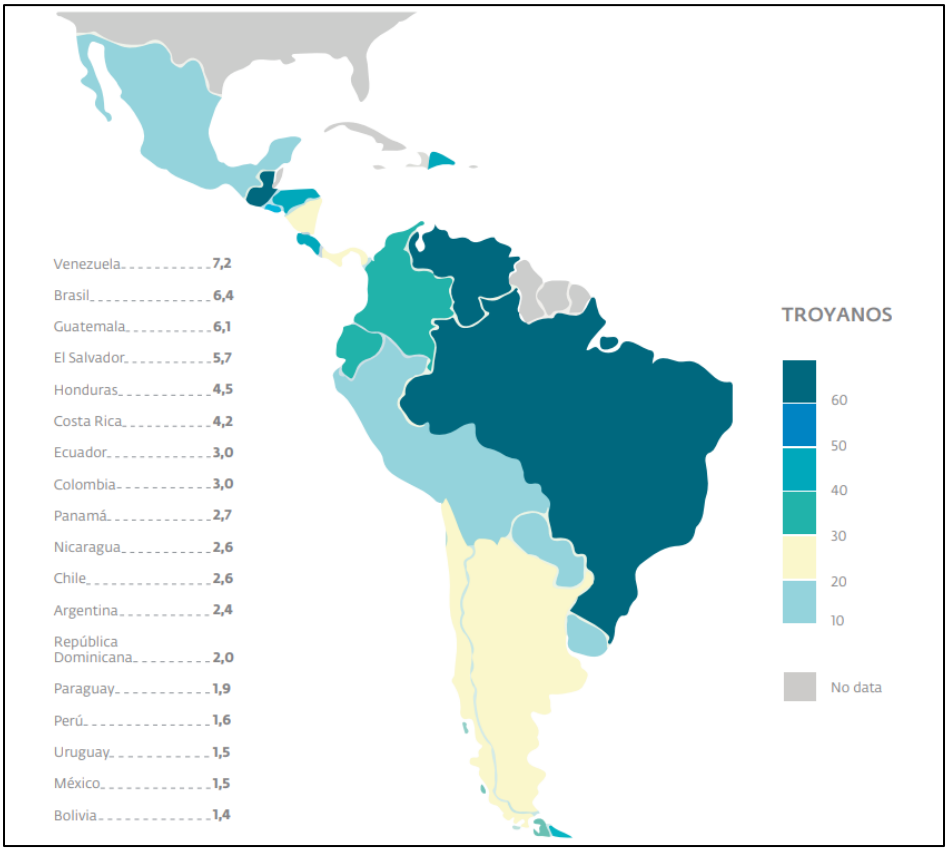
[9] El **phishing** consiste en engañar a personas para que revelen información personal sensible a través de medios informáticos engañosos. (NIST)

ransomware en los últimos dos años, el 96% expresó preocupación por esta amenaza. Los grupos de ransomware como Conti y Hive incluso comprometieron sistemas gubernamentales en Costa Rica.

Además del ransomware, el spyware[10] diseñado para robar datos e información confidencial está en aumento. Brasil y México encabezan las detecciones de este tipo de malware en la región. Los ciberdelincuentes también recurren a troyanos genéricos como droppers[11] y downloaders para infiltrar sistemas corporativos y descargar otras amenazas más peligrosas.

Según las investigaciones de ESET, muchas de estas campañas utilizan "malware común" conocido, lo que indica una falta de soluciones de seguridad adecuadas en las empresas de la región. Los atacantes no necesitan desarrollar malware sofisticado cuando el malware existente puede evadir las defensas.

Ilustración 3: Países con más detecciones de Troyanos en América Latina durante 2022



Fuente: ESET Security Report: Latinoamérica 2023

[10] El **spyware** es un software que se instala secreta o subrepticiamente en un sistema para recopilar información sobre individuos u organizaciones sin su conocimiento; un tipo de código malicioso. (NIST)
[11] Se denomina **dropper** a un tipo de troyano cuya función es descargar en el equipo víctima un malware que lleva embebido y cuyo payload generalmente se almacena cifrado. (Sol González)

A medida que el panorama de amenazas evoluciona, las empresas latinoamericanas deben mantenerse vigilantes y proactivas para salvaguardar sus sistemas y datos de los cada vez más sofisticados ataques de malware.

En este contexto, resulta fundamental comprender en profundidad qué es el malware, cómo opera y cuáles son las mejores prácticas para detectarlo y mitigarlo. Mediante el análisis exhaustivo de su comportamiento y características, los profesionales de la seguridad informática pueden desarrollar estrategias efectivas para proteger sistemas y datos críticos contra esta amenaza en el ciberespacio.

2.3. Tipos de malware por su comportamiento

El malware abarca una amplia gama de software malicioso, cada uno con características y comportamientos únicos. Desde los clásicos virus informáticos, diseñados para replicarse y propagarse de un sistema a otro, hasta los gusanos informáticos que se distribuyen de manera autónoma a través de redes, el panorama de amenazas es diverso y en constante evolución. (García Monje, 2017)

Los troyanos, o RAT (Remote Access Trojan), son una categoría particularmente peligrosa, ya que se presentan como programas legítimos, pero ocultan funciones maliciosas que permiten a los atacantes obtener acceso no autorizado a los sistemas comprometidos. Los keyloggers, por su parte, registran y roban información confidencial, como contraseñas y datos bancarios, mientras que los stealers se enfocan específicamente en robar información personal y financiera.

El spyware, diseñado para recopilar información sobre las actividades de los usuarios sin su conocimiento, representa una grave violación de la privacidad. Mientras tanto, el ransomware ha ganado notoriedad por su capacidad para bloquear el acceso a archivos y sistemas, exigiendo un rescate a cambio de restaurar el acceso. Nadie sabe realmente cuánto ganan los operadores de ransomware. Una investigación de la industria sitúa las demandas de rescate promedio en alrededor de $170.000, según Group-IB. Sin embargo, los investigadore indicaron también que los grupos más descarados piden decenas de millones de dólares: Sodinokibi (también conocido como REvil) exigió 50 millones de dólares cada uno a Acer y Quanta. (Kubovič, 2021)

Además, los criptomineros maliciosos aprovechan los recursos informáticos de las víctimas para llevar a cabo la minería de criptomonedas de forma no autorizada, una práctica conocida como cryptojacking. (Konoth, Wegberg, Moonsamy, & Bos, 2019)

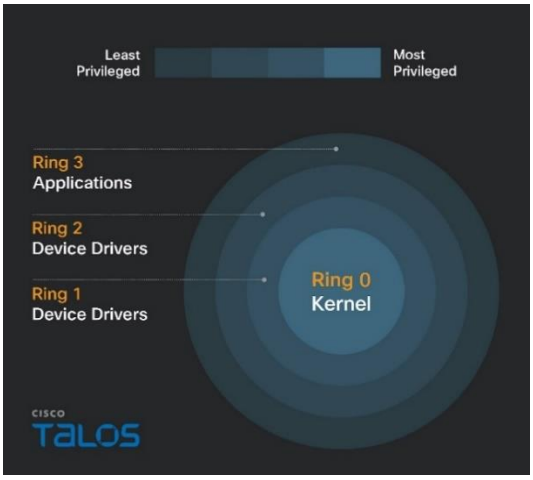
Sin embargo, una de las amenazas más sofisticadas y persistentes son las Amenazas Persistentes Avanzadas (APT, por sus siglas en inglés). Estas amenazas son llevadas a cabo por grupos de alto nivel, como agencias de inteligencia estatales o grupos de hackers respaldados por gobiernos, y tienen como objetivo específico infiltrarse en redes informáticas de organizaciones o entidades específicas para robar información confidencial, obtener acceso no autorizado a sistemas críticos o llevar a cabo actividades de espionaje industrial o gubernamental.

En la era digital actual, la seguridad informática es una prioridad fundamental. Comprender los diferentes tipos de malware y sus características es determinante para el desarrollo de estrategias de protección efectivas para mantenerse un paso adelante de los ciberdelincuentes.

2.4. Tipos de malware por privilegios

Mientras que la clasificación del malware por su funcionalidad y comportamiento es importante, otro aspecto fundamental es comprender los niveles de privilegios en los que opera. Los privilegios otorgados a este tipo de software malicioso determinan su capacidad para causar daños, evadir detección y llevar a cabo acciones maliciosas en un sistema comprometido. Esta clasificación por niveles de privilegios es esencial para los investigadores de seguridad, ya que les permite desarrollar estrategias efectivas de prevención y mitigación de riesgos. (Or-Meir, Nissim, Elovici, & Rokach, 2019)

Ilustración 4: Anillos de seguridad de la arquitectura x86



Fuente: Chris Neal, CISCO

En el nivel más básico, encontramos el malware que opera en modo usuario. Aunque tiene acceso limitado al sistema y a los recursos del usuario, este tipo de malware aún puede robar información

personal, instalar software no deseado o enviar spam. Sin embargo, su impacto se ve restringido por las limitaciones inherentes a los privilegios de usuario.

Un nivel más peligroso es el malware que opera en modo kernel[12]. Este tipo de malware, conocido como rootkit, tiene acceso a los niveles más profundos del sistema operativo, lo que le permite tomar el control completo del sistema, ocultar su presencia y realizar acciones sin restricciones, como controlar el tráfico de red, robar datos confidenciales o crear puertas traseras para el acceso remoto. (Poslušný, 2022)

Aún más preocupante es el malware que se ejecuta a nivel de hipervisor. El hipervisor es un software de virtualización que permite la ejecución de múltiples sistemas operativos y aplicaciones en una misma máquina física. El malware que opera a este nivel tiene privilegios elevados, incluso superiores a los del kernel del sistema operativo. Puede monitorear y manipular la ejecución de múltiples sistemas operativos y aplicaciones virtuales, eludiendo las medidas de seguridad implementadas en los sistemas operativos individuales.

Finalmente, el malware que infecta y compromete dispositivos de hardware físico, como dispositivos USB, dispositivos IoT (Internet de las cosas) y dispositivos médicos, representa una amenaza escalada. Al infectar el firmware[13] de estos dispositivos, este tipo de malware puede operar con privilegios muy elevados, incluso más allá de los privilegios de root en el sistema operativo.

A medida que el malware evoluciona y adquiere mayores privilegios, su capacidad para causar daños se vuelve más significativa. Los investigadores de seguridad deben mantenerse alerta y desarrollar estrategias de detección y mitigación acordes a estos niveles de privilegios. Sólo mediante una comprensión profunda de estas amenazas y sus capacidades podremos salvaguardar eficazmente nuestros sistemas y proteger la integridad de nuestra información.

2.5. Análisis del malware

Es una parte fundamental en la ciberseguridad moderna, ya que ayuda a comprender y combatir las constantes amenazas que enfrentan los sistemas informáticos. Sin embargo, en la actualidad, este proceso se enfrenta a una serie de desafíos significativos.

[12] El **kernel** es el elemento principal de los sistemas operativos,y es la interfaz fundamental entre el hardware de una computadora y sus procesos. Los comunica entre sí y gestiona los recursos de la manera más eficiente posible. (RedHat)
[13] El **firmware** es un programa de computadora y datos almacenados en el hardware, generalmente en una memoria de solo lectura (ROM) o en una memoria de solo lectura programable (PROM), de manera que los programas y datos no se pueden escribir ni modificar dinámicamente durante la ejecución de los programas. (NIST)

Uno de los problemas principales es la creciente sofisticación y diversificación del malware. Los ciberdelincuentes están constantemente desarrollando nuevas variantes que utilizan técnicas avanzadas de evasión y ocultamiento para eludir el análisis y la detección. Esto dificulta la tarea de los analistas, ya que deben estar al tanto de las últimas tendencias y técnicas utilizadas por los atacantes, además, el volumen de malware que se detecta diariamente es abrumador. Los analistas de seguridad se enfrentan a una avalancha constante de muestras que deben analizar y clasificar en un tiempo limitado. Esta gran cantidad de datos dificulta la identificación de amenazas emergentes y ralentiza el proceso de respuesta a incidentes de seguridad. (Gandotra, Bansal, & Sofat, 2014)

Otro desafío importante es la falta de herramientas y recursos especializados para el análisis de malware. A medida que aumenta la complejidad de este, se necesitan herramientas más avanzadas y sofisticadas para analizarlo de manera efectiva. Sin embargo, muchas organizaciones carecen de los recursos necesarios para adquirir y mantener estas herramientas, lo que dificulta su capacidad para detectar y responder adecuadamente a las amenazas.

2.1.5. Análisis estático

Consiste en examinar el código de un archivo sospechoso sin ejecutarlo. Este enfoque proporciona una visión detallada de las características de este archivo antes de que pueda causar daño en un sistema. Durante el análisis estático, se examinan diversos atributos del archivo, como el código fuente, el binario, las cadenas de texto y las estructuras de datos, en busca de indicadores de comportamiento malicioso. (Bhojani, 2014)

Una de las ventajas clave de este tipo de análisis es su capacidad para identificar patrones de código malicioso y firmas conocidas sin la necesidad de ejecutar el archivo sospechoso. Esto permite detectar y clasificar rápidamente el malware conocido, así como identificar posibles indicadores de compromiso en archivos desconocidos.

El análisis estático puede realizarse manualmente por expertos en seguridad informática, que examinan el código y buscan características sospechosas. Sin embargo, con el aumento del volumen y la complejidad del malware, se ha vuelto cada vez más común el uso de herramientas de análisis estático automatizado. Estas herramientas utilizan algoritmos y técnicas avanzadas para escanear archivos en busca de indicadores de comportamiento malicioso de manera eficiente y rápida, de esta forma, pueden analizar grandes cantidades de archivos en poco tiempo,

lo que las hace especialmente útiles en entornos donde se enfrentan a una gran cantidad de muestras a diario.

2.2.5. Análisis dinámico

Consiste en ejecutar y observar el comportamiento de un programa malicioso en un entorno controlado y monitorear sus acciones en tiempo real. A diferencia del análisis estático, que se centra en examinar el código fuente o el archivo en reposo, el análisis dinámico se enfoca en observar cómo el archivo sospechoso interactúa con el sistema durante su ejecución. (Stegger, 2021)

Durante este análisis, el archivo sospechoso se ejecuta en un entorno controlado y aislado, como un sandbox[14] o una máquina virtual, que simula un sistema operativo y entorno de red reales. Esto permite a los investigadores observar el comportamiento de este archivo sin arriesgar la integridad de sistemas reales.

El análisis dinámico proporciona información valiosa sobre las acciones maliciosas que realiza el malware, como la creación o modificación de archivos, la conexión a servidores remotos, la inyección de código en procesos legítimos, la modificación del registro del sistema y más. Además, permite identificar técnicas de evasión utilizadas por el este para evitar el análisis y la detección, como el uso de técnicas de ofuscación, encriptación de datos, carga dinámica de código y manipulación del entorno de ejecución.

Las herramientas de análisis dinámico, tanto manuales como automatizadas, juegan un papel crucial en este proceso. Las herramientas automatizadas permiten ejecutar automáticamente archivos sospechosos en entornos controlados y recopilar datos sobre su comportamiento. Estas herramientas pueden registrar actividades como llamadas al sistema, modificaciones en el sistema de archivos y registros, interacciones de red y cambios en la configuración del sistema.

El análisis dinámico es esencial para comprender completamente la funcionalidad y el impacto de un malware, así como para desarrollar estrategias efectivas de detección, prevención y mitigación. Al observar su comportamiento en un entorno seguro y controlado, los investigadores pueden identificar patrones de actividad maliciosa y desarrollar medidas de seguridad adecuadas para proteger los sistemas contra futuros ataques.

[14] **Sandbox** es un entorno de ejecución restringido y controlado que evita que software potencialmente malicioso, como código móvil, acceda a cualquier recurso del sistema excepto aquellos para los cuales el software está autorizado. (NIST)

2.6. Técnicas que utiliza el malware

El malware, o software malicioso, es una de las mayores amenazas en el mundo digital actual. Desde virus y troyanos hasta ransomware y spyware, el malware se ha convertido en una herramienta poderosa y omnipresente en manos de los ciberdelincuentes. ¿Cómo logra el malware infiltrarse y causar estragos en nuestros sistemas? Para entenderlo, es crucial explorar las diversas técnicas que utiliza para eludir las defensas y llevar a cabo sus operaciones maliciosas. Desde la ofuscación de código hasta la explotación de vulnerabilidades, el malware despliega un arsenal de estrategias ingeniosas para evadir la detección y mantenerse oculto en los dispositivos (Uppal, Mehra, & Verma, 2014). A continuación, se exploran algunas de estas tácticas utilizadas por el malware y su impacto en la seguridad cibernética.

Algunas de las estrategias más comunes utilizadas por el malware son la escalada de privilegios, los algoritmos de generación de dominios (DGA), el malware sin archivos, la técnica "Living off the Land", la inyección de DLL, la inyección de código y de hilos, los períodos prolongados de inactividad (Extended Sleeps), la explotación de vulnerabilidades de día cero y la técnica Stegosploit.

La escalada de privilegios permite al malware obtener acceso a recursos y funciones restringidas en un sistema operativo, aprovechando vulnerabilidades o mecanismos de seguridad deficientes (Microsoft, 2021). Por otro lado, los DGA generan una gran cantidad de nombres de dominio de forma dinámica, dificultando la detección y el bloqueo del tráfico malicioso por parte de los sistemas de seguridad. (Porolli, 2021)

El malware sin archivos opera directamente en la memoria del sistema, evitando dejar rastros en el disco duro y eludiendo la detección de los antivirus tradicionales. La técnica "Living off the Land" aprovecha herramientas y funciones legítimas presentes en los sistemas operativos y entornos de red para llevar a cabo actividades maliciosas, dificultando su detección. (Ongun, Stokes, & Or, 2021)

Otras estrategias empleadas por el malware incluyen la inyección de DLL, la inyección de código y la inyección de hilos en procesos legítimos, lo que permite ejecutar acciones maliciosas de forma encubierta (Walter, 2020). Los períodos prolongados de inactividad o "Extended Sleeps" ayudan al malware a ocultar su actividad y dificultar su detección.

Además, el malware puede explotar vulnerabilidades de día cero, desconocidas para los desarrolladores y sin solución disponible, lo que lo hace especialmente peligroso (Erica Eng,

2015). La técnica Stegosploit, por su parte, oculta código malicioso dentro de archivos de imagen, engañando a los sistemas de detección y a los usuarios.

Algunas variantes de malware pueden modificar las políticas de dominio y los tokens de acceso[15] en entornos de Windows, lo que les permite evadir restricciones de seguridad, propagarse en la red y obtener acceso no autorizado a recursos y operaciones.

Otra de las técnicas más peligrosas empleadas por el malware es la explotación de vulnerabilidades en sistemas o aplicaciones para eludir las características de seguridad. Aprovechando errores de programación, el malware puede ejecutar código malicioso y evitar ser detectado por el software de seguridad defensiva. (Intelligence, 2015)

Además, el malware puede alterar estratégicamente los permisos de archivos y directorios para ocultar sus componentes y actividades de las herramientas de seguridad y los usuarios del sistema. Al manipular estos permisos, el malware dificulta su detección y eliminación, prolongando su presencia en el sistema comprometido. (Team S. T., 2021)

Otra técnica utilizada es el ocultamiento de componentes, donde el malware manipula o esconde sus archivos, procesos y actividades maliciosas para evitar ser identificado por las herramientas de seguridad y los usuarios. (Arntz, 2015)

En algunos casos, el malware puede optar por deshabilitar directamente las defensas de seguridad presentes en el sistema, como desactivar antivirus, firewalls, el registro de eventos y modificar configuraciones críticas de seguridad. Estas acciones permiten al malware operar sin obstáculos. (Report, 2022)

Complementando estas técnicas, el malware también puede borrar evidencia de su presencia o actividad en el sistema comprometido, eliminando registros de eventos, historiales de comandos, archivos relacionados y modificando marcas de tiempo, dificultando así su detección y análisis.

Otras estrategias más sofisticadas incluyen la ejecución indirecta de comandos, aprovechando funcionalidades legítimas del sistema operativo y ofuscando los comandos utilizados, así como el enmascaramiento, donde el malware adopta características similares a elementos legítimos del sistema para evitar ser identificado. (Pradhan, 2022)

[15] Un **token de acceso** es un objeto que describe el contexto de seguridad de un proceso o subproceso. La información de un token incluye la identidad y los privilegios de la cuenta de usuario asociada al proceso o subproceso. (Microsoft)

En entornos empresariales, el malware puede incluso registrar un controlador de dominio falso para manipular datos en Active Directory[16], o alterar los controles de confianza que advierten a los usuarios sobre actividades no confiables. (Delpy & TOUX, 2018)

Por último, una técnica destacada es la ejecución de binarios firmados del sistema, donde el malware ejecuta contenido malicioso a través de archivos firmados por Microsoft o de confianza, evadiendo las defensas basadas en procesos y firmas.

Estas técnicas complejas demuestran la constante evolución del malware y la necesidad de mantener una postura de seguridad sólida y actualizada para proteger los sistemas y datos de estas amenazas en constante cambio.

2.1.6. Técnicas de Acceso inicial

En el implacable mundo de las amenazas cibernéticas, el malware ha desarrollado sofisticadas técnicas para obtener un punto de entrada inicial en redes y dispositivos, con el fin de instalar su carga maliciosa. Estas tácticas de acceso inicial son el primer paso en una cadena de eventos que pueden culminar en graves violaciones de seguridad y comprometer la integridad de los sistemas.

Una de las estrategias empleadas por el malware es la inyección de contenido, donde se aprovechan canales de transferencia de datos comprometidos para inyectar código malicioso en el tráfico de red en línea. En lugar de atraer a las víctimas a sitios web maliciosos, los ciberdelincuentes pueden manipular el tráfico y entregar cargas útiles adicionales a sistemas ya comprometidos. (Faou, 2023)

Otra técnica es el drive-by compromise, donde se aprovechan vulnerabilidades en sitios web legítimos o comprometidos para instalar malware en los sistemas de los visitantes sin su conocimiento ni interacción directa. Esto se logra mediante la inyección de código malicioso en el navegador del usuario, lo que permite la descarga e instalación silenciosa de malware. (Chen J. C., 2018)

El malware también puede recurrir a adiciones de hardware, como la inserción de dispositivos de almacenamiento o componentes comprometidos, para facilitar la persistencia y el control remoto

[16] **Active Directory** es un servicio de directorio de Microsoft para la gestión de identidades en redes de dominio de Windows. (NIST)

de un sistema infectado. Esta técnica física puede ser difícil de detectar y eliminar. (Golovanov, 2018)

Además, el compromiso de la cadena de suministro ha surgido como una estrategia altamente sofisticada, donde el malware se infiltra en los sistemas de destino a través de proveedores de servicios o productos confiables. Al comprometer la infraestructura de estos proveedores, el malware puede distribuirse y ejecutarse en los sistemas de los usuarios finales de manera encubierta. (IBM, 2017)

Finalmente, el malware puede explotar las relaciones de confianza entre organizaciones y sus proveedores o socios externos para obtener acceso no autorizado a redes y sistemas. Los atacantes se aprovechan de las conexiones legítimas y a menudo menos vigiladas con terceros, como contratistas de TI o proveedores de infraestructura, para comprometer cuentas válidas y utilizarlas en su beneficio. (Team C. T., REvil/Sodinokibi Ransomware, 2019)

2.2.6. Técnicas de evasión de análisis

En el incesante mundo de las amenazas cibernéticas, el malware ha desarrollado sofisticadas estrategias para evadir la detección y el análisis por parte de los sistemas de seguridad. Estas técnicas de evasión representan un desafío constante para los profesionales de la ciberseguridad, quienes deben mantenerse un paso adelante en la identificación y mitigación de estas amenazas.

Una de las técnicas más ampliamente utilizadas por los autores de malware es la ofuscación de código. Esta estrategia consiste en modificar el código para dificultar la extracción de firmas derivadas del código binario[17], lo que dificulta la detección por parte de los métodos tradicionales de análisis utilizados por el software antivirus. El polimorfismo y el metamorfismo son ejemplos destacados de ofuscación de código. El primero permite al malware cambiar su código binario y estructura interna de manera dinámica, generando variantes únicas que no coinciden con las firmas conocidas. Por otro lado, el metamorfismo reorganiza y modifica dinámicamente el código fuente, creando múltiples variantes del mismo malware con estructuras y comportamientos diferentes, dificultando su detección basada en patrones predefinidos.

Otra estrategia común empleada por los desarrolladores de malware son los empaquetadores y cifradores. Los empaquetadores comprimen el malware, reduciendo su tamaño y ocultando su

[17] El **código binario** es una codificación usada para la representación de textos, o procesadores de instrucciones de computadora, utilizando el sistema binario. (Wikipedia)

verdadero propósito, mientras que los cifradores utilizan algoritmos de cifrado para codificar el código malicioso, haciéndolo ilegible para los escáneres de antivirus convencionales. Ambas técnicas desempaquetan y descifran el código en la memoria RAM antes de ejecutarlo, evadiendo la detección basada en firmas en el disco.

El malware también puede estar programado para no activar su carga útil maliciosa si detecta la presencia de un depurador, utilizando técnicas anti-depuración. Estas técnicas permiten al malware detectar si está siendo ejecutado en un entorno depurado, comúnmente utilizado por los analistas de seguridad, y pueden intentar interferir con las funciones de depuración del sistema operativo. (Team C. T., 2022)

Finalmente, el malware puede evadir entornos de virtualización y de análisis, como máquinas virtuales y entornos de pruebas automatizados, utilizando técnicas de evasión de virtualización y sandbox. Esto implica comprobar la configuración del hardware, buscar archivos y procesos característicos de entornos virtualizados, o ejecutar instrucciones específicas de CPU que indican la presencia de una máquina virtual. Una vez detectado, el malware puede cambiar su comportamiento para permanecer latente y evitar el análisis por parte de los sistemas de seguridad. (Torello & Guibernau, 2021)

Estas técnicas de evasión de análisis representan un desafío constante para los profesionales de la ciberseguridad, quienes deben mantenerse al tanto de estas estrategias y desarrollar métodos de detección y mitigación más avanzados.

2.3.6. Técnicas de evasión de detección

El constante avance tecnológico ha llevado a una escalada en las tácticas utilizadas por el malware para evadir la detección por parte de los sistemas de seguridad. Estas técnicas de evasión, empleadas por el este con el fin de eludir los mecanismos de defensa, representan un desafío continuo para los profesionales de la seguridad informática. El malware utiliza una variedad de estrategias sofisticadas para ocultar su presencia y evitar su identificación. Una de las técnicas más comunes es el camuflaje, que implica ocultar su código malicioso al mimetizarse con archivos o procesos legítimos del sistema operativo. Esta técnica puede incluir la modificación de nombres de archivos, ubicaciones o atributos para pasar desapercibido ante los sistemas de detección.

Además del camuflaje, puede emplea la inyección de código como una técnica efectiva para evadir la detección. Esta técnica consiste en insertar su propio código malicioso en procesos

legítimos del sistema, lo que le permite ejecutarse de manera encubierta y eludir las medidas de seguridad que detectan la creación de nuevos procesos maliciosos. (Hosseini, 2017)

Otra técnica comúnmente utilizada es la explotación de vulnerabilidades del sistema operativo o de aplicaciones para ejecutar su código malicioso de forma sigilosa. Al aprovechar estas vulnerabilidades, puede eludir las medidas de seguridad existentes y propagarse dentro del sistema sin ser detectado.

Además, el malware puede emplear técnicas basadas en el comportamiento para evadir la detección por parte de los sistemas de seguridad. Esto incluye la modificación dinámica de su comportamiento en respuesta a la detección o el análisis, como interrumpir sus actividades maliciosas durante ciertos períodos de tiempo o activar mecanismos de autodestrucción para evitar su captura y análisis.

2.4.6. Técnicas de evasión de red

En el panorama de las amenazas cibernéticas, el malware ha desarrollado sofisticadas técnicas para evadir la detección a través de conexiones de red. Estas tácticas son esenciales para los actores maliciosos que buscan mantener su presencia en sistemas comprometidos y comunicarse de manera encubierta con servidores de comando y control u otras infraestructuras maliciosas.

Una de las estrategias más efectivas empleadas por el malware es la conexión indirecta o inversa. En esta técnica, el sistema comprometido establece una conexión saliente hacia el servidor de comando y control, aprovechando que la mayoría de las veces los firewalls[18] y otras medidas de seguridad no están configurados para bloquear las conexiones salientes. De esta manera, el malware puede comunicarse de forma discreta con su infraestructura sin ser detectado fácilmente por las defensas de red tradicionales.

Otra táctica común es el uso de sesiones cifradas, donde las comunicaciones entre el malware y su servidor de comando y control se cifran o codifican, ocultando el contenido y dificultando que las soluciones de seguridad tradicionales puedan inspeccionar o detectar el tráfico malicioso. (Research, 2018)

El malware también puede aprovechar los protocolos de capa de aplicación, como HTTP, HTTPS, FTP o protocolos de correo electrónico, para comunicarse y transferir datos entre sistemas,

[18] Un **firewall** es una puerta de enlace que limita el acceso entre redes de acuerdo con la política de seguridad local. (NIST)

camuflando sus actividades dentro del tráfico de red legítimo (Chen, Sasson, & Zelivansky, 2021). Además, puede utilizar protocolos OSI que no están en la capa de aplicación, como ICMP, UDP o SOCKS, para ocultar sus comunicaciones. (Holmes, 2015)

Una táctica adicional es el uso de puertos no estándar, combinando protocolos y puertos que normalmente no se asocian entre sí, lo que dificulta la detección y el análisis de las comunicaciones maliciosas. (Harbison, 2021)

El tunneling[19] es otra técnica utilizada por el malware, donde los paquetes de datos se encapsulan dentro de otros protocolos de red legítimos, como SSH, para pasar desapercibidos mientras se comunican con su servidor de comando y control. (Gatlan, 2019)

Además, el malware puede utilizar proxies[20] internos o externos para enrutar el tráfico de red y ocultar la comunicación con su infraestructura de comando y control, evitando así la detección y el seguimiento. Incluso pueden encadenar múltiples proxies para ocultar aún más el origen del tráfico malicioso.

Estas tácticas de evasión de red representan un desafío constante para los profesionales de la ciberseguridad, quienes deben mantenerse al tanto de estas estrategias y desarrollar métodos de detección y mitigación más avanzados. Sólo mediante una comprensión profunda de estas amenazas y sus tácticas de evasión podremos salvaguardar eficazmente nuestras redes y sistemas.

2.5.6. Técnicas de Ejecución

En el implacable mundo de las amenazas cibernéticas, el malware ha desarrollado sofisticadas técnicas para iniciar su funcionamiento en sistemas objetivo. Estas tácticas de ejecución son cruciales para que el software malicioso pueda llevar a cabo sus acciones maliciosas y comprometer la integridad de los sistemas infectados.

Una de las estrategias empleadas por el malware es el uso de intérpretes de comandos y scripts[21] nativos del sistema operativo. Aprovechando líneas de comandos como cmd.exe en Windows, bash en Linux/Unix o PowerShell, el malware puede ejecutar secuencias de comandos o

[19] El **tunneling** es una tecnología que permite a una red enviar sus datos a través de las conexiones de otra red. El túnel funciona encapsulando un protocolo de red dentro de paquetes transportados por la segunda red.
[20] Un **proxy** es un dispositivo o programa intermediario que proporciona comunicación y otros servicios entre un cliente y un servidor.
[21] Un **Script** es una secuencia de instrucciones, que van desde una simple lista de comandos del sistema operativo hasta declaraciones completas de un lenguaje de programación, que un intérprete puede ejecutar automáticamente.

instrucciones específicas que facilitan diversas actividades maliciosas, como la propagación, la exfiltración de datos o la escalada de privilegios. (Pantazopoulos, 2018)

Además, el malware puede recurrir al uso directo de las Interfaces de Programación de Aplicaciones (API) nativas del sistema operativo. Estas API de bajo nivel permiten interactuar directamente con componentes críticos, permitiendo al malware realizar una amplia gama de acciones sin necesidad de invocar herramientas o comandos externos que podrían ser detectados por soluciones de seguridad. (Salem, 2022)

Otra técnica utilizada es la creación de tareas programadas en el sistema operativo del host. Esto permite que el malware se ejecute automáticamente en momentos específicos o bajo ciertas condiciones, asegurando su persistencia en el sistema afectado, incluso después de reinicios o en intervalos definidos.

El malware también ha explorado el uso de entornos de computación sin servidor en la nube, como AWS Lambda, Google Cloud Functions o Azure Functions. Al implementar su código malicioso como una función en estos entornos, el malware puede ejecutar sus operaciones de manera distribuida y escalable, sin requerir la configuración y administración de servidores tradicionales. (Muir, 2022)

El uso de herramientas legítimas de despliegue de software, como Microsoft SCCM, Puppet o Ansible, también ha sido explotado por el malware. Al abusar de estas herramientas, el malware puede distribuir y ejecutar sus componentes maliciosos en sistemas comprometidos de manera más efectiva y encubierta. (Marvi, y otros, 2023)

Finalmente, los atacantes pueden recurrir a técnicas de ingeniería social, como el phishing, para engañar a los usuarios y hacer que ejecuten voluntariamente el malware, abriendo archivos adjuntos maliciosos o haciendo clic en enlaces perjudiciales.

2.6.6. Técnicas de persistencia

El software malicioso, emplea diversas técnicas para mantener su presencia en los sistemas infectados de manera persistente, incluso después de reiniciar o apagar el dispositivo. Estas técnicas aseguran que el malware permanezca activo y pueda ejecutar sus funciones maliciosas de forma continua. Algunas de las técnicas más comunes se describen a continuación.

Una técnica ampliamente utilizada es la ejecución automática al iniciar sesión o arrancar el sistema. El malware modifica la configuración del sistema, como entradas en el Registro de

Windows, tareas programadas, carpetas de inicio, servicios del sistema y scripts de inicio, para asegurar su ejecución automática durante el arranque o el inicio de sesión del usuario.

El malware también puede abusar de extensiones de navegador para obtener acceso persistente a los sistemas de las víctimas. Las extensiones maliciosas pueden robar información, navegar a sitios web en segundo plano e incluso minar criptomonedas. (Brinkmann, 2017)

Además, el malware puede comprometer clientes de software, como clientes SSH, FTP, correo electrónico y navegadores web, modificando sus binarios para ejecutar cargas útiles maliciosas cuando estas aplicaciones están en uso. (Hrčka, 2021)

La ejecución desencadenada por eventos aprovecha eventos específicos del sistema o del usuario para ejecutar código malicioso automáticamente, permitiendo al malware permanecer latente hasta que se cumplan ciertas condiciones.

Otra técnica es la modificación del proceso de autenticación, donde los atacantes alteran los mecanismos de autenticación de sistemas, aplicaciones o servicios para facilitar el acceso no autorizado o evadir controles de seguridad. (Dumont, M.Léveillé, & Porcher, 2018)

El abuso de características de arranque automático en aplicaciones de Microsoft Office, como macros, add-ins[22] maliciosos y plantillas automáticas, también se utiliza para ejecutar código malicioso al iniciar estas aplicaciones. (Raggi, 2021)

Estas técnicas de persistencia permiten al malware mantener su presencia en los sistemas comprometidos, dificultando su detección y eliminación, y facilitando la realización de actividades maliciosas de manera continua.

2.7. Técnicas para la detección de malware

Métodos utilizados para identificar la presencia de software malicioso en sistemas informáticos. Algunas de estas técnicas son la firma de virus, análisis heurístico, detección de comportamiento, entre otras.

2.1.7. Firma de virus

[22] Un **add-ins** o complemento es un programa de software que amplía las capacidades de programas más grandes.

Enfoque clásico utilizado en la detección de malware. Consiste en identificar patrones específicos conocidos de código malicioso, también llamados firmas, dentro de archivos o programas sospechosos. Estas firmas son secuencias únicas de bytes que están asociados con muestras de malware previamente identificadas y analizadas (Uppal, Mehra, & Verma, 2014). Un software antivirus o un sistema de detección de intrusiones busca coincidencias exactas entre las firmas conocidas de malware y los archivos en el sistema que están siendo escaneados. Si se encuentra una coincidencia, se considera que el archivo es malicioso y se toman las medidas adecuadas, como la cuarentena o eliminación del archivo infectado.

Esta técnica es eficaz para detectar malware conocido y variantes conocidas de amenazas. Sin embargo, tiene limitaciones significativas, ya que no puede detectar malware desconocido o variantes modificadas que no coincidan exactamente con las firmas conocidas. Además, los atacantes pueden evadir esta técnica fácilmente modificando ligeramente el código malicioso para evitar que coincida con las firmas conocidas.

2.2.7. Análisis Heurístico

Se centra en el comportamiento y las características generales del software para identificar posibles amenazas. A diferencia del enfoque de la Firma de Virus, que se basa en la identificación de patrones específicos conocidos de malware, el análisis heurístico examina el comportamiento del programa para determinar si es malicioso o sospechoso. En lugar de buscar características exactas asociadas con el malware conocido, el análisis heurístico evalúa el comportamiento del programa en busca de acciones o actividades que podrían indicar un comportamiento malicioso. Esto puede incluir actividades como la modificación del registro del sistema, la creación o eliminación de archivos en ubicaciones críticas, la inyección de código en procesos legítimos, la comunicación con dominios maliciosos conocidos, entre otros. (Miao, 2015)

El análisis heurístico puede utilizar reglas predefinidas o algoritmos de aprendizaje automático para identificar comportamientos sospechosos. Por ejemplo, un motor de antivirus puede estar configurado para detectar cualquier programa que intente modificar archivos críticos del sistema operativo sin autorización. Una de las ventajas de este análisis es su capacidad para detectar malware desconocido o variantes modificadas que no coinciden con las firmas de virus conocidas. Sin embargo, también puede generar falsos positivos, ya que algunas acciones legítimas del software pueden ser interpretadas como maliciosas por los algoritmos heurísticos.

2.3.7. Detección de Comportamiento

Utilizado para identificar y mitigar amenazas basadas en el análisis del comportamiento de los sistemas y el tráfico de red. A diferencia de la detección basada en firmas, que se centra en identificar patrones específicos de código malicioso, la detección de comportamiento se enfoca en detectar actividades anómalas o sospechosas que podrían indicar la presencia de amenazas. Esta técnica implica monitorear constantemente el comportamiento de los sistemas, aplicaciones y usuarios en busca de desviaciones de los patrones normales de actividad. (Bayer, Comparetti, Hlauschek, Kruegel, & Kirda, 2009)

Para llevar a cabo la detección de comportamiento, se utilizan técnicas como el análisis de registros de eventos, la monitorización de la actividad del sistema y la red, la detección de anomalías estadísticas y el uso de algoritmos de aprendizaje automático para identificar patrones de comportamiento anómalos.

Una de las ventajas de la detección de comportamiento es su capacidad para detectar amenazas desconocidas y ataques sofisticados que pueden eludir las técnicas de detección basadas en firmas. Sin embargo, al igual que el análisis heurístico, también puede generar un mayor número de falsos positivos, ya que cualquier desviación de los patrones normales de comportamiento puede ser considerada como una potencial amenaza.

2.4.7. Entorno Controlado para Análisis de malware

Es un espacio seguro y aislado diseñado específicamente para evaluar muestras de malware de manera controlada y sin comprometer sistemas operativos reales. Utiliza técnicas como máquinas virtuales o contenedores para crear un entorno virtual donde el malware pueda ser ejecutado de forma segura y aislada.

Sus características principales incluyen el aislamiento total del entorno del resto de la red y los sistemas operativos, lo que impide la propagación del malware fuera del entorno controlado. También se implementan herramientas de monitorización para registrar y analizar todas las actividades realizadas por el malware, como cambios en el sistema de archivos, comunicaciones de red y modificaciones en la memoria.

El análisis se realiza de forma dinámica, ejecutando el malware en un entorno controlado para observar su comportamiento en tiempo real y comprender sus funcionalidades sin poner en

peligro la seguridad del sistema. Se utilizan herramientas especializadas como depuradores, sistemas de detección de intrusiones y herramientas de análisis de memoria para llevar a cabo este proceso.

2.5.7. Indicadores de Compromiso (IoC) para Detección de Amenazas

Las amenazas informáticas evolucionan con ataques cada vez más sofisticados, siendo el malware el vector de ataque más utilizado. Los ataques de día cero (0days) representan un desafío particular, ya que no existe análisis previo del código, por lo que los indicadores de compromiso (IoC) son menos efectivos inicialmente.

Los indicadores de compromiso (IoC por sus siglas en inglés) son artefactos forenses recolectados de una intrusión que son identificados en la red, en un host o equipo. Estos IoC ayudan a los profesionales de seguridad a identificar cualquier tipo de amenaza que indique una brecha de seguridad a través de una vulnerabilidad. (Ponce Larreategui, 2021)

Tipos de IoC.

- **Artefactos basados en red**: Recibidos desde servidores, puertos, proxys, etc. Incluyen capturas de paquetes, estado de la red y sesiones.
- **Artefactos basados en host**: Recibidos desde el equipo, como registros del sistema y sistema de archivos.
- **IPs:** Identificar las direcciones IP del comando y control es clave para detectar conexiones maliciosas.
- **URLs**: Identificar las URLs asociadas a una botnet y sus IPs relacionadas para su bloqueo.
- **Puertos y servicios**: Analizar puertos y servicios como DNS, HTTP, TCP, UDP, etc. utilizados.
- **Registros**: Los cambios en el registro (persistencia) indican una computadora infectada.
- **Procesos**: Revisar procesos en ejecución, cargas de DLL, parámetros, etc. es clave para identificar malware.

Estas piezas de información forense como direcciones IP, URLs, hashes de archivos, firmas de malware, permiten detectar actividades maliciosas y amenazas en los sistemas.

2.6.7. Tabla de referencia de las Técnicas MITRE

La tabla de referencia de las Técnicas MITRE (MITRE Techniques Reference) es una herramienta invaluable en la comunidad de ciberseguridad. Esta tabla presenta una recopilación organizada de tácticas y técnicas utilizadas por adversarios en ciberataques, catalogadas según el marco de ATT&CK de MITRE. La tabla proporciona una visión general rápida de las diferentes formas en que los atacantes pueden comprometer sistemas y redes, lo que ayuda a los profesionales de seguridad a comprender mejor las amenazas y a fortalecer las defensas. Además, sirve como una referencia valiosa para la investigación de incidentes, la planificación de defensa y la evaluación de la postura de seguridad de una organización. Para obtener la lista completa de técnicas, consultar en https://attack.mitre.org/techniques/enterprise/. Para ver detalles sobre las técnicas encontradas en los análisis de malware realizados en esta investigación ver **Anexo A**.

2.8. El futuro del malware con Inteligencia Artificial

El futuro del malware con Inteligencia Artificial es un tema que despierta preocupación y atención en el ámbito de la ciberseguridad. Con el avance constante de la tecnología, los ciberdelincuentes podrán adoptar herramientas cada vez más sofisticadas para llevar a cabo sus ataques. La integración de la Inteligencia Artificial en el desarrollo de malware representa un cambio significativo en el panorama de la seguridad informática. Esta nueva generación de malware podría aprender de su entorno, adaptarse a las defensas de seguridad y evolucionar de manera autónoma, lo que lo haría extremadamente difícil de detectar y combatir. Ante este escenario, es crucial que la comunidad de la ciberseguridad esté alerta y desarrolle estrategias innovadoras para hacer frente a esta amenaza emergente y proteger la integridad de los sistemas y datos en un mundo cada vez más digitalizado.

2.1.8. Gusano informático con IA Generativa

En un estudio innovador realizado por Stav Cohen, Ron Bitton, y Ben Nassi, se presenta "Morris II", el primer gusano informático conceptualizado para atacar ecosistemas impulsados por Inteligencia Artificial Generativa (GenAI). Este gusano, nombrado en honor al primer gusano de Internet, el "Gusano Morris", explora vulnerabilidades únicas en aplicaciones GenAI, replicándose a través de entradas adversarias auto replicantes.

Morris II utiliza prompts adversarios auto replicantes para inducir a modelos GenAI a replicar y propagar el código malicioso sin interacción humana, evidenciando un método de ataque novedoso y altamente efectivo. El gusano se probó en asistentes de correo electrónico

potenciados por GenAI, demostrando su capacidad para realizar actividades maliciosas como el envío de spam y la exfiltración de datos de manera autónoma.

El descubrimiento de Morris II abre un debate urgente sobre las implicaciones de seguridad de los ecosistemas GenAI. A medida que estos sistemas se vuelven más integrados en nuestro entorno digital, la necesidad de estrategias de seguridad robustas se hace evidente. El estudio propone contramedidas, incluyendo el desarrollo de modelos GenAI que puedan identificar y neutralizar prompts adversarios, así como la importancia de una colaboración más estrecha entre investigadores en seguridad y desarrolladores de GenAI para prevenir ataques futuros.

El trabajo de Cohen, Bitton, y Nassi no solo destaca una nueva clase de amenaza cibernética, sino que también sirve como un llamado a la acción para la comunidad tecnológica global. Al revelar las vulnerabilidades dentro de los ecosistemas GenAI, "Morris II" subraya la importancia crítica de priorizar la seguridad en el diseño e implementación de tecnologías GenAI. Este estudio marca un paso crucial hacia la comprensión y mitigación de los riesgos asociados con la inteligencia artificial generativa, asegurando un futuro digital más seguro para todos. (Cohen, Bitton, & Nassi, 2024)

2.2.8. Polimorfismo con inteligencia artificial

Second Part To Hell en su repositorio de Github nos habla sobre LLMorpher un innovador malware que utiliza la inteligencia artificial para evadir los sistemas de detección.

Los virus informáticos tradicionalmente se escriben en código ejecutable de un lenguaje de programación. Sin embargo, un nuevo enfoque propuesto plantea codificar completamente los virus en lenguaje natural utilizando la poderosa inteligencia artificial GPT de OpenAI.

En lugar de instrucciones de código concretas, el virus consiste en una lista de oraciones bien definidas escritas en inglés. Luego, GPT traduce estas descripciones en inglés a código ejecutable. Debido a la ambigüedad inherente del lenguaje natural, GPT puede generar diferentes códigos con el mismo comportamiento, creando así una nueva forma de metamorfismo viral llamada "Lingüístico-Morfismo".

SPTH presenta dos ejemplos prácticos, "LLMorphism I" y "LLMorphism II", que son los primeros virus codificados completamente en lenguaje natural explotando las capacidades de GPT. El proceso involucra crear listas de "prompts" o instrucciones en inglés que describen el

comportamiento deseado del virus. GPT luego traduce estas instrucciones a código ejecutable de Python. (Hell., 2023)

Además, GPT puede reformular las mismas instrucciones en inglés de diferentes maneras sin cambiar su significado, lo que permite una mutación lingüística del propio código viral. Esta capacidad de reformulación lingüística y generación de código variable abre la puerta a una nueva clase de virus informáticos que desdibujan la frontera entre lenguaje natural y de programación.

El artículo concluye que, si bien esta técnica aún es rudimentaria, el rápido avance de los modelos de lenguaje como GPT permitirá codificar ideas cada vez más complejas directamente en lenguaje natural. Esta intersección de virus informáticos e inteligencia artificial plantea emocionantes y preocupantes perspectivas. (Hyppönen, 2023)

2.3.8. DeepDGA

Una Red Generativa Adversaria Profunda para Generación y Detección de Algoritmos de Generación de Dominios Maliciosos

Los algoritmos de generación de dominios (DGAs) son utilizados por muchas familias de malware para establecer conexiones de comando y control (C&C). Estos algoritmos generan de forma pseudoaleatoria grandes cantidades de nombres de dominio diariamente, dificultando las contramedidas defensivas. Si bien existen varios métodos para detectar dominios generados por DGAs, muchos se basan en conjuntos de datos limitados y pueden ser eludidos por nuevas variantes de DGA.

En el artículo DeepDGA: Adversarially-Tuned Domain Generation and Detection, los investigadores proponen DeepDGA, un novedoso marco de aprendizaje profundo que utiliza redes generativas adversarias (GANs) para generar nombres de dominio adversarios diseñados específicamente para evadir detectores de DGA basados en aprendizaje automático. DeepDGA consta de un modelo generador y un modelo discriminador que compiten en rondas adversarias. El generador aprende a crear nombres de dominio artificiales que son difíciles de distinguir de los dominios reales, mientras que el discriminador intenta identificar los dominios generados versus los reales.

Los experimentos muestran que DeepDGA puede producir nombres de dominio que eluden significativamente la detección por parte de un clasificador de bosque aleatorio entrenado con características de dominio creadas manualmente. Además, al aumentar el conjunto de

entrenamiento con los ejemplos adversarios generados por DeepDGA, el clasificador de bosque aleatorio se endureció contra familias de DGA nunca antes vistas, mejorando su capacidad de detección.

Esta investigación demuestra el uso novedoso de GANs para generar ejemplos adversarios en el contexto de la detección de DGA, destacando tanto los desafíos como las oportunidades de esta técnica para mejorar la robustez de los sistemas de detección de malware basados en aprendizaje automático. (Anderson, Woodbridge, & Filar, 2016)

2.4.8. MaskDGA

Una técnica de evasión de caja negra contra clasificadores DGA y defensas adversarias

El artículo MaskDGA: A Black-box Evasion Technique Against DGA Classifiers and Adversarial Defenses, presenta MaskDGA, una técnica práctica de aprendizaje adversario que agrega perturbaciones a representaciones a nivel de carácter de nombres de dominio generados algorítmicamente (AGD) para evadir clasificadores DGA, sin necesidad de conocer la arquitectura o parámetros del clasificador.

MaskDGA entrena un modelo sustituto discriminativo en datos públicos de AGDs, genera nuevos AGDs, construye un mapa de saliencia de Jacobian (JSM) para cada AGD añadiendo perturbaciones a la mitad de los caracteres con mayores gradientes en el JSM. Esto produce nuevos nombres de dominio adversarios que evaden la detección.

La evaluación en cuatro clasificadores DGA recientes mostró que MaskDGA reduce la puntuación F1 promedio de 0.977 a 0.495. También se evaluó contra técnicas de defensa como reentrenamiento adversario y destilación, demostrando que MaskDGA puede mejorar la robustez pero que los clasificadores DGA idealmente deberían incorporar características adicionales más allá de las de nivel de carácter.

El artículo discute la implementación práctica de MaskDGA, que requiere cargar un modelo sustituto pre entrenado de ~10MB y realizar inferencia, lo cual es factible incluso en dispositivos embebidos. Los resultados sugieren que los clasificadores DGA actuales basados únicamente en características de nivel de carácter son vulnerables a ataques adversarios y se necesitan mecanismos de detección más robustos. (Sidi, Nadler, & Shabtai, 2019)

3. Diseño de metodología de análisis

Esta sección se adentra en el proceso de creación de una metodología diseñada específicamente para abordar los desafíos inherentes al análisis de malware en entornos controlados. Desde la identificación de requisitos clave hasta la definición de pasos y técnicas de análisis, esta sección ofrece una visión detallada del enfoque metodológico adoptado. Además, se discuten las consideraciones prácticas y teóricas que influyen en la configuración de la metodología, incluyendo la selección de herramientas y la evaluación de riesgos.

Se presenta una hoja de ruta detallada que servirá como guía para la implementación y ejecución de la metodología de análisis, sentando las bases para un análisis de malware eficaz y orientado a resultados en entornos controlados.

3.1. Fases del proyecto

Tabla 2: Descripción de fases del proyecto

Descripción de fases del proyecto	
Fase	**Descripción**
Fase de investigación	En esta etapa inicial, se llevó a cabo una investigación sobre los distintos aspectos relacionados con el análisis de malware en entornos controlados. Se recopilaron y revisaron literaturas especializadas. El objetivo principal fue comprender en profundidad el panorama actual de la detección y análisis de malware, así como identificar brechas y oportunidades para el desarrollo de una metodología avanzada y efectiva.
Fase de diseño e implementación	Una vez completada la fase de investigación, se procede al diseño detallado de la metodología de análisis. Con base en este análisis, se establecen los pasos y procedimientos necesarios para llevar a cabo el análisis de malware en entornos controlados. Además, se desarrollan y/o adaptan las herramientas y recursos técnicos requeridos para la implementación práctica de la metodología diseñada.
Fase de pruebas	Se requiere un software de virtualización confiable y robusto para crear y gestionar máquinas virtuales en el sandbox. Algunas opciones populares incluyen VMware, VirtualBox, o Hyper-V de Microsoft.
Fase de documentación	Finalmente, se procede a documentar de manera detallada todos los aspectos de la metodología de análisis desarrollada. Se elabora un informe completo que incluye la descripción de cada paso y técnica de análisis, así como las herramientas utilizadas y los resultados obtenidos durante las pruebas. Esta documentación servirá como referencia y guía para la implementación y replicación de la metodología en diferentes

	contextos y proyectos relacionados con la detección y análisis de malware en entornos controlados.

Fuente: Elaboración propia

3.2. Conceptos teóricos fundamentales

El diseño e implementación de entornos controlados para el análisis de malware requiere del conocimiento de una serie de conceptos teóricos fundamentales. Estos conceptos sirven como cimientos para desarrollar estrategias efectivas que permitan comprender y mitigar las amenazas cibernéticas. A continuación, se presentan algunos de los conceptos clave:

3.1.2. Virtualización

La virtualización es una técnica de software que permite crear versiones virtuales de recursos tecnológicos, como sistemas operativos, dispositivos de almacenamiento, redes o incluso hardware completo. Esto se logra mediante una capa de software llamada hipervisor o monitor de máquina virtual, que se ejecuta en un sistema anfitrión y permite la creación y gestión de máquinas virtuales. (Villar & Gómez)

La virtualización proporciona un entorno seguro y aislado para estudiar el comportamiento de programas maliciosos sin poner en riesgo el sistema anfitrión. Además, la virtualización facilita la automatización y escalabilidad del análisis de malware, permitiendo ejecutar múltiples muestras en paralelo en diferentes máquinas virtuales para un procesamiento más rápido.

3.2.2. Hipervisor o monitor de máquinas virtuales

Es la capa que permite crear y ejecutar máquinas virtuales en un sistema anfitrión. El hipervisor actúa como una capa de abstracción entre el hardware físico del sistema y los sistemas operativos invitados. (Fernandez, 2015)

Un hipervisor permite la creación y administración de máquinas virtuales aisladas y controladas para estudiar programas maliciosos de manera segura. Sus principales funciones incluyen el aislamiento de las máquinas virtuales para evitar la propagación del malware, la gestión eficiente

de los recursos de hardware, la capacidad de realizar instantáneas y revertir estados, la clonación de máquinas virtuales y la automatización de tareas.

3.3.2. Máquina virtual

Una máquina virtual (VM, por sus siglas en inglés Virtual Machine) es un entorno de sistema operativo virtualizado que se ejecuta sobre un sistema físico a través de un software de virtualización conocido como hipervisor. Básicamente, una máquina virtual es una simulación completa de un sistema de computadora que se comporta como si fuera una máquina física independiente, con su propio procesador virtual, memoria, almacenamiento, interfaces de red y dispositivos. (Villar & Gómez)

Las máquinas virtuales son esenciales por que proporcionan un entorno de prueba aislado, permiten la reversión a estados anteriores, posibilitan la creación de múltiples configuraciones para probar diferentes comportamientos de malware, facilitan el monitoreo y análisis del malware, así como el uso de herramientas forenses, y permiten la automatización del análisis de malware.

3.4.2. Sandbox

Un sandbox es un entorno aislado y controlado donde se ejecutan muestras de programas potencialmente maliciosos con fines de análisis y observación. Pueden ser implementados mediante diversas tecnologías, como máquinas virtuales, contenedores, entornos de prueba dedicados, entre otros. Su objetivo principal es proporcionar un entorno seguro y controlado para ejecutar código o programas potencialmente peligrosos sin comprometer la integridad del sistema principal. (Rivera Guevara R. P., 2018)

Los sandbox se utilizan ampliamente en centros de operaciones de seguridad, laboratorios de malware y empresas de seguridad para analizar de forma segura amenazas cibernéticas, comprender su comportamiento y desarrollar contramedidas efectivas. También son componentes clave en soluciones de prevención y detección de malware basadas en análisis de comportamiento.

3.5.2. Sistema operativo

Un sistema operativo es un conjunto esencial de programas informáticos que administra eficientemente los recursos de una computadora, actuando como intermediario entre el hardware

y las aplicaciones de usuario. Sus funciones principales incluyen la gestión de recursos como la memoria RAM y el almacenamiento, proporcionar una interfaz de usuario para interactuar con el sistema y ejecutar programas, administrar archivos y directorios, controlar la ejecución de procesos, garantizar la seguridad del sistema y los datos, y permitir la comunicación y el intercambio de recursos en redes.

3.6.2. Archivo ejecutable

Un archivo ejecutable es un tipo de archivo que contiene instrucciones en forma de código máquina que pueden ser directamente ejecutadas por un procesador o sistema operativo. Estos archivos se identifican por extensiones específicas según el sistema operativo, como .exe o .com en Windows, sin extensión específica o .app en macOS, y sin extensión en Linux y Unix. Cuando se ejecuta un archivo ejecutable, el sistema operativo carga el código máquina en la memoria y asigna los recursos necesarios para su ejecución. Los archivos ejecutables pueden ser de diferentes tipos, como aplicaciones completas, utilidades y herramientas, controladores de dispositivos o archivos de script. (Rivera Guevara R. , 2014)

3.3. Requisitos y consideraciones

La planificación juega un papel crucial en la creación de entornos seguros y eficientes para el análisis de malware y la realización de pruebas de seguridad. La correcta configuración de hardware y software, junto con medidas adecuadas de aislamiento y seguridad, son fundamentales para garantizar la integridad de los datos, proteger los sistemas y maximizar la efectividad de las investigaciones.

3.1.3. Requisitos de hardware

El hardware de un sandbox para el análisis de malware juega un papel crucial en la capacidad de ejecutar de manera eficiente y segura los procesos de análisis. La selección adecuada de hardware garantiza un rendimiento óptimo y la capacidad de manejar cargas de trabajo intensivas. A continuación, se detallan los requisitos de hardware recomendados para un sandbox de análisis de malware:

Tabla 3: Descripción de requisitos de hardware

Descripción de requisitos de hardware.	
Requisito	**Descripción**
CPU (Unidad Central de Procesamiento):	Se requiere una CPU potente y multi núcleo para manejar múltiples procesos simultáneamente durante el análisis de malware. Se recomienda una CPU de al menos cuatro núcleos con soporte para tecnologías de virtualización para ejecutar máquinas virtuales de manera eficiente.
Memoria RAM	La cantidad de RAM disponible afecta directamente la capacidad del sandbox para ejecutar procesos de análisis de malware. Se recomienda una cantidad mínima de 8 GB de RAM para un funcionamiento adecuado. Sin embargo, para cargas de trabajo más intensivas, como análisis de malware complejos o simultáneos, se pueden requerir cantidades mayores de RAM.
Almacenamiento	Se debe disponer de suficiente espacio de almacenamiento para alojar sistemas operativos de máquinas virtuales, muestras de malware, herramientas de análisis y datos generados durante el proceso de análisis, sistema de respaldo y restauración para proteger los datos y facilitar la recuperación en caso de fallas. Se recomienda al menos 512 GB de almacenamiento y utilizar unidades de estado sólido (SSD) en lugar de discos duros tradicionales (HDD) para mejorar el rendimiento y la velocidad de acceso a los datos.
Soporte de Virtualización	Es fundamental que el hardware tenga soporte para tecnologías de virtualización, como Intel VT-x o AMD-V, para ejecutar máquinas virtuales de manera eficiente y segura. La virtualización basada en hardware mejora el rendimiento y la seguridad del sandbox al proporcionar aislamiento de recursos entre las máquinas virtuales.
Red aislada	Una red aislada es una red virtual separada del entorno de red principal, diseñada específicamente para contener el malware y prevenir su propagación. Se logra mediante el uso de un switch o un enrutador virtuales que crea segmentos de red separados para simular diversos entornos de red dentro del sandbox. Esto permite aislar el malware del resto de la infraestructura de red, reduciendo así el riesgo de contaminación cruzada. Además, la red aislada proporciona la capacidad de monitorear y capturar el tráfico de red dentro del sandbox, lo que facilita el análisis del comportamiento del malware y la identificación de posibles patrones de comunicación maliciosa.

Fuente: Elaboración propia

3.2.3. Requisitos de software

Los requisitos de software son diversos y abarcan desde el sistema operativo hasta las herramientas de análisis específicas. A continuación, se detallan los requisitos de software recomendados para un sandbox de análisis de malware:

Tabla 4: Descripción de requisitos de software

Descripción de requisitos de software.	
Requisito	**Descripción**
Sistema Operativo	El sandbox debe estar basado en un sistema operativo estable y seguro que admita la ejecución de máquinas virtuales. Se recomienda utilizar una distribución de Linux o Mac para este propósito, ya que ofrecen flexibilidad y opciones de configuración avanzadas. Además, se recomienda que el sistema operativo anfitrión sea diferente del sistema operativo invitado el cual será el objetivo de análisis donde se ejecutaran las muestras de malware.
Plataforma de Virtualización	La cantidad de RAM disponible afecta directamente la capacidad del sandbox para ejecutar procesos de análisis de malware. Se recomienda una cantidad mínima de 8 GB de RAM para un funcionamiento adecuado. Sin embargo, para cargas de trabajo más intensivas, como análisis de malware complejos o simultáneos, se pueden requerir cantidades mayores de RAM.
Almacenamiento	Se requiere un software de virtualización confiable y robusto para crear y gestionar máquinas virtuales en el sandbox. Algunas opciones populares incluyen VMware, VirtualBox, o Hyper-V de Microsoft.
Herramientas de Análisis de Malware	El sandbox debe estar equipado con un conjunto completo de herramientas de análisis de malware que permitan examinar el comportamiento y las características del software malicioso. Esto incluye herramientas para análisis estático y dinámico, como IDA Pro, Wireshark, y Sysinternals Suite, entre otras.
Actualizaciones y Parches	Es importante mantener actualizado el software del sandbox, incluyendo el sistema operativo, las herramientas de análisis de malware, y las soluciones de seguridad. Aplicar parches de seguridad de manera regular ayuda a mitigar vulnerabilidades y proteger el entorno contra posibles amenazas.
Automatización	Se recomienda utilizar herramientas de automatización para simplificar y agilizar las tareas de análisis de malware. Esto puede incluir el uso de scripts y herramientas de administración remota para ejecutar análisis de manera eficiente y coordinada.

Capacidades de Captura y Registro	El sandbox debe tener la capacidad de capturar y registrar datos de manera efectiva durante el análisis de malware. Esto puede incluir capturas de pantalla, registros de actividad del sistema, tráfico de red, análisis de memoria y cualquier otra información relevante para el análisis y la investigación.

Fuente: Elaboración propia

3.3.3. Consideraciones de aislamiento y seguridad

El aislamiento y la seguridad son aspectos críticos en el diseño y la implementación de un sandbox para el análisis de malware. Estas consideraciones aseguran que el entorno de análisis sea capaz de ejecutar y estudiar el software malicioso de manera controlada, sin comprometer la integridad de los sistemas circundantes. A continuación, se detallan las principales consideraciones de aislamiento y seguridad que deben tenerse en cuenta:

Tabla 5: Descripción de consideraciones de aislamiento y seguridad

Descripción de consideraciones de aislamiento y seguridad.	
Consideración	**Descripción**
Aislamiento físico	El sandbox debe estar físicamente aislado del resto de la infraestructura de TI de la organización o de su red de trabajo personal, ubicarlo en un área restringida y segura, con acceso controlado y monitorizado.
Aislamiento de red	Crear una red virtual aislada y desconectada de la red de trabajo o red personal para el sandbox, utilizar una red de área local virtual (VLAN) dedicada o una red física completamente separada, implementar firewalls y dispositivos de seguridad perimetral para controlar y monitorear el tráfico entrante y saliente del sandbox.
Virtualización segura	Utilizar un hipervisor de virtualización seguro y actualizado, configurar el hipervisor para limitar los recursos asignados a las máquinas virtuales y evitar la fuga de malware, Implementar mecanismos de instantáneas y restauración rápida de máquinas virtuales para facilitar el análisis y la recuperación.
Configuración de máquinas virtuales	Utilizar imágenes de máquinas virtuales configuradas de forma segura y con las últimas actualizaciones., deshabilitar servicios y funciones innecesarias en las máquinas virtuales para reducir la superficie de ataque, implementar mecanismos de monitoreo y registro de actividades dentro de las máquinas virtuales.
Gestión de muestras de malware	Establecer procedimientos estrictos para el manejo y almacenamiento de muestras de malware, almacenar las muestras de malware en un sistema de archivos cifrado y con acceso restringido, implementar mecanismos de destrucción segura de muestras de malware después del análisis.

Controles de acceso y autenticación	Implementar controles de acceso basados en roles para limitar el acceso al sandbox solo al personal autorizado.
Gestión de actualizaciones y parches	Mantener actualizados el sistema operativo, el hipervisor y todas las herramientas de software utilizadas en el sandbox, aplicar parches de seguridad de forma oportuna para mitigar vulnerabilidades conocidas.
Plan de respuesta a incidentes	Desarrollar e implementar un plan de respuesta a incidentes en caso de fuga de malware o compromisos de seguridad, establecer procedimientos para contener, erradicar y recuperarse de incidentes de seguridad, realizar pruebas periódicas del plan de respuesta a incidentes.

Fuente: Elaboración propia

3.4. Herramientas y recursos

En esta sección, detallaremos las herramientas y recursos disponibles para facilitar la implementación efectiva de un Sandbox para el análisis de malware. Descubriremos las capacidades, ventajas y consideraciones clave de estas herramientas.

3.1.4. Software de virtualización

Oracle VM VirtualBox

Oracle VM VirtualBox, el software de virtualización multiplataforma de código abierto más popular del mundo, permite a los desarrolladores entregar código más rápido, ya que pueden ejecutar múltiples sistemas operativos en un solo dispositivo. Los equipos de TI y los proveedores de soluciones usan VirtualBox para reducir los costos operativos y acortar el tiempo necesario para implementar aplicaciones de forma segura en entornos locales y en la nube (ORACLE.COM).

Usar VirtualBox para implementar un Sandbox destinado al análisis de malware ofrece varias ventajas. En primer lugar, su interfaz intuitiva y fácil de usar simplifica el proceso de configuración, lo que resulta especialmente útil para usuarios con poca experiencia en virtualización. Además, VirtualBox es compatible con una amplia gama de sistemas operativos invitados, lo que permite crear entornos de Sandbox personalizados para adaptarse a las necesidades específicas de análisis de malware.

Otra ventaja es su flexibilidad en términos de configuración de red. VirtualBox ofrece opciones avanzadas que permiten simular diferentes escenarios de red, lo que es crucial para estudiar el comportamiento del malware en entornos de red específicos. Además, la capacidad de crear

instantáneas y clonar máquinas virtuales facilita la gestión de múltiples entornos de Sandbox y permite revertir rápidamente a estados anteriores en caso de problemas durante el análisis.

VirtualBox también ofrece la posibilidad de asignar recursos de hardware específicos a las máquinas virtuales según sea necesario, lo que permite simular diferentes configuraciones de hardware y optimizar el rendimiento del Sandbox. Por último, la amplia comunidad de usuarios y la documentación detallada de VirtualBox proporcionan soporte adicional y recursos de resolución de problemas, lo que hace que sea una opción sólida para implementar un Sandbox para análisis de malware.

3.2.4. Sistemas Operativos

Los sistemas operativos utilizados para implementar un sandbox destinado al análisis de malware son una parte fundamental en la construcción de un entorno seguro y controlado para estudiar el comportamiento de programas maliciosos. Estos sistemas, como Ubuntu, REMnux o Windows 10 Enterprise, ofrecen características específicas que los hacen ideales para este propósito. Desde la seguridad avanzada hasta las opciones de gestión y control exhaustivas, cada sistema operativo aporta su conjunto único de herramientas y capacidades para garantizar un análisis efectivo y seguro del malware. En esta introducción, exploraremos las ventajas y consideraciones clave de los sistemas operativos seleccionados para implementar un sandbox de análisis de malware.

Arch Linux

Arch Linux es una distribución de GNU/Linux desarrollada de manera independiente, orientada a la arquitectura x86-64 y de propósito general, que busca proporcionar las últimas versiones estables del software mediante un modelo de lanzamiento continuo. Se instala por defecto como un sistema base mínimo, permitiendo al usuario agregar solo lo necesario (wiki.archlinux.org). Arch Linux ofrece una combinación única de flexibilidad, control, disponibilidad de software y comunidad activa que lo hace una elección sólida como sistema operativo principal en un entorno de Sandbox para análisis de malware.

Ubuntu Desktop

Ubuntu es un sistema operativo Linux basado en Debian que se ejecuta desde el escritorio hasta la nube, abarcando todos tus dispositivos conectados a Internet. Es el sistema operativo más popular del mundo en nubes públicas y en nubes basadas en OpenStack. Es la plataforma

número uno para contenedores; desde Docker hasta Kubernetes y LXD, Ubuntu puede ejecutar tus contenedores a gran escala. Rápido, seguro y simple, Ubuntu alimenta millones de PCs en todo el mundo.

El desarrollo de Ubuntu está liderado por Canonical Ltd. Canonical genera ingresos a través de la venta de soporte técnico y otros servicios relacionados con Ubuntu. El proyecto Ubuntu está comprometido públicamente con los principios del desarrollo de software de código abierto; se anima a las personas a utilizar software libre, estudiar cómo funciona, mejorarlo y distribuirlo. (docker.com).

Ubuntu es altamente personalizable y ofrece una amplia gama de herramientas y aplicaciones que pueden ser útiles para el análisis de malware. Además, es compatible con una amplia gama de herramientas de análisis de malware y software de seguridad, lo que facilita la configuración de un entorno de análisis completo dentro de la máquina virtual. Ubuntu cuenta con una gran comunidad de usuarios y desarrolladores que pueden proporcionar soporte técnico y recursos adicionales para ayudar en la configuración y el uso del sandbox de análisis de malware.

REMnux

REMnux es un conjunto de herramientas de Linux para el análisis y la ingeniería inversa de software malicioso. Proporciona una colección cuidadosamente seleccionada de herramientas gratuitas creadas por la comunidad. Los analistas pueden utilizar REMnux para investigar malware sin tener que buscar, instalar y configurar las herramientas por separado(remnux.org). Es una distribución especializada de Linux creada por Lenny Zeltser para análisis de malware y respuesta a incidentes. Se distribuye como una imagen de máquina virtual para plataformas como VMware y VirtualBox. Incluye una amplia gama de herramientas de código abierto y comerciales para analizar malware y realizar investigaciones forenses. Está preconfigurado con herramientas como disassemblers, debuggers, análisis de tráfico de red, documentos maliciosos, emuladores de malware y análisis de memoria, junto con scripts y recursos para automatizar tareas de análisis y respuesta a incidentes.

Windows 10 Enterprise Trial

Windows 10 Enterprise está diseñado para satisfacer las necesidades de organizaciones grandes y medianas, proporcionando a los profesionales de TI: Protección avanzada contra amenazas de seguridad modernas, Opciones flexibles de implementación, actualización y soporte, Gestión completa de dispositivos y aplicaciones, así como control sobre ellos. Windows 10, versión 21H2, facilita la protección de los puntos finales, la detección de ataques avanzados, la automatización

de la respuesta a amenazas emergentes y la mejora de la postura de seguridad. También ayuda a agilizar la implementación y las actualizaciones, y a entregar dispositivos listos para la empresa directamente desde el fabricante a los usuarios. Este software de evaluación está diseñado para profesionales de TI interesados en probar Windows 10 Enterprise en nombre de su organización. No recomendamos instalar esta evaluación si no eres un profesional de TI o no estás gestionando profesionalmente redes corporativas o dispositivos (microsoft.com).

3.3.4. Herramientas para análisis

Cuckoo Sandbox

Es un software de código abierto para automatizar el análisis de archivos sospechosos. Para ello, utiliza componentes personalizados que monitorean el comportamiento de los procesos maliciosos mientras se ejecutan en un entorno aislado (cuckoosandbox.org).

Cuckoo Sandbox ofrece una variedad de ventajas para el análisis de malware. En primer lugar, su capacidad para automatizar gran parte del proceso de análisis ahorra tiempo y esfuerzo a los investigadores. Al ejecutar el malware en un entorno aislado y controlado, Cuckoo permite observar su comportamiento sin riesgo para el sistema principal, lo que facilita la comprensión de su funcionalidad y la identificación de posibles amenazas. Además, la generación de informes detallados proporciona una visión completa del comportamiento del malware, lo que ayuda a los analistas a tomar decisiones informadas sobre cómo mitigar la amenaza. La capacidad de integrarse con otras herramientas y servicios también es una ventaja significativa, ya que permite aprovechar recursos adicionales para mejorar el análisis.

VirusTotal API

Permite cargar y escanear archivos, enviar y escanear URLs, acceder a informes de escaneo terminados y realizar comentarios automáticos en URLs y muestras sin necesidad de utilizar la interfaz web. En otras palabras, te permite construir scripts simples para acceder a la información generada por VirusTotal (virustotal.com).

Yara

YARA es una herramienta versátil diseñada principalmente para ayudar a los investigadores de malware a identificar y clasificar muestras de malware. Funciona creando reglas que describen características del malware, como patrones textuales o binarios. Estas reglas, que contienen

cadenas y expresiones booleanas, pueden personalizarse para detectar familias específicas de malware o comportamientos particulares (virustotal).

MISP

MISP (Malware Information Sharing Platform & Threat Sharing) es una solución de software de código abierto para recopilar, almacenar, distribuir y compartir indicadores de seguridad cibernética y amenazas sobre análisis de incidentes de seguridad cibernética y análisis de malware. MISP está diseñado por y para analistas de incidentes, profesionales de seguridad y TIC para respaldar sus operaciones diarias para compartir información estructurada de manera eficiente. (misp-project)

3.5. Instalación y configuración del entorno controlado

En esta sección, se desarrolla el proceso de instalación y configuración del entorno controlado, un aspecto fundamental en el ámbito de la ciberseguridad y el análisis de malware. Durante esta sección, se explora desde la instalación de máquinas virtuales hasta la configuración de redes aisladas y la implementación de medidas de seguridad adicionales. Se muestra cómo configurar adecuadamente el entorno para maximizar la eficacia del análisis de malware y minimizar los riesgos asociados con la manipulación de muestras maliciosas.

3.1.5. Sistema Operativo Anfitrión

Tabla 6: Detalles del hardware del sistema operativo anfitrión

Tipo	Detalles
CPU	AMD Ryzen 7 5800X 8-Core
Memoria RAM	2 x 16 GB Módulos.
Disco Duro	SSD ADATA SU750 512 GB
BIOS	B550M Pro4

Fuente: Elaboración propia

Descarga del Archivo ISO de Arch Linux

- Desde la URL https://geo.mirror.pkgbuild.com/iso/2024.02.01, descargar el archivo archlinux-2024.02.01-x86_64.iso.
- Descarga también el archivo sha256sums.txt y verificar la integridad del archivo ISO descargado.

Creación de una USB de Arranque

- Emplear la herramienta Rufus-4.4 para crear una unidad USB de arranque con el archivo ISO de Arch Linux.

Instalación del Sistema Operativo

- Durante el proceso de instalación, simplificar la tarea utilizando la herramienta archinstall. Antes de comenzar, actualizar la herramienta con el comando: sudo pacman -Sy archinstall
- Durante el proceso, elegir el entorno gráfico KDE y habilitar el repositorio multilib.
- Después de la instalación, actualizar el sistema con el comando: sudo pacman -Syu

Instalación de VirtualBox

- Instalar VirtualBox y los módulos necesarios con los siguientes comandos: sudo pacman -S virtualbox-host-modules-arch y sudo pacman -S virtualbox

Tabla 7: Detalles del sistema operativo anfitrión

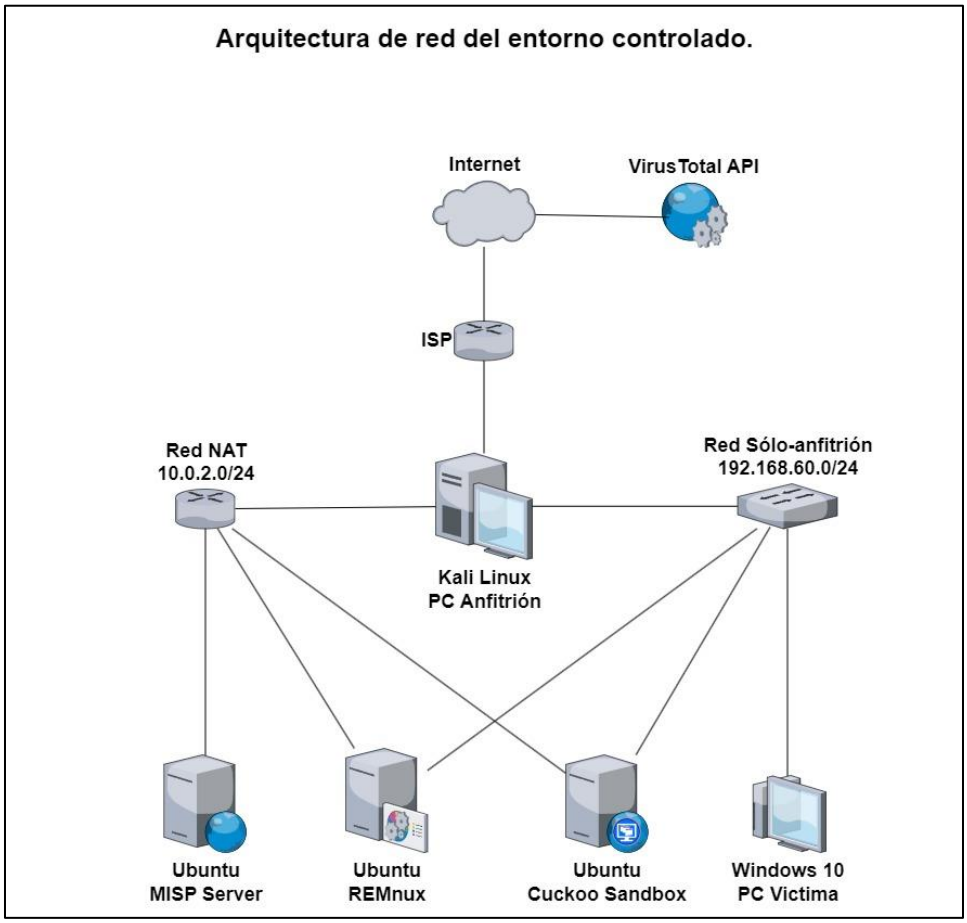
Tipo	Detalle
Operating System	Arch Linux
KDE Plasma Version	6.0.2
KDE Frameworks Version	6.0.0
Kernel Version	6.8.2-arch2-1 (64-bit)
Graphics Platform	Wayland
Processors	16 × AMD Ryzen 7 5800X 8-Core Processor
Memory	31.3 GiB de RAM
Graphics Processor	NV196
Product Name	B550M Pro4

Fuente: Elaboración propia

3.2.5. Configuración de red virtual

Para las máquinas virtuales se utilizaron dos redes virtuales, una red solo anfitriona con IP 192.168.60.0/24 para la conectividad con la maquina victima con las máquinas de análisis y una red NAT con IP 10.0.2.0/24 con acceso a internet para poder hacer uso del servicio de VirusTotal API.

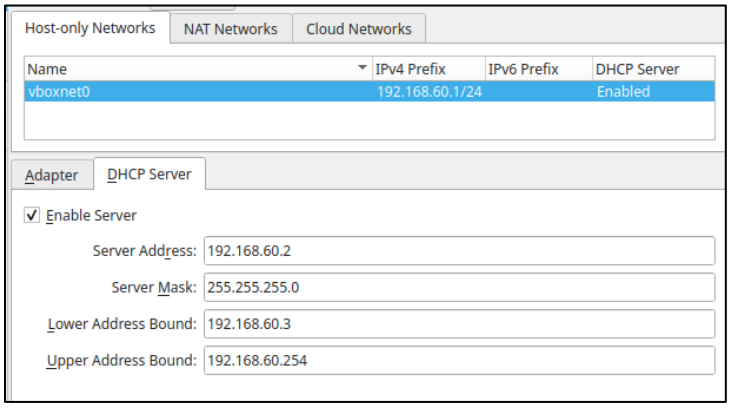
Ilustración 5: Diseño de estructura de red del entorno controlado



Fuente: Elaboración propia

En las herramientas de red de virtualbox, se creó la red solo anfitriona y se configuro el servidor de DHCP

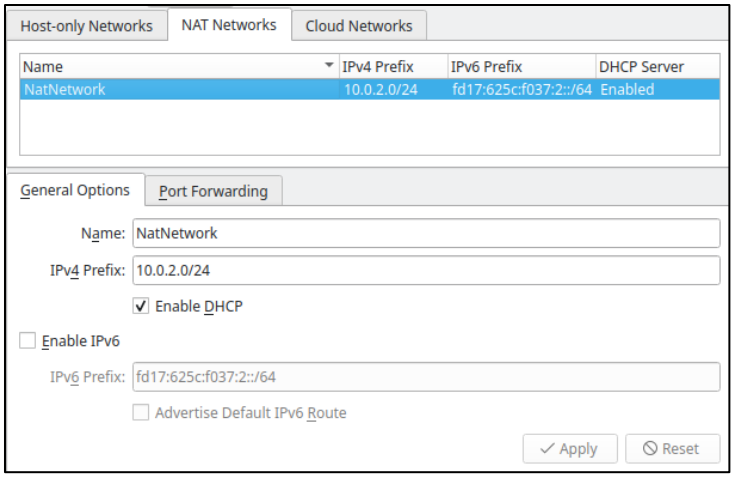
Ilustración 6: Detalles de la red solo anfitrión



Fuente: captura de pantalla

En las herramientas de red de virtualbox, también se creó la red NAT

Ilustración 7: Detalles de la red NAT



Fuente: captura de pantalla

3.3.5. Instalación y configuración de maquina víctima

Descarga del Archivo ISO

- Abrir un navegador web y acceder a la URL proporcionada para descargar el archivo ISO de Windows 10 Enterprise: https://www.microsoft.com/en-us/evalcenter/download-windows-10-enterprise.
- Seleccionar la versión de 64 bits.

Verificación de la Integridad del Archivo Descargado

- Abrir una terminal en el sistema operativo.
- Navegar hasta el directorio donde se encuentra el archivo ISO descargado, o copiar el archivo en un directorio accesible desde la terminal.
- Ejecutar el siguiente comando para verificar la integridad del archivo descargado utilizando su hash SHA256: sha256sum.
- Comparar el hash SHA256 resultante con el proporcionado en el texto. Si coinciden, significa que el archivo se ha descargado correctamente y no se ha corrompido durante el proceso de descarga.

Configuración de VirtualBox

- Iniciar el asistente de creación de máquinas virtuales.
- Asignar un nombre a la máquina virtual, elegir "Windows" como tipo y "Windows 10 (64-bit)" como versión.
- Asignar la cantidad de memoria RAM que se desea dedicar a la máquina virtual. Se recomienda al menos 4 GB para Windows 10.
- En el asistente de creación de discos virtuales elegir el tipo de archivo de disco duro virtual (VDI es el predeterminado).
- Seleccionar la opción de tamaño dinámico para el tamaño del disco duro y especificar el tamaño deseado. Se recomienda al menos 20 GB para Windows 10.
- Finalizar la creación de la máquina virtual.
- Seleccionar la máquina virtual recién creada en la lista de VirtualBox y hacer clic en "Configuración".
- En la sección "Almacenamiento", seleccionar el controlador de almacenamiento vacío SATA y hacer clic en el ícono del disco al lado.
- Seleccionar "Elegir un archivo de disco óptico" y navegar hasta el archivo ISO de Windows 10 descargado anteriormente.
- Hacer clic en "Abrir" para adjuntar el archivo ISO a la unidad óptica virtual de la máquina virtual.
- En la sección "Red", seleccionar el adaptador solo anfitrión y seleccionar la red virtual que previamente fue creada.

Instalación de Windows 10 en la Máquina Virtual

- Iniciar la máquina virtual.
- La máquina virtual arrancará desde el archivo ISO de Windows 10. Seguir las instrucciones en pantalla para instalar Windows 10 en la máquina virtual.

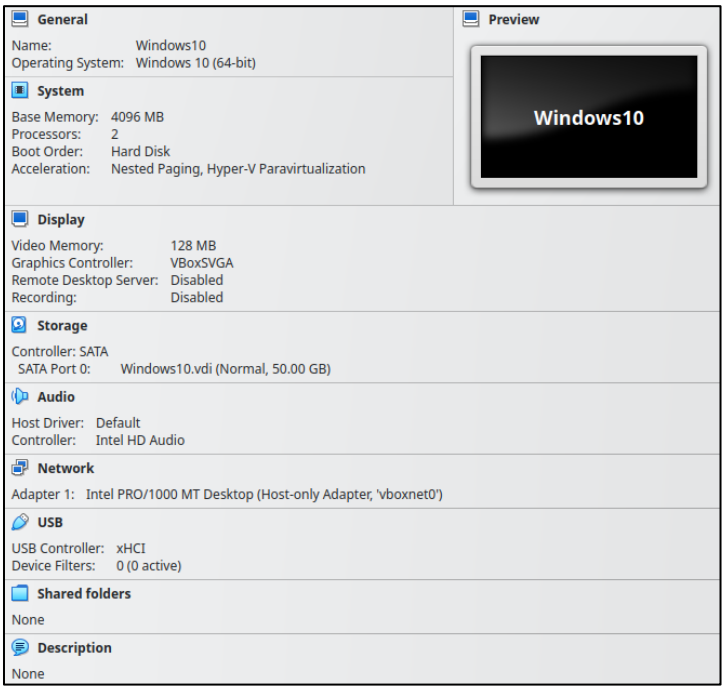
Tabla 8: Detalles del sistema operativo víctima

Tipo	Detalle
Operating System	Windows 10 Enterprise Evaluation
Version	22H2
OS Build	19045.2006
Device Name	WIN10
Processors	AMD Ryzen 7 5800X 8-Core Processor
Memory	4 GiB de RAM

System Type	X64-based
Product Name	B550M Pro4

Fuente: Elaboración propia

Ilustración 8: Detalles de configuración de máquina virtual víctima

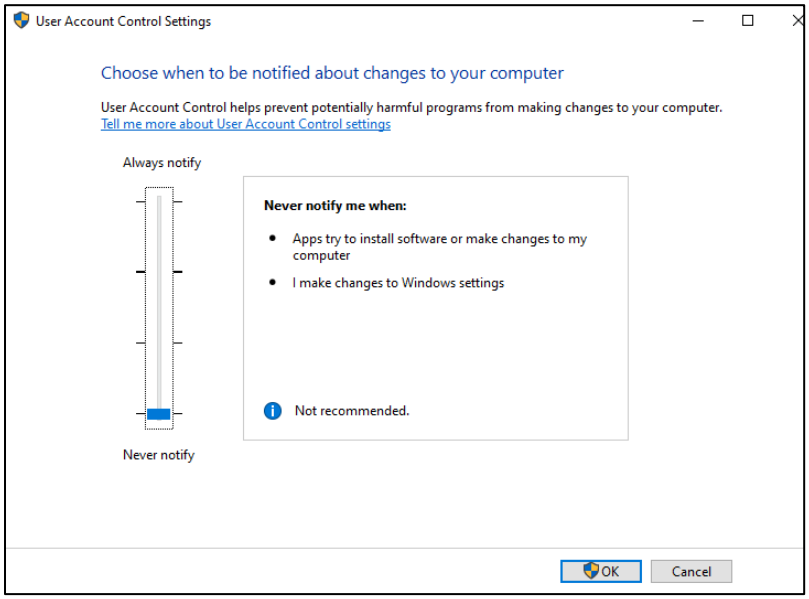


Fuente: Captura de pantalla

Deshabilitar UAC en Windows 10

- En la ventana de "User Account Control Setting", hay control deslizante con diferentes niveles de seguridad.
- Mover el control deslizante hacia abajo para reducir la seguridad y deshabilitar las notificaciones de UAC.

Ilustración 9: Pantalla de configuración de UAC



Fuente: Captura de pantalla de Windows

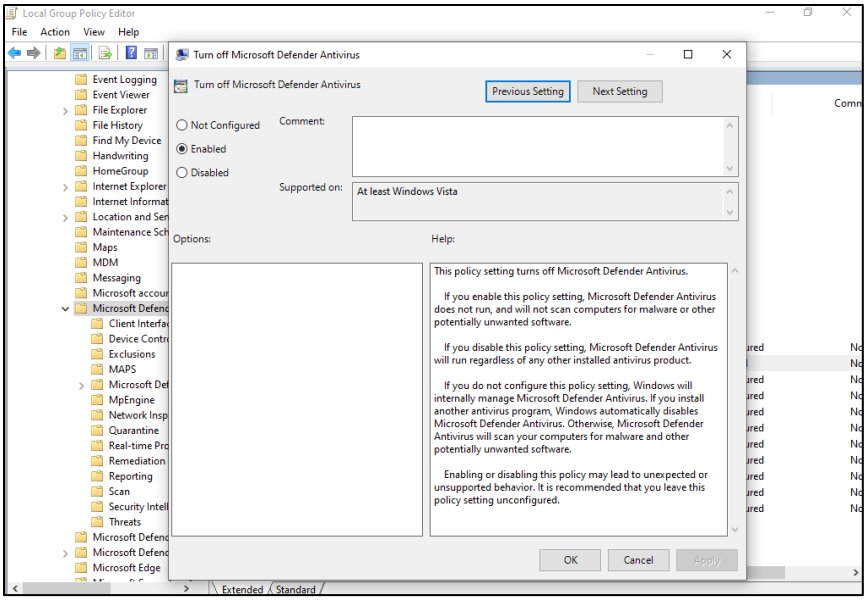
Deshabilitar la Seguridad en Windows

- Acceder a la Configuración de Seguridad.
- Deshabilitar todas las opciones de seguridad.

Deshabilitar Microsoft Defender Antivirus usando el Editor de Directivas de Grupo Local

- Presionar las teclas "Windows + R" en el teclado para abrir el cuadro de diálogo "Ejecutar".
- Ingresar "gpedit.msc" en el cuadro de diálogo para abrir el Editor de Directivas de Grupo Local.
- En el Editor de Directivas de Grupo Local, navegar a la siguiente ruta: Administrative Templates -> Windows Components -> Microsoft Defender Antivirus
- En la carpeta "Microsoft Defender Antivirus", buscar y hacer doble clic en la opción llamada " Turn off Microsoft Defender Antivirus".
- En la ventana de configuración, seleccionar la opción "Enable" y hacer clic en "Apply" para guardar los cambios.

Ilustración 10: Pantalla de des habilitación de Microsoft Defender Antivirus

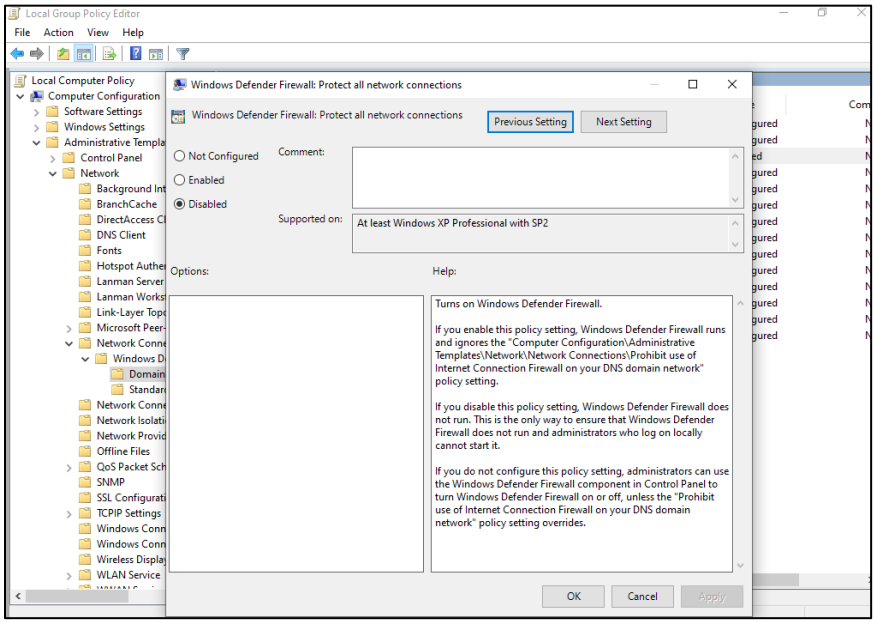


Fuente: Captura de pantalla de Windows

Deshabilitar el Firewall de Windows usando el Editor de Directivas de Grupo Local

- Presionar las teclas "Windows + R" en el teclado para abrir el cuadro de diálogo "Ejecutar".
- Ingresar "gpedit.msc" en el cuadro de diálogo para abrir el Editor de Directivas de Grupo Local.
- En el Editor de Directivas de Grupo Local, navegar a la siguiente ruta: Administrative Templates -> Network -> Network Connections -> Windows Defender Firewall -> Domain Profile
- Deshabilitar la Opción para “Protect All Network Connections”.
- En la ventana de configuración, seleccionar la opción "Disable" y hacer clic en "Apply" para guardar los cambios.

Ilustración 11: Pantalla de des habilitación del Firewall de Windows



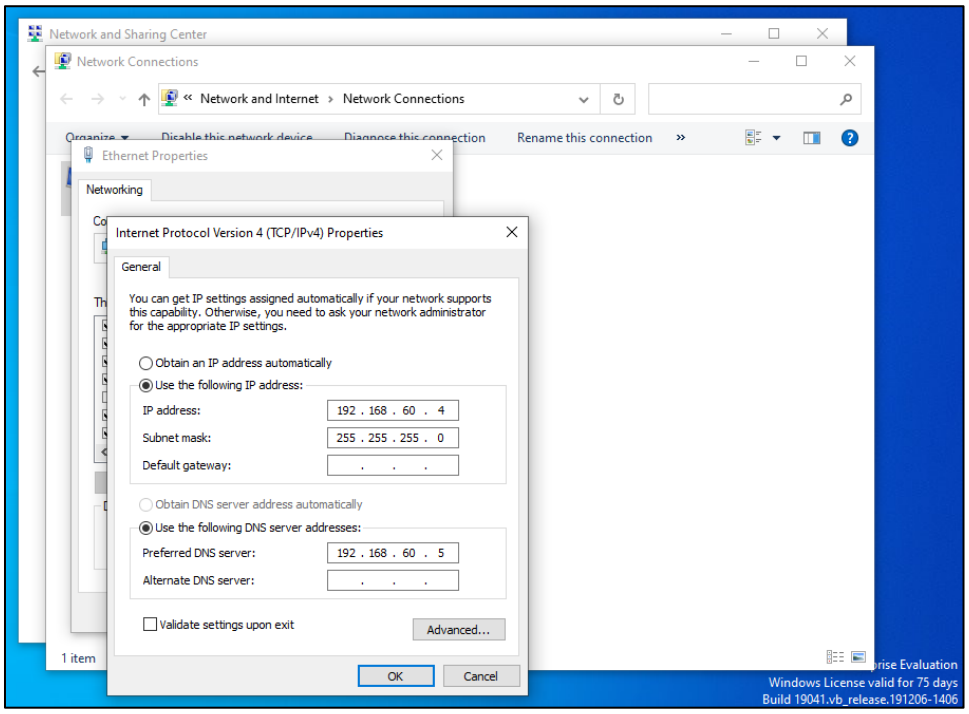
Fuenta: Captura de pantalla de windows

Asignar la IP y el DNS en Windows 10

- Abrir el Panel de control.
- Navegar hasta Panel de control\Redes e Internet\Centro de redes y recursos compartidos.
- Hacer clic en "Cambiar configuración del adaptador".
- Se abrirá una nueva ventana con la lista de adaptadores de red disponibles. Hacer clic derecho en la conexión de red que deseamos configurar y selecciona "Propiedades".
- En la ventana de propiedades de la conexión de red, desplazar hacia abajo hasta encontrar "Protocolo de Internet versión 4 (TCP/IPv4)" en la lista de elementos, seleccionar y luego hacer clic en "Propiedades".
- En la ventana de propiedades de TCP/IPv4, seleccionar la opción "Usar la siguiente dirección IP".
- Ingresar la dirección IP deseada en el campo "Dirección IP". En este caso, ingresar "192.168.60.4".
- Ingresar la máscara de subred en el campo correspondiente. Si está utilizando una máscara de subred estándar de clase C, será "255.255.255.0".
- Hacer clic en "Aceptar" para guardar los cambios.
- En la misma ventana de propiedades de TCP/IPv4, seleccionar la opción "Usar las siguientes direcciones de servidor DNS".

- Ingresar la dirección IP del servidor DNS en el primer campo. En este caso, ingresar "192.168.60.5" que es la IP donde se instaló el servicio de DNS falso.
- Clic en "Aceptar" para guardar los cambios y cerrar la ventana.

Ilustración 12: Ventana de configuración de IP y DNS en Windows 10



Fuente: Captura de pantalla de Windows

Instalación de Python 2.7

- Abrir un navegador web y acceder a la URL https://www.python.org/downloads/release/python-2710/.
- Descarga el archivo python-2.7.10.msi haciendo clic en el enlace correspondiente.
- Una vez descargado, hacer doble clic en el archivo .msi descargado para iniciar el instalador de Python 2.7.
- Seguir las instrucciones del instalador para completar la instalación de Python 2.7. Asegúrese de marcar la casilla que dice "Agregar Python al PATH" durante la instalación para que Python se agregue automáticamente a tu variable de entorno PATH.

Instalación de Pillow

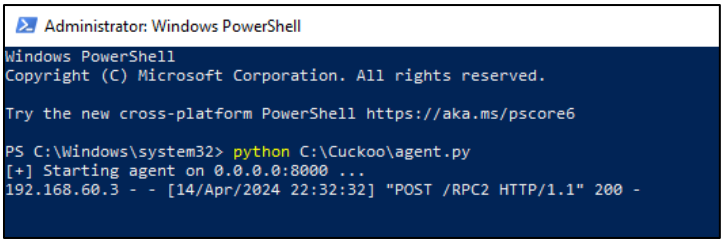
- Abrir un navegador web y accede a la URL https://pypi.python.org/packages/2.7/P/Pillow/Pillow-2.9.0.win32-py2.7.exe.
- Descargar el archivo Pillow-2.9.0.win32-py2.7.exe haciendo clic en el enlace de descarga.

- Una vez descargado, hacer doble clic en el archivo .exe descargado para iniciar el instalador de Pillow.
- Seguir las instrucciones del instalador para completar la instalación de Pillow. Asegurarse de seleccionar la opción que instale Pillow para Python 2.7 durante el proceso de instalación.

Descarga y Configuración del Cuckoo Agent.py

- Descarga el código fuente de Cuckoo Sandbox Agent desde el repositorio oficial en GitHub: https://github.com/cuckoosandbox/cuckoo/blob/2.0-rc2/agent/agent.py
- Crear una carpeta llamada Cockoo en el disco local C y ahí colocar el script agent.py.
- Abrir una consola con privilegios de administrador y navegar hasta el directorio donde se encuentra agent.py.
- Ejecutar el siguiente comando para iniciar el agente Cuckoo: python agent.py

Ilustración 13: Ejecución de agente de cockoo sandbox en Windows 10



Fuente: Captura de pantalla de Windows

Una vez terminada la instalación y configuración de la maquina víctima, cerrar todo, apagar y crear una instantánea.

3.4.5. Instalación y configuración de máquina virtual para análisis estático

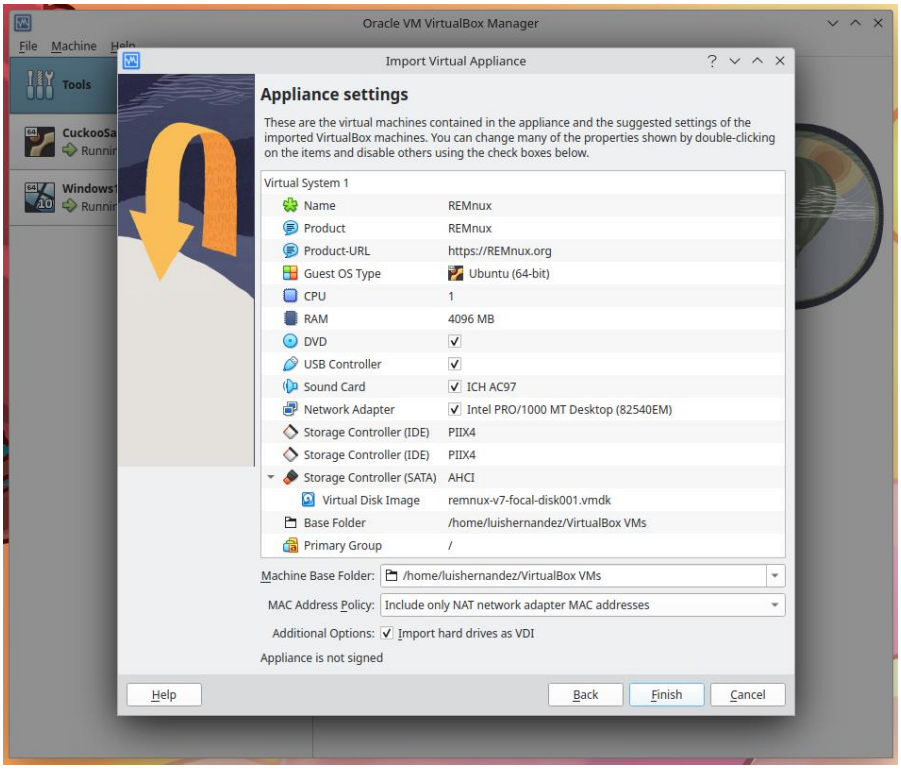
Instalación de REMnux OVA en VirtualBox

Descargar el Archivo OVA de REMnux

- Abrir un navegador web y accede al sitio web oficial de REMnux: https://remnux.org/.
- Ir a la sección de descargas y descargar la versión más reciente de REMnux disponible como archivo OVA.
- Abrir VirtualBox.
- En la barra de menú, ir a "Archivo" y seleccionar "Importar servicio virtualizado".

- Seleccionar el archivo OVA de REMnux que se descargó anteriormente.
- Hacer clic en "Siguiente" y luego en "Importar" para iniciar el proceso de importación.
- En la sección "Red", seleccionar el adaptador solo anfitrión y seleccionar la red virtual que previamente se creó, además, agregar un segundo adaptador de red y seleccionar la red NAT que previamente se configuro.
- Iniciar la Máquina Virtual de REMnux.

Ilustración 14: Detalles de la configuración de máquina de análisis estático

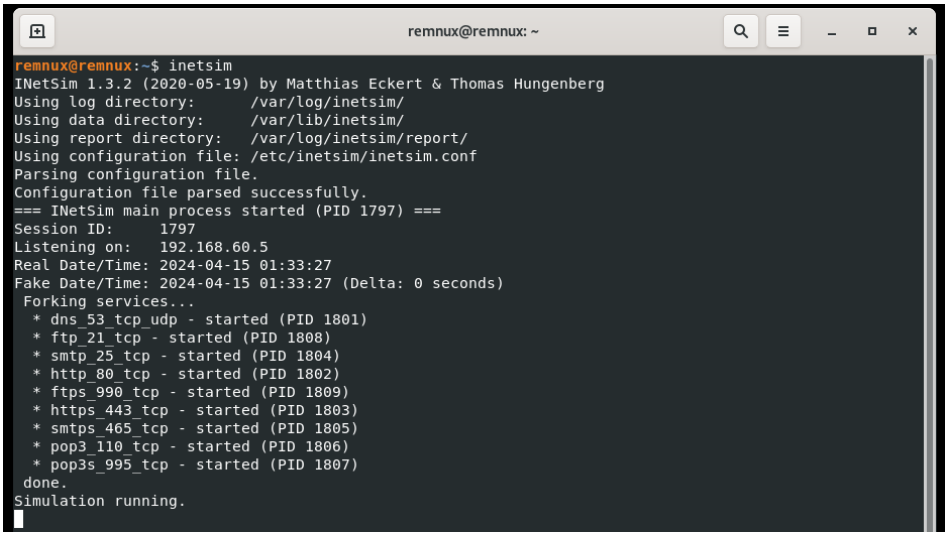


Fuente: Captura de pantalla

Configuración de DNS falso con Inetsim

- Abrir el archivo de configuración de inetsim con el comando: sudo nano /etc/inetsim/inetsim.conf
- Buscar la línea # start_service dns y remover el símbolo de numeral (#) para habilitar el servicio de DNS falso.
- Iniciar el servicio de inetsim.

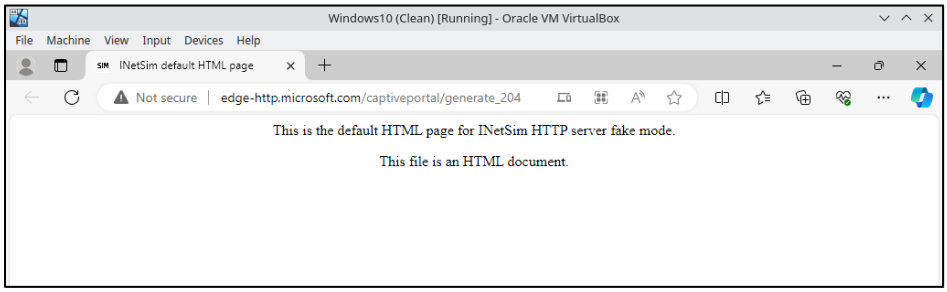
Ilustración 15: Consola de remnux con el servicio de inetsim corriendo



Fuente: Captura de pantalla

- Verificar en la maquina victima que esta consulta el DNS falso, que previamente se había configurado con la IP 192.168.60.5

Ilustración 16: Maquina victima consulta DNS falso con inetsim



Fuente: Captura de pantalla

Una vez terminada la instalación y configuración de la máquina, cerrar todo, apagar y crear una instantánea.

3.5.5. Instalación y configuración de máquina virtual de análisis dinámico

Instalación de Ubuntu Desktop 22.04 en VirtualBox

- Abrir un navegador web y acceder al sitio web oficial de Ubuntu: https://ubuntu.com/download/desktop.
- Descargar la imagen de disco (archivo ISO) de Ubuntu Desktop 22.04 LTS.
- Abrir VirtualBox.

- Hacer clic en el botón "Nueva" en la barra de herramientas para crear una nueva máquina virtual.
- Asignar un nombre a la máquina virtual.
- Seleccionar "Linux" como tipo de sistema operativo.
- Seleccionar "Ubuntu (64-bit)" como versión del sistema operativo.
- Asignar la cantidad de memoria RAM que desea dedicar a la máquina virtual. Se recomienda al menos 4 GB para Ubuntu Desktop en el cual se instalará Cuckoo Sandbox.
- Seleccionar el tipo de disco duro virtual (VDI es una opción común).
- Asignar el tamaño del disco duro virtual. Se recomienda al menos 150 GB para Ubuntu Desktop y los archivos de Cuckoo Sandbox.
- Finalizar la creación de la máquina virtual.
- Una vez que se haya creado la máquina virtual, hacer clic derecho sobre ella y selecciona "Configuración".
- Ir a la pestaña "Almacenamiento".
- En el controlador "Controlador IDE", seleccionar el icono de disco óptico vacío y elegir "Seleccionar un archivo de disco óptico virtual".
- Seleccionar la imagen de disco (archivo ISO) de Ubuntu Desktop 22.04 que se descargó anteriormente.
- En la sección "Red", seleccionar el adaptador solo anfitrión y seleccionar la red virtual que previamente se creó, además, agregar un segundo adaptador de red y seleccionar la red NAT que previamente se había configurado.
- Hacer clic en "Aceptar" para guardar los cambios.
- Inicia la máquina virtual haciendo clic en el botón "Iniciar".
- La máquina virtual arrancará desde la imagen de disco de Ubuntu Desktop.
- Seguir las instrucciones en pantalla para instalar Ubuntu Desktop en la máquina virtual.

Tabla 9: Detalles del sistema operativo para cuckoo sanbox

Tipo	**Detalle**
Operating System	Ubuntu 22.04.4 LTS
Kernel Version	Linux 6.5.0-26-generic
Architecture	x86-64
Processors	16 × AMD Ryzen 7 5800X 8-Core Processor
Memory	4 GiB of RAM

Fuente: Elaboración propia

Ilustración 17: Detalles de configuración de máquina virtual para análisis dinámico

Fuente: Captura de pantalla

Instalación de Cuckoo Sandbox en Ubuntu

Tabla 10: Instalación de Cuckoo Sandbox en Ubuntu

Instalación de Cuckoo Sandbox en Ubuntu.	
Descripción	**Comandos e instrucciones**
Actualizar el Sistema	sudo apt update sudo apt upgrade

Instalar Herramientas y Dependencias	sudo apt install net-tools sudo apt-get install python2.7 python2.7-dev libffi-dev libssl-dev sudo apt-get install python3 sudo apt-get install python3-virtualenv python-setuptools sudo apt-get install libjpeg-dev zlib1g zlib1g-dev libcap-ng-dev swig sudo apt-get install libpcre3 libpcre3-dbg libpcre3-dev build-essential sudo apt-get install libpcap-dev libnet1-dev libyaml-0-2 libyaml-dev pkg-config make sudo apt-get install libmagic-dev libjansson-dev libnss3-dev libgeoip-dev libcap-ng0 sudo apt-get install liblua5.1-dev libhiredis-dev libevent-dev python3-yaml rustc cargo sudo apt install python2-setuptools-whl sudo apt install python2-pip-whl sudo apt install samba-common-bin sudo apt-get install apparmor-utils sudo apt install curl sudo apt-get install mitmproxy
Instalar pip para Python 2.7	curl https://bootstrap.pypa.io/pip/2.7/get-pip.py --output get-pip.py sudo python2.7 get-pip.py
Configurar TCPDump y AppArmor	sudo groupadd pcap sudo usermod -a -G pcap luishernandez sudo chgrp pcap /usr/bin/tcpdump sudo setcap cap_net_raw,cap_net_admin=eip /usr/bin/tcpdump sudo aa-disable /usr/bin/tcpdump
Instalar libssl1.1	wget http://archive.ubuntu.com/ubuntu/pool/main/o/openssl/libssl1.1_1.1.0g-2ubuntu4_amd64.deb sudo dpkg -i libssl1.1_1.1.0g-2ubuntu4_amd64.deb
Instalar MongoDB	curl -fsSL https://www.mongodb.org/static/pgp/server-4.4.asc | sudo apt-key add - apt-key list echo "deb [arch=amd64,arm64] https://repo.mongodb.org/apt/ubuntu focal/mongodb-org/4.4 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-4.4.list deb [arch=amd64,arm64] https://repo.mongodb.org/apt/ubuntu focal/mongodb-org/4.4 multiverse sudo apt update sudo apt install mongodb-org sudo systemctl start mongod.service sudo systemctl enable mongod.service

Configurar Virtualenv y Pip	virtualenv venv_cuckoo -p python2 . venv_cuckoo/bin/activate (venv_cuckoo) pip install -U pip setuptools (venv_cuckoo) pip install -U cuckoo
Iniciar Cuckoo	(venv_cuckoo) cuckoo -d
Ajustar las Fuentes de Repositorio	(venv_cuckoo) sudo nano /etc/apt/sources.list
Agregar la siguiente línea al final del archivo	deb http://security.ubuntu.com/ubuntu bionic-security main
Instalar libssl1.0-dev	(venv_cuckoo) sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 3B4FE6ACC0B21F32 (venv_cuckoo) sudo apt update && apt-cache policy libssl1.0-dev (venv_cuckoo) sudo apt-get install libssl1.0-dev (venv_cuckoo) sudo pip2 install m2crypto==0.24.0
Descargar el código fuente de Yara Source code(tar.gz)	https://github.com/VirusTotal/yara/releases/tag/v4.2.3
Instalar Yara	(venv_cuckoo) tar -xzvf yara-4.2.3.tar.gz (venv_cuckoo) mv yara-4.2.3/ /home/luishernandez/venv_cuckoo/lib/python2.7/site-packages/ (venv_cuckoo) cd /home/luishernandez/venv_cuckoo/lib/python2.7/site-packages/yara-4.2.3/ (venv_cuckoo) ./bootstrap.sh (venv_cuckoo) ./configure --enable-macho --enable-magic --enable-dex (venv_cuckoo) make -j 4 (venv_cuckoo) sudo make install (venv_cuckoo) sudo ldconfig (venv_cuckoo) mitmdump (venv_cuckoo) mv .mitmproxy/mitmproxy-ca-cert.p12 .cuckoo/analyzer/windows/bin/ (venv_cuckoo) mv mitmproxy-ca-cert.p12 cert.p12
Unirse a la Comunidad de Cuckoo	(venv_cuckoo) cuckoo community
Instalar ssdeep	wget https://sourceforge.net/projects/ssdeep/files/ssdeep-2.13/ssdeep-2.13.tar.gz/download -O ssdeep-2.13.tar.gz tar -zxf ssdeep-2.13.tar.gz ./configure make sudo make install
Instalar pySSDeep	sudo apt install libfuzzy-dev git clone https://github.com/bunzen/pySSDeep.git python setup.py build python setup.py install
Instalar pydeep	git clone --recursive https://github.com/kbandla/pydeep.git

	python setup.py build python setup.py test python setup.py install

Fuente: Elaboración propia

Configuración de Cuckoo sandbox

Tabla 11: Configuración de Cuckoo sandbox

Instalación de Cuckoo Sandbox en Ubuntu	
Descripción	**Comandos e instrucciones**
Abrir el archivo auxiliary.conf en un editor de texto y realizar las siguientes modificaciones.	[sniffer] tcpdump = /usr/bin/tcpdump [mitm] enabled = yes mitmdump = /usr/bin/mitmdump [replay] mitmdump = /usr/bin/mitmdump [services] enabled = yes
Abrir el archivo cuckoo.conf en un editor de texto y realizar las siguientes modificaciones.	[cuckoo] version_check = no ignore_vulnerabilities = yes api_token = ROx14rmZ********* machinery = physical [resultserver] ip = 192.168.60.3
Abrir el archivo physical.conf en un editor de texto y realizar las siguientes modificaciones.	[physical] machines = physical1 user = windows10 password = windows10 interface = enp0s3 [physical1] label = WIN10 ip = 192.168.60.4

Abrir el archivo reporting.conf en un editor de texto y realizar las siguientes modificaciones.	[misp] enabled = yes url = http://10.0.2.15 apikey = [mongodb] enabled = yes [mattermost] show_virustotal = yes show_signatures = yes show_urls = yes
Abrir el archivo processing.conf en un editor de texto y realizar las siguientes modificaciones.	[misp] enabled = yes url = http://10.0.2.15 apikey = [virustotal] enabled = yes timeout = 60 scan = yes key = eb0a4caf923b62420e43114bea3d111b3d64cba5eb1c06755c5b75f773ef4eed

Fuente: Elaboración propia

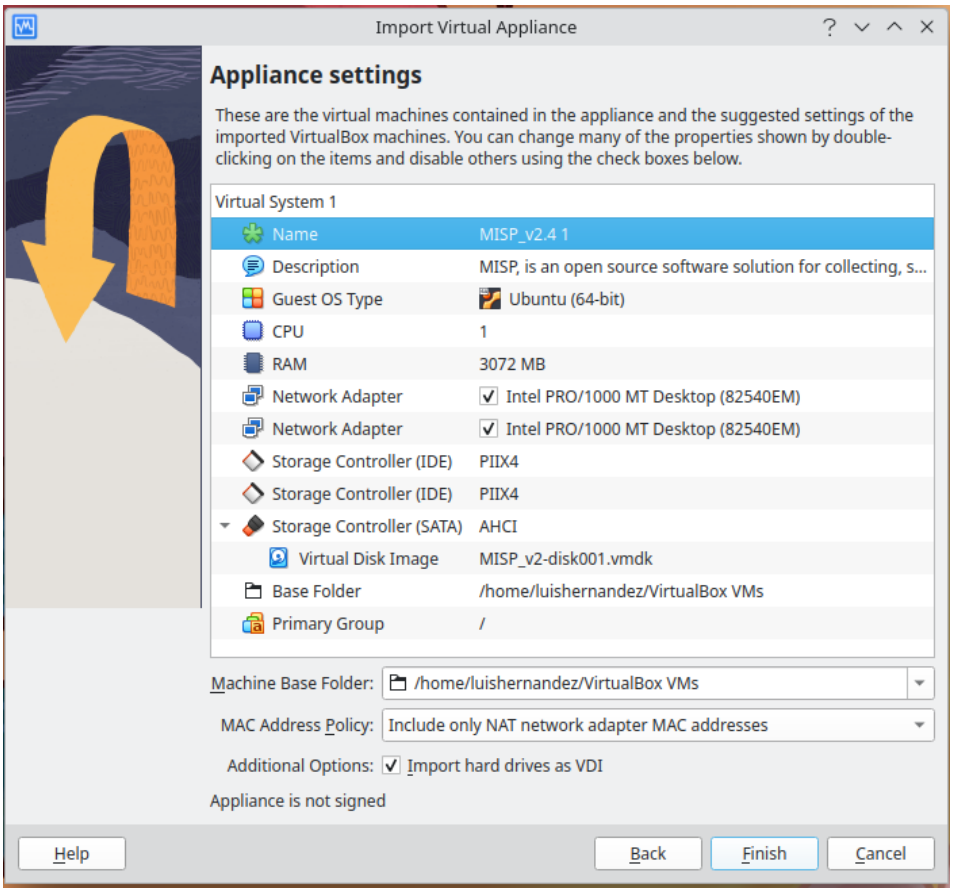
Una vez terminada la instalación y configuración de la máquina, cerrar todo, apagar y crear una instantánea.

3.6.5. Instalación de máquina virtual con MISP

Descargar el Archivo OVA de MISP

- Abrir un navegador web y accede al sitio web oficial de descarga de MISP https://vm.misp-project.org/latest/
- Abrir VirtualBox.
- En la barra de menú, ir a "Archivo" y seleccionar "Importar servicio virtualizado".
- Seleccionar el archivo OVA de MISP que se descargó anteriormente.
- Hacer clic en "Siguiente" y luego en "Importar" para iniciar el proceso de importación.
- En la sección "Red", seleccionar el adaptador de red NAT y seleccionar la red virtual que previamente se había creado.
- Iniciar máquina virtual con MISP.

Ilustración 18: Detalles de la configuración de máquina de MISP



Fuente: Captura de pantalla

3.7.5. Proceso para realizar un análisis de malware

Iniciar Cuckoo Sandbox

Una vez se hayan creado las instantáneas de cada maquinar virtual del entorno controlado ya se podrá realizar los análisis de malware.

- Entrar en el entorno virtual de Python e iniciar Cuckoo Sandbox.

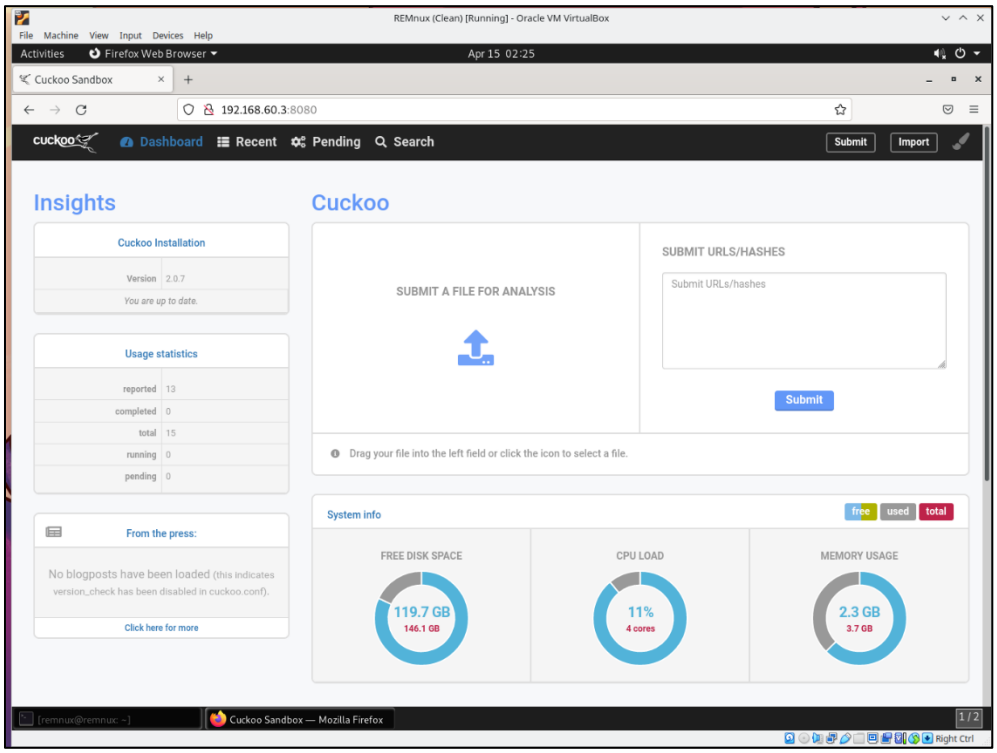
```
luishernandez@cs:~$ . venv_cuckoo/bin/actívate
(venv_cuckoo) luishernandez@cs:~$ cuckoo -d
```

- Cuando se muestre el mensaje [cuckoo.core.scheduler] INFO: Waiting for analysis tasks. Ya se podrá iniciar el análisis.

- Abrir otra terminal para iniciar la interface web de Cucko Sandbox.

```
luishernandez@cs:~$ . venv_cuckoo/bin/actívate
(venv_cuckoo) luishernandez@cs:~$ cuckoo web -host 192.168.60.3 -port 8080
```

Ilustración 19: Interfaz web de cockoo



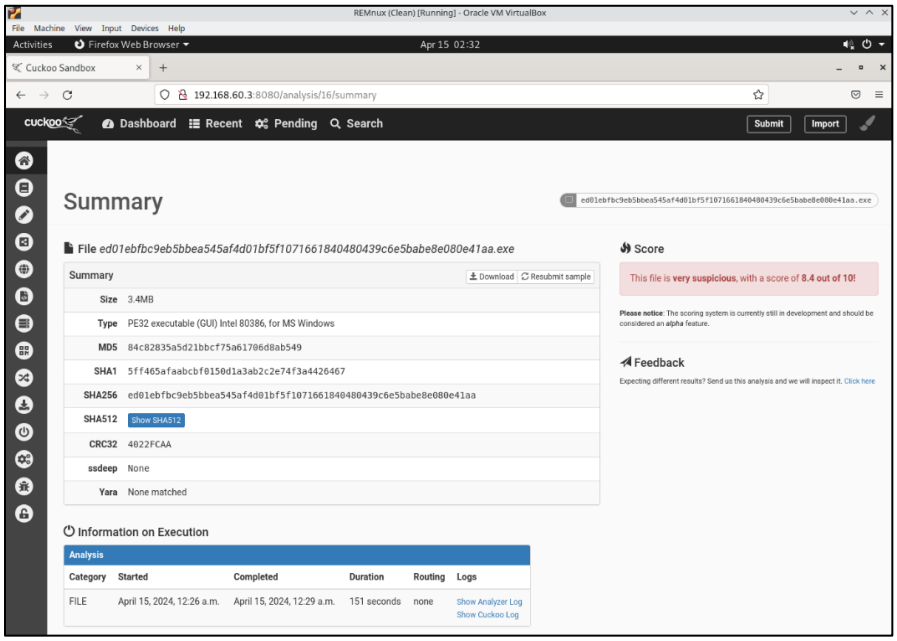
Fuente: Captura de pantalla

Descarga y análisis de muestras

- Iniciar la máquina virtual para el análisis estático (remnux) desde aquí se descargarán las muestras de malware y se podrá usar el kit de herramientas que ofrece Remnux para realizar un análisis estático del malware previo a realizar el análisis dinámico.
- Clonar el repositorio de TheZoo el cual contiene muestras de malware se podrán analizar. Guia de uso e instalación en https://www.github.com/ytisf/theZoo
- Dentro de la carpeta theZoo/malware/Binaries se encuentran las muestras de malware.
- Seleccionar una muestra y descomprimir su contenido, la contraseña por defecto es “infected”.

- Abrir la interfaz web de cockoo a través de un navegador web, ingresar a la IP de la maquina donde se instaló Cuckoo 192.168.60.3 y el puerto 8080 que se especificó cuando se inició previamente la interface web.
- Seleccionar “Submit a file for analisys” y navegar hasta el directorio donde se encuentra el ejecutable que se descomprimió, seleccionar el ejecutable y hacer clic en Open.
- Seleccionar la casilla donde se listo el malware y hacer clic en Analize.
- Se indicará que el análisis se está procesando y se debe esperar a que finalice.
- Una vez finalizado el análisis se podrá ver los resultados en la interfaz web.

Ilustración 20: Interfaz de resultados del análisis de malware



Fuente: Captura de pantalla

4. Desarrollo de pruebas de la metodología de análisis

La metodología desarrollada en este proyecto de investigación proporciona un entorno seguro para ejecutar y monitorear muestras de malware, lo que permite observar su comportamiento y sus posibles impactos en un sistema. A través de este análisis, se busca comprender mejor las tácticas, técnicas y procedimientos utilizados por cada muestra de malware, así como identificar posibles patrones o características comunes entre ellas.

Al examinar los resultados obtenidos en el entorno controlado de Cuckoo Sandbox, se espera obtener una visión más clara de la funcionalidad y el impacto de cada muestra de malware, lo que puede proporcionar información valiosa para el desarrollo de estrategias de detección y mitigación de amenazas en entornos reales.

4.1. Resultados de los análisis de malware

En esta sección, se presenta un análisis detallado de los resultados obtenidos a partir del entorno controlado de Cuckoo Sandbox. Se han seleccionado diversas muestras de malware, incluyendo Cerber, WannaCry, NSIS, Alina, Hupigon, Starbinuq y Dofoil, con el objetivo de evaluar su comportamiento y características en un entorno simulado.

4.1.1. Cerber

Tabla 12: Detalles de análisis de malware 1

Detalles			
Fecha de análisis	28 de abril de 2024, 00:48	**Duración**	129 segundos
Nombre	cerber.exe		
Tamaño	604.5 KB		
Tipo ejecutable	PE32 (GUI) Intel 80386, para MS Windows		
MD5	8b6bc16fd137c09a08b02bbe1bb7d670		
SHA1	c69a0f6c6f809c01db92ca658fcf1b643391a2b7		
SHA256	e67834d1e8b38ec5864cfa101b140aeaba8f1900a6e269e6a94c90fcbfe56678		
CRC32	ED332B67		
ssdeep	6144:yYghlI5/u8f1mr+4RJ99MpDa52RX5wRDhOOU0qsR:yYKlYmDXEpDHRXP01		
Resultado Yara			
Ninguno coincide			
Resultado de motores Antivirus en Virustotal			
Antivirus	**Malware detectado**	**Antivirus**	**Malware detectado**
Bkav	W32.GoodTegoMetAAB.Trojan	MicroWorld-eScan	Trojan.Ransom.Kryptik.A
Lionic	Trojan.Win32.Cerber.4!c	Rising	Trojan.Kryptik!1.AACA (CLASSIC)
Cynet	Malicious (score: 99)	Emsisoft	Trojan.Ransom.Kryptik.A (B)
CAT-QuickHeal	Ransom.Cerber.A4	F-Secure	Trojan.TR/AD.Cerber.rrsbl
Skyhigh	BehavesLike.Win32.Generic.jm	DrWeb	Trojan.Encoder.4691
ALYac	Trojan.Ransom.Cerber	Zillya	Trojan.GenKryptik.Win32.6945
Cylance	unsafe	TrendMicro	Ransom_HPCERBER.SMALY5A
VIPRE	Trojan.Ransom.Kryptik.A	Trapmine	malicious.high.ml.score

Sangfor	Trojan.Win32.Save.a	FireEye	Generic.mg.8b6bc16fd137c09a
BitDefender	Trojan.Ransom.Kryptik.A	Sophos	Mal/Cerber-K
K7GW	Trojan (005224381)	Ikarus	Trojan.Agent
K7AntiVirus	Trojan (005224381)	Jiangmin	Trojan.Zerber.ccs
Arcabit	Trojan.Ransom.Kryptik.A	Webroot	W32.Malware.gen
VirIT	Trojan.Win32.Genus.FGOWH	Google	Detected
Symantec	Packed.Generic.459	Avira	TR/AD.Cerber.rrsbl
Elastic	malicious (high confidence)	Antiy-AVL	Trojan[Ransom]/Win32.Zerber
ESET-NOD32	a variant of Win32/Kryptik.HJVC	Kingsoft	malware.kb.a.999
APEX	Malicious	Gridinsoft	Ransom.Win32.Cerber.sd!s1
McAfee	Generic .jy	Xcitium	TrojWare.Win32.Cerber.AGQJ@720o4m
Avast	Win32:RansomX-gen [Ransom]	Microsoft	Ransom:Win32/Cerber
ClamAV	Win.Ransomware.Cerber-6922156-0	ViRobot	Trojan.Win32.S.Agent.619008.G
Kaspersky	HEUR:Trojan.Win32.Generic	ZoneAlarm	HEUR:Trojan.Win32.Generic
Alibaba	Malware:Win32/km_242de.None	GData	Trojan.Ransom.Kryptik.A
NANO-Antivirus	Trojan.Win32.Zerber.epgdfm	Varist	W32/Cerber.BF.gen!Eldorado
SUPERAntiSpyware	Ransom.Cerber/Variant	AhnLab-V3	Win-Trojan/Cerber.Exp

Fuente: Elaboración propia

Tabla 13: Comportamientos sospechosos de malware 1

Comportamientos sospechosos.	
TTP (Táctica, Técnica y Procedimiento)	
T1057, short: Process Discovery T1083, short: File and Directory Discovery T1047, shor%WINDIR%\Management Instrumentation T1204, short: User Execution T1023, short: Shortcut Modification T1082, short: System Information Discovery	
Consultas para el nombre de la computadora	
GetComputerNameA	computer_name: WIN10
Comprueba si un depurador está depurando el proceso	
IsDebuggerPresent	
Recopila información para tomar huellas dactilares del sistema (MachineGuid, DigitalProductId, SystemBiosDate)	
registry	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\MachineGuid

Comprueba la cantidad de memoria en el sistema, esto se puede utilizar para detectar máquinas virtuales que tienen poca memoria disponible (T1082)	
GlobalMemoryStatusEx	
Asigna memoria de lectura, escritura y ejecución (normalmente para descomprimirse)	
NtAllocateVirtualMemory, NtProtectVirtualMemory	
Consulta el tamaño del disco que podría usarse para detectar máquinas virtuales con un tamaño fijo pequeño o asignación dinámica	
GetDiskFreeSpaceExW	root_path:C:\, root_path: C:\Windows, root_path: C:\Windows\system32
Crea documentos (de oficina) en el sistema de archivos	
file c:\Users\windows10\documents\smkiowuklqiepdf.docm file c:\Users\windows10\documents\nmqqegfokykl.pptx file c:\Users\windows10\documents\hjijvbnibgqqymvvjw.ppt file c:\Users\windows10\documents\bmhzdmqjvq.pptx file c:\Users\windows10\documents\wyhabknyrwtg.ppt	
Crea un acceso directo a un archivo ejecutable (T1023, T1204)	
file C:\Users\windows10\AppData\Roaming\Microsoft\Windows\Recent\Internet Options.lnk file C:\Users\windows10\AppData\Roaming\Microsoft\Windows\Recent\Cuckoo.lnk file C:\Users\windows10\AppData\Roaming\Microsoft\Windows\Recent\C.lnk file C:\Users\windows10\AppData\Roaming\Microsoft\Windows\Recent\Network and Sharing Center.lnk file C:\Users\windows10\AppData\Roaming\Microsoft\Windows\Recent\Network and Internet.lnk file C:\Users\windows10\AppData\Roaming\Microsoft\Windows\Recent\agent.lnk	
Ejecuta una o más consultas WMI (T1047)	
wmi	
Cambia la protección de la memoria de lectura y escritura a lectura y ejecución (probablemente para evitar la detección al configurar todos los indicadores RWX al mismo tiempo)	
NtProtectVirtualMemory	
Comprueba las direcciones de los adaptadores que se pueden utilizar para detectar interfaces de red virtuales	
GetAdaptersAddresses	
Comprueba el identificador único local en el sistema en busca de un privilegio sospechoso	
LookupPrivilegeValueW	privilege_name: SeDebugPrivilege
Se encontraron URL potencialmente maliciosas en el volcado de memoria del proceso	
url http://p27dokhpz2n7nvgr.129p1t.top/3F67-830F-3954-0446-944F url http://p27dokhpz2n7nvgr.14ewqv.top/3F67-830F-3954-0446-944F url http://p27dokhpz2n7nvgr.14vvrc.top/3F67-830F-3954-0446-944F url http://p27dokhpz2n7nvgr.1apgrn.top/3F67-830F-3954-0446-944F url http://p27dokhpz2n7nvgr.12hygy.top/3F67-830F-3954-0446-944F url http://p27dokhpz2n7nvgr.onion/3F67-830F-3954-0446-944F	
Termina otro proceso	
NtTerminateProcess	
Intentos de detectar Cuckoo Sandbox mediante la presencia de un archivo (T1083, T1057)	

file c:\Cuckoo\agent.py, file c:\nxtcyxym\analyzer.py
URL encontradas relacionadas con Tor en volcado de memoria de proceso (por ejemplo, servicios cebolla, Tor2Web y ransomware)
url https://www.torproject.org/download/download-easy.html.en url https://www.torproject.org/downlo url https://www.torproject.org/ url http://p27dokhpz2n7nvgr.onion/3F67-830F-3954-0446-944F
Escribe un posible mensaje de rescate en el disco
buffer: CERBER RANSOMWARE ----- YOUR DOCUMENTS, PH0TOS, DATABASES AND OTHER IMPORTANT FILES HAVE BEEN ENCRYPTED! ----- The only way to decrypt y0ur files is to receive the private key and decryption program. To receive the private key and decryption program go to any decrypted folder, inside there is the special file (*_READ_THIS_FILE_*) with complete instructions how to decrypt your files. If you cannot find any (*_READ_THIS_FILE_*) file at your PC, follow the instructions below: ----- 1. Download "Tor Browser" from https://www.torproject.org/ and install it. 2. In the "Tor Browser" open your personal page here: http://p27dokhpz2n7nvgr.onion/3F67-830F-3954-0446-944F Note! This page is available via "Tor Browser" only. ----- Also you can use temporary addresses on your personal page without using "Tor Browser". ----- 1. http://p27dokhpz2n7nvgr.12hygy.top/3F67-830F-3954-0446-944F 2. http://p27dokhpz2n7nvgr.14ewqv.top/3F67-830F-3954-0446-944F 3. http://p27dokhpz2n7nvgr.14vvrc.top/3F67-830F-3954-0446-944F 4. http://p27dokhpz2n7nvgr.129p1t.top/3F67-830F-3954-0446-944F 5. http://p27dokhpz2n7nvgr.1apgrn.top/3F67-830F-3954-0446-944F ----- Note! These are temporary addresses! They will be available for a limited amount of time! ----- offset: 0 file_handle: 0x000004a4 filepath: C:\nxtcyxym\lib\api_R_E_A_D___T_H_I_S___ATEAH_.txt
Se reanudó un hilo suspendido en un proceso remoto potencialmente indicativo de inyección de proceso
Process 4468 resumed a thread in remote process 3400, NtResumeThread, Process 4468 resumed a thread in remote process 1376, Process 4468 resumed a thread in remote process 3496, Process 4468 resumed a thread in remote process 6860…

Fuente: Elaboración propia

Descripción:

El análisis revela que el malware realiza una amplia variedad de actividades, incluyendo la búsqueda y recopilación de información sensible del sistema, como nombres de equipo y configuraciones de red. Además, el malware muestra un comportamiento de descubrimiento de procesos y archivos, así como la modificación de accesos directos y la ejecución de consultas WMI para obtener más información del sistema.

Una observación destacada es el intento de evitar la detección y el análisis mediante la finalización de procesos, la comprobación de la presencia de depuradores y la detección de entornos de análisis como Cuckoo Sandbox. Además, se identifica la inyección de código en procesos remotos y la manipulación de memoria, indicando un intento de ejecutar código malicioso en el sistema comprometido.

La presencia de URLs potencialmente maliciosas y la escritura de posibles mensajes de rescate en el disco sugieren que el malware está diseñado para realizar actividades de ransomware, cifrando archivos y exigiendo un rescate para su recuperación.

Tácticas, Técnicas y Procedimientos (TTP):

- T1057 - Descubrimiento de procesos
- T1083 - Descubrimiento de archivos y directorios
- T1047 - Ejecución de consultas WMI
- T1204 - Ejecución por parte del usuario
- T1023 - Modificación de accesos directos
- T1082 - Descubrimiento de información del sistema

Comportamientos e IOCs:

- Recopila información del sistema como nombre de equipo, hardware, software instalado, etc.
- Crea archivos de señuelo como documentos de Office en C:\Users\windows10\documents\
- Crea accesos directos en carpeta Inicio reciente apuntando a ejecutable malicioso
- Realiza consultas WMI
- Busca indicios de ejecución en entorno de análisis como Cuckoo Sandbox
- Cifra archivos de la víctima y deja nota de rescate apuntando a URLs .onion en la red Tor
- Intenta terminar otros procesos
- Indica inyección de código en procesos remotos
- Busca presencia de privilegio SeDebugPrivilege
- URLs de C&C: http://p27dokhpz2n7nvgr.onion, http://p27dokhpz2n7nvgr.129p1t.top, etc.
- Rutas de archivos: C:\nxtcyxym\lib\api_R_E_A_D___T_H_I_S___ATEAH_.txt

Conclusión:

Cerber es un ransomware que sigue un patrón típico de este tipo de amenazas, recopilando información de la víctima, creando persistencia, detectando entornos análisis, cifrando datos y exigiendo rescate a través de la red Tor. El comportamiento del malware analizado revela un alto grado de sofisticación y planificación, con el objetivo de comprometer y controlar el sistema infectado para fines maliciosos.

4.2.1. WannaCry

Tabla 14: Detalles de análisis de malware 2

Detalles			
Fecha de análisis	28 de abril de 2024, 01:11	**Duración**	188 segundos
Nombre	ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa.exe		
Tamaño	3.4MB		
Tipo ejecutable	PE32 (GUI) Intel 80386, para MS Windows		
MD5	84c82835a5d21bbcf75a61706d8ab549		
SHA1	5ff465afaabcbf0150d1a3ab2c2e74f3a4426467		
SHA256	ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa		
CRC32	4022FCAA		
ssdeep	98304:QqPoBhz1aRxcSUDk36SAEdhvxWa9P593R8yAVp2g3x:QqPe1Cxcxk3ZAEUadzR8yc4gB		
Resultado Yara			
WannaDecryptor - Detection for common strings of WannaDecryptor Wanna_Sample_84c82835a5d21bbcf75a61706d8ab549 - Specific sample match for WannaCryptor ransom_telefonica - Ransmoware Telefonica Wanna_Cry_Ransomware_Generic - Detects WannaCry Ransomware on Disk and in Virtual Page WannaCry_Ransomware - Detects WannaCry Ransomware WannaCry_Ransomware_Dropper - WannaCry Ransomware Dropper wannacry_static_ransom - Detects WannaCryptor spreaded during 2017-May-12th campaign and variants			
Resultado de motores Antivirus en Virustotal			
Antivirus	**Malware detectado**	**Antivirus**	**Malware detectado**
Bkav	W32.WanaCryptBTTc.Worm	Emsisoft	Trojan.Ransom.WannaCryptor.A (B)
Lionic	Trojan.Win32.Wanna.toNn	F-Secure	Trojan.TR/Ransom.JB
tehtris	Generic.Malware	DrWeb	Trojan.Encoder.11432
Cynet	Malicious (score: 100)	Zillya	Trojan.WannaCry.Win32.2
CAT-QuickHeal	Ransom.WannaCrypt.A4	TrendMicro	Ransom_WANA.A
Skyhigh	BehavesLike.Win32.RansomWannaCry.wc	Trapmine	malicious.high.ml.score
Cylance	unsafe	FireEye	Generic.mg.84c82835a5d21bbc

VIPRE	Trojan.Ransom.WannaCryptor.A	Sophos	Troj/Ransom-EMG
Sangfor	Ransom.Win32.Save.WannaCry	Ikarus	Trojan-Ransom.WannaCry
Alibaba	Ransom:Win32/WannaCry.ali1020010	Jiangmin	Trojan.Wanna.eo
K7GW	Trojan (0050d7171)	Avira	TR/Ransom.JB
K7AntiVirus	Trojan (0050d7171)	MAX	malware (ai score=100)
Baidu	Win32.Trojan.WannaCry.c	Antiy-AVL	Trojan[Ransom]/Win32.Scatter
VirIT	Trojan.Win32.WannaCry.B	Kingsoft	Win32.Troj.Undef.a
Symantec	Ransom.Wannacry	Gridinsoft	Ransom.Win32.Filecoder.dd
Elastic	malicious (high confidence)	Xcitium	Malware@#4gwtqo9z2tkf
ESET-NOD32	Win32/Filecoder.WannaCryptor.D	Arcabit	Trojan.Ransom.WannaCryptor.A
APEX	Malicious	ViRobot	Trojan.Win32.S.WannaCry.3514368.N
Paloalto	generic.ml	ZoneAlarm	Trojan-Ransom.Win32.Wanna.zbu
ClamAV	Win.Ransomware.Wannacryptor-9940180-0	GData	Win32.Trojan-Ransom.WannaCry.A
Kaspersky	Trojan-Ransom.Win32.Wanna.zbu	AhnLab-V3	Trojan/Win32.WannaCryptor.R200571
BitDefender	Trojan.Ransom.WannaCryptor.A	BitDefenderTheta	Gen:NN.ZexaF.36804.wt0@aGEmS3di
NANO-Antivirus	Trojan.Win32.Ransom.eoptnj	TACHYON	Ransom/W32.WannaCry.Zen
MicroWorld-eScan	Trojan.Ransom.WannaCryptor.A	VBA32	TrojanRansom.WannaCrypt
Rising	Ransom.WanaCrypt!1.AAEB (CLASSIC)	Malwarebytes	Generic.Malware.AI.DDS

Fuente: Elaboración propia

Tabla 15: Comportamientos sospechosos de malware 2

Comportamientos sospechosos.
TTP (Táctica, Técnica y Procedimiento)
T1188, short: Multi-hop Proxy
T1060, short: Registry Run Keys / Startup Folder
T1053, short: Scheduled Task
T1112, short: Modify Registry
T1057, short: Process Discovery
T1204, short: User Execution
T1023, short: Shortcut Modification
T1158, short: Hidden Files and Directories
T1129, short: Execution through Module Load
T1082, short: System Information Discovery
T1045, short: Software Packing

Consultas para el nombre de la computadora	
GetComputerNameA	computer_name: WIN10
Comprueba si un depurador está depurando el proceso	
IsDebuggerPresent	
Se observó la salida de la consola de línea de comando	
WriteConsoleW buffer: 'vssadmin' is not recognized as an internal or external command, operable program or batch file. WriteConsoleW buffer: 'bcdedit' is not recognized as an internal or external command, operable program or batch file. WriteConsoleW buffer: 'bcdedit' is not recognized as an internal or external command, operable program or batch file. WriteConsoleW buffer: 'wbadmin' is not recognized as an internal or external command, operable program or batch file. WriteConsoleW buffer: The operation completed successfully.	
Utiliza las API de Windows para generar una clave criptográfica	
CryptGenKey, CryptExportKey	buffer: ¤RSA1□/)wÔêûècÙVª¢lú~½pçlµý□ü¼g]tÌ 7□Ué+m□VØÎcléû÷2□í□ ãÚ²Ñ□Ìpßñ□idô}×□□ä¡>b¾nMVrØ DråJáEt□u□à□jJDN JÛ6ÓO·òÕñÏ×5÷êád¨=}¯Fh%~□□[□}□ þ^\í$>©ètr!/O□6)#ï¨c□Ö®Ô:¬ü□¦ 1 □'°□:5JÄ□qà□□v&Ã´ø¦çQ□ Î ÍE□å□94 `| ¨g,Û¦`_ÐÞh7Ò¸Cxz¦Q¥ãìCÛ!çðyqþkÑÀ crypto_handle: 0x00812378 flags: 0 crypto_export_handle: 0x00000000 blob_type: 6
El ejecutable utiliza un empaquetador conocido (T1045)	
packer	Armadillo v1.71
Inicia la escucha de los servidores	
bind ip_address: 127.0.0.1 socket: 480 port: 0 listen socket: 480 backlog: 1 accept ip_address: 127.0.0.1 socket: 480 port: 49792 bind ip_address: 127.0.0.1 socket: 772 port: 9050 listen socket: 772 backlog: 2147483647 accept ip_address: 127.0.0.1 socket: 772 port: 49813 accept ip_address: 127.0.0.1 socket: 772 port: 49814 accept ip_address: 127.0.0.1 socket: 772 port: 50493 accept ip_address: 127.0.0.1 socket: 772 port: 51160	
Asigna memoria de lectura, escritura y ejecución (normalmente para descomprimirse)	
NtAllocateVirtualMemory, NtProtectVirtualMemory	
Un proceso intentó retrasar la tarea de análisis	
ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa.exe intentó dormir 1173 segundos, en realidad retrasó el tiempo de análisis en 1173 segundos	

Consulta el tamaño del disco que podría usarse para detectar máquinas virtuales con un tamaño fijo pequeño o asignación dinámica (T1082)	
GetDiskFreeSpaceExW	root_path: C:\, root_path: C:\Windows, root_path: C:\Windows\system32
Crea documentos (office) en el sistema de archivos	
file C:\Users\windows10\Documents\XxgQlAQpYjIyMXwmQ.docm file C:\Users\windows10\Documents\IixnbgXLWRMBGh.ppt file C:\Users\windows10\Documents\hSwzYQKJCY.docm file C:\Users\windows10\Documents\lbcMXNqsYu.doc file C:\Users\windows10\Documents\zHgsIaeWjAQwt.docm file C:\Users\windows10\Documents\cmCbnIIGFj.docx file C:\Users\windows10\Documents\FopkyUdisDCzFIgYXNPM.pptx file C:\Users\windows10\Documents\BSGwwnwSUI.docx	
Crea archivos ejecutables en el sistema de archivos (T1129)	
file C:\Users\windows10\AppData\Local\Temp\TaskData\Tor\ssleay32.dll file C:\Users\windows10\AppData\Local\Temp\TaskData\Tor\libeay32.dll file C:\Users\windows10\AppData\Local\Temp\TaskData\Tor\libssp-0.dll file C:\Users\windows10\AppData\Local\Temp\TaskData\Tor\libevent_extra-2-0-5.dll file C:\Users\windows10\AppData\Local\Temp\TaskData\Tor\libevent-2-0-5.dll file C:\Users\windows10\AppData\Local\Temp\m.vbs file C:\Users\windows10\AppData\Local\Temp\TaskData\Tor\zlib1.dll file C:\Users\windows10\AppData\Local\Temp\326861714322885.bat file C:\Users\windows10\AppData\Local\Temp\taskse.exe file C:\Users\windows10\AppData\Local\Temp\TaskData\Tor\libgcc_s_sjlj-1.dll file C:\Users\windows10\AppData\Local\Temp\TaskData\Tor\libevent_core-2-0-5.dll file C:\Users\windows10\AppData\Local\Temp\taskdl.exe file C:\Users\windows10\AppData\Local\Temp\TaskData\Tor\tor.exe	
Crea archivos ocultos o de sistema (T1158)	
NtCreateFile	create_disposition: 5 (FILE_OVERWRITE_IF) file_handle: 0x00000280 filepath: C:\Users\windows10\Desktop\~SDCA5F.tmp desired_access: 0x40100080 (FILE_READ_ATTRIBUTES|SYNCHRONIZE|GENERIC_WRITE) file_attributes: 2 (FILE_ATTRIBUTE_HIDDEN) filepath_r: \??\C:\Users\windows10\Desktop\~SDCA5F.tmp create_options: 96 (FILE_NON_DIRECTORY_FILE|FILE_SYNCHRONOUS_IO_NONALERT) status_info: 3 (FILE_OVERWRITTEN) share_access: 0 ()
Crea 645 tipos de archivos MIME desconocidos que indican que el ransomware escribe archivos cifrados en el disco.	

c:\python27\include\bytesobject.h.wncry, c:\python27\tools\scripts\readme.txt.wncry, %TEMP%\319.WNCRYT, %TEMP%\720.WNCRYT, %TEMP%\316.WNCRYT, %LOCALAPPDATA%\packages\microsoft.windows.search_cw5n1h2txyewy\ac\appcache\865rcn5r\1\c__windows_systemapps_microsoft.windows.search_cw5n1h2txyewy_cache_desktop_16[1].txt.wncry, c:\python27\tcl\tcl8.5\msgs\fi.msg.wncry, c:\python27\lib\test\ieee754.txt.wncry, c:\python27\lib\email\test\data\msg_16.txt.wncry…
Crea un acceso directo a un archivo ejecutable (T1023, T1204)
file C:\Users\windows10\AppData\Local\Temp\@WanaDecryptor@.exe.lnk
Instalar Tor en la máquina (T1188)
%APPDATA%\tor\cached-certs, %APPDATA%\tor\cached-consensus, %APPDATA%\tor\cached-descriptors, %APPDATA%\tor\geoip
Se reanudó un hilo suspendido en un proceso remoto potencialmente indicativo de inyección de proceso
Process 1544 resumed a thread in remote process 2400, NtResumeThread, Process 1544 resumed a thread in remote process 5844, Process 1544 resumed a thread in remote process 708, Process 1544 resumed a thread in remote process 6864…
Escribe un posible mensaje de rescate en el disco
buffer: Q: What's wrong with my files? A: Ooops, your important files are encrypted. It means you will not be able to access them anymore until they are decrypted. If you follow our instructions, we guarantee that you can decrypt all your files quickly and safely! Let's start decrypting! Q: What do I do? A: First, you need to pay service fees for the decryption. Please send $300 worth of bitcoin to this bitcoin address: 12t9YDPgwueZ9NyMgw519p7AA8isjr6SMw Next, please find an application file named "@WanaDecryptor@.exe". It is the decrypt software. Run and follow the instructions! (You may need to disable your antivirus for a while.) Q: How can I trust? A: Don't worry about decryption. We will decrypt your files surely because nobody will trust us if we cheat users. * If you need our assistance, send a message by clicking <Contact Us> on the decryptor window. offset: 0 file_handle: 0x000001e8 filepath: C:\Users\windows10\AppData\Local\Temp\@Please_Read_Me@.txt
Elimina una gran cantidad de archivos del sistema que indican ransomware, malware de limpieza o destrucción del sistema.
%LOCALAPPDATA%\Packages\Microsoft.Windows.PinningConfirmationDialog_cw5n1h2txyewy\AC\INetCache\~SDEFDC.tmp, %LOCALAPPDATA%\Packages\Microsoft.Windows.NarratorQuickStart_8wekyb3d8bbwe\AppData\~SDEEE5.tmp, %LOCALAPPDATA%\Microsoft\input\es-AR\~SDDD67.tmp, %LOCALAPPDATA%\Microsoft\Internet Explorer\TabRoaming\~SDDE98.tmp, C:\Python27\tcl\tcl8.5\msgs\ta_in.msg.WNCRYT, %LOCALAPPDATA%\Packages\Microsoft.DesktopAppInstaller_8wekyb3d8bbwe\AppData\~SDE38B.tmp, %LOCALAPPDATA%\Microsoft\OneDrive\19.043.0304.0013_1\sv\~SDE066.tmp…
Se instala automáticamente para ejecución automática al iniciar Windows(T1060, T1053)
HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Run\yyqdyhfxxmyx066 reg_value "C:\Users\windows10\AppData\Local\Temp\tasksche.exe"
Agrega una extensión de archivo de ransomware WannaCry conocida a los archivos que han sido cifrados

%TEMP%\707.WNCRYT, %TEMP%\385.WNCRYT, C:\Python27\tcl\tcl8.5\msgs\en_au.msg.WNCRY, C:\Python27\Lib\email\test\data\msg_23.txt.WNCRYT, %TEMP%\637.WNCRYT, %TEMP%\361.WNCRYT, C:\Python27\tcl\tcl8.5\msgs\bn_in.msg.WNCRY, C:\Python27\Lib\email\test\data\msg_31.txt.WNCRYT, %TEMP%\hibsys.WNCRYT, %TEMP%\344.WNCRYT, %TEMP%\556.WNCRYT, C:\Python27\tcl\tcl8.5\msgs\he.msg.WNCRYT…	
Muestra interés en procesos en ejecución específicos (T1057)	
process wmic.exe, process cmd.exe	
Crea o establece una clave de registro para una larga serie de bytes, posiblemente para almacenar una configuración binaria o de malware (T1112)	
NtSetValueKey	key_handle: 0x000002bc index: 0 reg_type: 7 (REG_MULTI_SZ) regkey: HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\Session Manager\PendingFileRenameOperations
Busca procesos en ejecución potencialmente para identificar procesos de evasión de sandbox, inyección de código o volcado de memoria (T1057)	
Process32NextW	smss.exe, wininit.exe, csrss.exe, services.exe, fontdrvhost.exe, dwm.exe, upfc.exe, spoolsv.exe, svchost.exe

Fuente: Elaboración propia

Descripción:

El malware exhibe una diversidad de técnicas para lograr sus objetivos, desde la modificación del registro del sistema hasta la ejecución de tareas programadas y la creación de accesos directos a archivos ejecutables. Además, se observa el uso de empaquetadores conocidos y la creación de archivos ejecutables y ocultos en ubicaciones temporales.

Un aspecto importante es la instalación de Tor en la máquina comprometida, junto con la creación de servidores de escucha locales. Esto sugiere la intención del malware de establecer conexiones anónimas y posiblemente comunicarse con servidores de comando y control remotos.

El comportamiento del malware también revela su capacidad para cifrar archivos en el sistema, como se evidencia por la creación de archivos con extensiones específicas asociadas al ransomware WannaCry. Además, se observa la escritura de posibles mensajes de rescate en el disco, indicando la intención de exigir un rescate a cambio de la restauración de los archivos cifrados.

Tácticas, Técnicas y Procedimientos (TTP):

- T1188 - Proxy multinivel

- T1060 - Claves de ejecución en el registro
- T1053 - Programación de tareas
- T1112 - Modificación del registro
- T1057 - Descubrimiento de procesos
- T1204 - Ejecución por parte del usuario
- T1023 - Modificación de accesos directos
- T1158 - Archivos y directorios ocultos
- T1129 - Ejecución mediante carga de módulos
- T1082 - Descubrimiento de información del sistema
- T1045 - Empaquetado de software

Comportamientos e IOCs:

- Genera una clave criptográfica usando CryptoAPI
- Se empaqueta con Armadillo v1.71
- Inicia servidores en puertos locales
- Duerme el proceso de análisis durante 1173 segundos
- Crea archivos ofimáticos de señuelo en %USERPROFILE%\Documents
- Crea archivos ejecutables en %TEMP%\TaskData\Tor y %APPDATA%\tor (instalación de Tor)
- Crea archivos ocultos y de sistema
- Crea cientos de archivos con extensión .wncryt/.wncry (cifrado)
- Crea acceso directo a @WanaDecryptor@.exe
- Deja nota de rescate en @Please_Read_Me@.txt con direcciones bitcoin
- Elimina gran cantidad de archivos del sistema
- Se instala para ejecución automática en HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Run
- Expresa interés en procesos wmic.exe, cmd.exe
- Crea claves de registro binarias en HKLM\SYSTEM\ControlSet001\Control\Session Manager\PendingFileRenameOperations
- Enumera otros procesos en ejecución
- Posible inyección de código en procesos remotos

Conclusión:

WannaCry sigue un patrón típico de ransomware, instalando Tor, cifrando archivos con extensiones características, dejando una nota de rescate con direcciones bitcoin, estableciendo persistencia y realizando diversas comprobaciones anti-análisis. El análisis del comportamiento del malware revela una sofisticada campaña de ataque diseñada para comprometer, controlar y extorsionar a los usuarios mediante la encriptación de archivos y la exigencia de pagos de rescate.

4.3.1. NSIS

Tabla 16: Detalles de análisis de malware 3

Detalles			
Fecha de análisis	28 de abril de 2024, 1:20	**Duración**	137 segundos
Nombre	798_abroad.exe		
Tamaño	1.6MB		
Tipo ejecutable	PE32 (GUI) Intel 80386, para MS Windows, Nullsoft Installer self-extracting archive		
MD5	f88e9b7446a6e57943728cce3cc70720		
SHA1	0030e2b87acebaa040e3f872c13e39af88b733b9		
SHA256	2fd5b075ab9dffe8b421a4942ecdac322d8f0fceca597a644a6a9e631901e8bc		
CRC32	FF9209F1		
ssdeep	24576:sM3B3GYeUXVdhaR/R8mRtCB8J7aTlqCTFfRFeAq05KWg6KH3mrMa1ihn58Y:pR3HeUXo/2UJuIfFjeCKWg6W3UTEhnd		
Resultado Yara			
Ninguno coincide			
Resultado de motores Antivirus en Virustotal			
Antivirus	**Malware detectado**	**Antivirus**	**Malware detectado**
Lionic	Trojan.NSIS.Agent.ISUw	TrendMicro	PUA.Win32.Meinhudong.A
Cynet	Malicious (score: 99)	Trapmine	malicious.high.ml.score
CAT-QuickHeal	Trojan.NSIS.Startpage.AB	FireEye	Trojan.GenericKD.65418229
Skyhigh	GenDownloader.vb	Sophos	Mal/Generic-R
ALYac	Trojan.GenericKD.65418229	Ikarus	Trojan.Win32.Delf
Cylance	unsafe	Google	Detected
VIPRE	Trojan.GenericKD.65418229	Avira	TR/Dropper.Gen
K7AntiVirus	Unwanted-Program (00586e011)	MAX	malware (ai score=100)
BitDefender	Trojan.GenericKD.65418229	Antiy-AVL	Trojan[Packed]/Win32.Dico
K7GW	Unwanted-Program (00586e011)	Kingsoft	Win32.Troj.Agent.a
Cybereason	malicious.446a6e	Xcitium	ApplicUnwnt@#3mf20o95e64o1

Arcabit	Trojan.Generic.D3E633F5	Microsoft	PUADlManager:Win32/Meinhudong
VirIT	Trojan.Win32.DownLoader9.RPG	ZoneAlarm	Trojan-Clicker.NSIS.Agent.a
Symantec	Trojan.ADH	GData	NSIS.Application.Meinhudong.F
ESET-NOD32	Win32/Meinhudong.A potentially unwanted	Varist	W32/Trojan.UDRU-4165
McAfee	Artemis!F88E9B7446A6	DeepInstinct	MALICIOUS
Avast	Win32:Malware-gen	VBA32	TrojanClicker.Agent
Kaspersky	Trojan-Clicker.NSIS.Agent.a	Malwarebytes	Generic.Malware/Suspicious
NANO-Antivirus	Trojan.Win32.Meinhudong.fovrra	Panda	Trj/CI.A
MicroWorld-eScan	Trojan.GenericKD.65418229	TrendMicro-HouseCall	PUA.Win32.Meinhudong.A
Rising	Adware.Agent!1.BEFB (CLASSIC)	Tencent	Nsis.Trojan.Agent.Wwhl
Emsisoft	Trojan.GenericKD.65418229 (B)	Yandex	PUA.StartPage.Gen.JV
F-Secure	Trojan.TR/Dropper.Gen	Fortinet	W32/StartPage.ED!tr
DrWeb	Trojan.DownLoader9.11888	AVG	Win32:Malware-gen
Zillya	Trojan.GenericCRTD.Win32.4459	CrowdStrike	win/malicious_confidence_100% (W)

Fuente: Elaboración propia

Tabla 17: Comportamientos sospechosos de malware 3

Comportamientos sospechosos	
TTP (Táctica, Técnica y Procedimiento)	
T1071, short: Standard Application Layer Protocol T1082, short: System Information Discovery T1089, short: Disabling Security Tools T1012, short: Query Registry T1204, short: User Execution T1023, short: Shortcut Modification T1129, short: Execution through Module Load T1045, short: Software Packing	
Consultas para el nombre de la computadora	
GetComputerNameA	computer_name: WIN10
Comprueba si un depurador está depurando el proceso	
IsDebuggerPresent	

Comprueba la cantidad de memoria en el sistema, esto se puede utilizar para detectar máquinas virtuales que tienen poca memoria disponible (T1082)	
GlobalMemoryStatusEx	
Asigna memoria de lectura, escritura y ejecución (normalmente para descomprimirse)	
NtAllocateVirtualMemory, NtProtectVirtualMemory	
Un proceso intentó retrasar la tarea de análisis	
ailiao.exe intentó dormir 301 segundos, en realidad retrasó el tiempo de análisis en 301 segundos	
Consulta el tamaño del disco que podría usarse para detectar máquinas virtuales con un tamaño fijo pequeño o asignación dinámica	
GetDiskFreeSpaceExW	root_path: C:\Users\windows10\AppData\Local\Microsoft\Windows\Explorer root_path: C:\Windows root_path: C:\Windows\system32
Cambia la protección de la memoria de lectura y escritura a lectura y ejecución (probablemente para evitar la detección al configurar todos los indicadores RWX al mismo tiempo)	
NtProtectVirtualMemory	
Comprueba las direcciones de los adaptadores que se pueden utilizar para detectar interfaces de red virtuales	
GetAdaptersAddresses	
Comprueba el identificador único local en el sistema en busca de un privilegio sospechoso	
LookupPrivilegeValueW	privilege_name: SeDebugPrivilege
Se reanudó un hilo suspendido en un proceso remoto potencialmente indicativo de inyección de proceso	
Process 4936 resumed a thread in remote process 812, NtResumeThread, Process 4936 resumed a thread in remote process 6028, Process 4936 resumed a thread in remote process 1152, Process 4936 resumed a thread in remote process 549	
La actividad de la red contiene más de un agente de usuario único(T1071)	
ailiao.exe Mozilla/5.0 (Windows; U; Windows NT 6.1; zh-CN; rv:1.9.2.3) Gecko/20100401 Firefox/3.6.3 ailiao.exe Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.2; WOW64; Trident/7.0)	
Intenta modificar la configuración de seguridad del navegador (T1089)	
registry	HKEY_CURRENT_USER\SOFTWARE\Microsoft\Internet Explorer\Main\FeatureControl\FEATURE_BROWSER_EMULATION\ailiao.exe
Consultas de aplicaciones potencialmente instaladas (T1012)	
RegOpenKeyExA	regkey_r: Software\Microsoft\Windows\CurrentVersion\Uninstall\°®ÁÄ base_handle: 0x80000002 key_handle: 0x00000000 options: 0 access: 0x00020019 regkey: HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Uninstall\㴰□╞
Crea archivos ejecutables en el sistema de archivos (T1129)	

file C:\ProgramData\Microsoft\Windows\Start Menu\Programs\°®ÁÄ\Ð¶ÔØ°®ÁÄ.lnk
file C:\Users\windows10\AppData\Local\Temp\nszB1DC.tmp\ailiao.exe
file C:\Program Files (x86)\ailiao\ailiao.exe
file C:\Program Files (x86)\ailiao\ailiaou.exe
file C:\Program Files (x86)\ailiao\ailiaotp.exe
file C:\Program Files (x86)\ailiao\ailiao.lnk
file C:\ProgramData\Microsoft\Windows\Start Menu\Programs\°®ÁÄ\°®ÁÄ.lnk
file C:\Program Files (x86)\ailiao\aldesk.exe
file C:\Program Files (x86)\ailiao\uninst.exe
file C:\Users\windows10\AppData\Local\Temp\nszB1DC.tmp\System.dll
file C:\Users\Public\Desktop\°®ÁÄ.lnk
Crea un acceso directo a un archivo ejecutable (T1023, T1204)
file C:\Program Files (x86)\ailiao\ailiao.lnk
file C:\ProgramData\Microsoft\Windows\Start Menu\Programs\°®ÁÄ\Ð¶ÔØ°®ÁÄ.lnk
file C:\Users\windows10\AppData\Roaming\Microsoft\Windows\SendTo\Bluetooth File Transfer.LNK
file C:\ProgramData\Microsoft\Windows\Start Menu\Programs\°®ÁÄ\°®ÁÄ.lnk
file C:\Users\Public\Desktop\°®ÁÄ.lnk

Fuente: Elaboración propia.

Descripción:

El comportamiento del malware revela su intención de modificar la configuración de seguridad del navegador y su capacidad para crear archivos ejecutables y accesos directos en ubicaciones específicas del sistema.

Es especialmente preocupante la capacidad del malware para evadir la detección mediante la modificación de la configuración de seguridad del navegador y la creación de archivos ocultos. Además, se detecta la realización de consultas al registro del sistema para obtener información sobre aplicaciones instaladas.

Tácticas, Técnicas y Procedimientos (TTP):

- T1071 - Protocolo de capa de aplicación estándar
- T1082 - Descubrimiento de información del sistema
- T1089 - Deshabilitación de herramientas de seguridad
- T1012 - Consultas al registro
- T1204 - Ejecución por parte del usuario
- T1023 - Modificación de accesos directos
- T1129 - Ejecución mediante carga de módulos

- T1045 - Empaquetado de software

Comportamientos e IOCs:

- Recopila información del sistema como nombre de equipo, memoria, discos
- Retrasa el proceso de análisis durante 301 segundos
- Cambia permisos de memoria de RW a RX, posible descompresión
- Enumera adaptadores de red e interfaces
- Busca presencia de privilegio SeDebugPrivilege
- Indica posible inyección de código en procesos remotos
- Múltiples agentes de usuario en conexiones de red
- Intenta modificar configuración de seguridad de Internet Explorer
- Consulta aplicaciones instaladas en registro
- Crea varios ejecutables en %PROGRAMFILES%\ailiao y carpetas de inicio
- Crea accesos directos a ejecutables en distintas ubicaciones
- Posible técnica de empaquetado de ejecutables

Conclusión:

Se trata de un troyano downloader que realiza una serie de comprobaciones anti-análisis, recopila información del sistema, despliega componentes en varias carpetas del sistema y modifica el navegador, posiblemente como parte de una campaña de distribución de malware adicional.

4.4.1. Alina

Tabla 18: Detalles de análisis de malware 4

Detalles			
Fecha de análisis	28 de abril de 2024, 01:54	**Duración**	130 segundos
Nombre	3_4.exe		
Tamaño	59.5KB		
Tipo ejecutable	PE32 (GUI) Intel 80386, para MS Windows, UPX compressed		
MD5	1efeb85c8ec2c07dc0517ccca7e8d743		
SHA1	5563e4c2987eda056b3f74716c00d3014b9306bc		
SHA256	036e4f452041f9d573f851d48d92092060107d9ea32e0c532849d61a598b8a71		
CRC32	F16B2627		

ssdeep	768:ZGCOlsTLHcg46+MVjENmeB0hX66PeFo+3mkTr27gPQ1LqgovbBbcrFXEb/VN8CtT:Qqs7ZEey1A34L1LqXb0FXEZt4WY/Kco		
Resultado Yara			
suspicious_packer_section - The packer/protector section names/keywords UPX - (no description)			
Resultado de motores Antivirus en Virustotal			
Antivirus	**Malware detectado**	**Antivirus**	**Malware detectado**
Bkav	W32.AIDetectMalware	MicroWorld-eScan	Gen:Variant.Ransom.LockBit.32
Lionic	Trojan.Win32.Alinaos.trpL	Rising	Spyware.POSCardStealer!8.644 (TFE:5:QmN8hfJ6YBR)
Elastic	malicious (moderate confidence)	Emsisoft	Gen:Variant.Ransom.LockBit.32 (B)
Cynet	Malicious (score: 100)	F-Secure	Trojan.TR/Downloader.Gen
Skyhigh	BehavesLike.Win32.Generic.qc	DrWeb	Trojan.PWS.Banker1.8391
ALYac	Spyware.Infostealer.POS	Zillya	Trojan.POSCardStealer.Win32.244
Cylance	unsafe	TrendMicro	BKDR_ALINA.NA
VIPRE	Gen:Variant.Ransom.LockBit.32	Trapmine	malicious.high.ml.score
Sangfor	Trojan.Win32.Save.a	FireEye	Generic.mg.1efeb85c8ec2c07d
K7AntiVirus	Spyware (004148c71)	Sophos	Troj/Trackr-Gen
BitDefender	Gen:Variant.Ransom.LockBit.32	Ikarus	Trojan-PSW.Agent
K7GW	Spyware (004148c71)	Jiangmin	Trojan.Generic.esysp
Cybereason	malicious.c8ec2c	Google	Detected
Arcabit	Trojan.Ransom.LockBit.32	Avira	TR/Downloader.Gen
VirIT	Trojan.Win32.Generic.CCCD	MAX	malware (ai score=100)
Symantec	Infostealer.Alina	Antiy-AVL	Trojan/Win32.Unknown
tehtris	Generic.Malware	Kingsoft	malware.kb.b.1000
ESET-NOD32	Win32/Spy.POSCardStealer.D	Gridinsoft	Spy.Win32.Alinaos.cc!s2
APEX	Malicious	Xcitium	TrojWare.Win32.Spy.POSCardStealer.AD@8qcspw
McAfee	GenericRXAA-AA!1EFEB85C8EC2	Microsoft	TrojanSpy:Win32/Alinaos
Avast	Win32:Malware-gen	ZoneAlarm	Trojan-Spy.Win32.Alinaos.dw
ClamAV	Win.Trojan.POSCardStealer-6	GData	Gen:Variant.Ransom.LockBit.32

Kaspersky	Trojan-Spy.Win32.Alinaos.dw	Varist	W32/POSCardStealer.C.gen!Eldorado
Alibaba	TrojanSpy:Win32/Alinaos.4f8ccc82	AhnLab-V3	Trojan/Win32.Reedum.R332664
NANO-Antivirus	Trojan.Win32.Banker1.ebnywb	BitDefenderTheta	AI:Packer.C9E2E23F1D

Fuente: Elaboración propia

Tabla 19: Comportamientos sospechosos de malware 4

Comportamientos sospechosos	
TTP (Táctica, Técnica y Procedimiento)	
T1071, short: Standard Application Layer Protocol T1053, short: Scheduled Task T1060, short: Registry Run Keys / Startup Folder T1057, short: Process Discovery T1129, short: Execution through Module Load T1045, short: Software Packing	
Consultas para el nombre de la computadora	
GetComputerNameA	computer_name: WIN10
El ejecutable utiliza un empaquetador conocido (T1045)	
packer	UPX 2.90 [LZMA] -> Markus Oberhumer, Laszlo Molnar & John Reiser
Crea archivos ejecutables en el sistema de archivos (T1129)	
file C:\Users\windows10\AppData\Roaming\dwm.exe	
Busca procesos en ejecución potencialmente para identificar procesos de evasión de sandbox, inyección de código o volcado de memoria (T1057)	
Process32NextW	winlogon.exe, services.exe, lsass.exe, fontdrvhost.exe, svchost.exe, upfc.exe, spoolsv.exe, wlms.exe, WmiPrvSE.exe, MicrosoftEdgeUpdate.exe, taskhostw.exe
Comprueba las direcciones de los adaptadores que se pueden utilizar para detectar interfaces de red virtuales	
GetAdaptersAddresses	
Se reanudó un hilo suspendido en un proceso remoto potencialmente indicativo de inyección de proceso	
Process 6956 resumed a thread in remote process 7028, NtResumeThread	
Se instala automáticamente para ejecución automática al iniciar Windows (T1060, T1053)	
reg_key HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\dwm reg_value C:\Users\windows10\AppData\Roaming\dwm.exe	
Comunicaciones de red indicativas de posible inyección de código originadas en el proceso dwm.exe (T1071)	

WSASend	buffer: POST /whynot/sam.php HTTP/1.1 Accept: text/*, application/octet-stream Content-Type: application/x-www-form-urlencoded User-Agent: Alina v3.4 Host: someligeoas.com Content-Length: 385 Cache-Control: no-cache act=l&b=fe82843c&c=WIN10&v=v3.4&p=C: \Users\windows10\AppData\Roaming\dwm.exe&ldata=f0c2c5d8dfcac7c7c8c3cec8c09 1999e8b979b95f68befcec7cedfcecf8bc4c7cf8bcdc2c7ce8be891f7fed8ced9d8f7dcc2c 5cfc4dcd89a9bf7eadbdbefcadfcaf7e7c4c8cac7f7ffcec6dbf798f49f85ced3cea1f0d8dfca d9dff4dedbcfcadfcef4dfc3d9cecacf9199939e8b979b95f68bdedbcfcadfce8bdfc3d9cec acf8bc7cadec5c8c3cecf8bd8dec8c8ced8d8cddec7c7d2a1 socket: 1112
WSASend	buffer: POST /whynot/sam.php HTTP/1.1 Accept: text/*, application/octet-stream Content-Type: application/x-www-form-urlencoded User-Agent: Alina v3.4 Host: someligeoas.com Content-Length: 78 Cache-Control: no-cache act=d&b=fe82843c&c=WIN10&v=v3.4&p=C:\Users\windows10\AppData\Roaming\dw m.exe&= socket: 1224
InternetConnectA	username: service: 3 hostname: 208.98.63.228 internet_handle: 0x00cc0004 flags: 0 password: port: 80
HttpOpenRequestA	connect_handle: 0x00cc0008 http_version: HTTP/1.1 flags: 2147483648 http_method: POST referer: path: /forum/login.php
URL encontradas en la memoria que apuntan a una dirección IP en lugar de un dominio (potencialmente indicativo de tráfico de Comando y Control)	
url http://208.98.63.228/ url http://208.98.63.228/forum/login.php	

Fuente: Elaboración propia

Descripción:

El malware utiliza una variedad de tácticas y técnicas para llevar a cabo sus acciones maliciosas. Esto incluye la creación de archivos ejecutables en ubicaciones específicas del sistema, la

búsqueda de procesos en ejecución y la modificación del registro para lograr la ejecución automática al iniciar Windows.

Además, se observa una comunicación de red desde el proceso dwm.exe, que parece estar enviando datos a un servidor remoto. Esta comunicación se realiza mediante solicitudes HTTP POST a un dominio específico, lo que sugiere una posible actividad de comando y control.

Otro comportamiento preocupante es la reanudación de un hilo suspendido en un proceso remoto, lo que podría ser indicativo de una inyección de proceso en curso.

La presencia de direcciones IP en las URL encontradas en la memoria también es preocupante y sugiere una posible comunicación con servidores de comando y control utilizando direcciones IP en lugar de nombres de dominio.

Tácticas, Técnicas y Procedimientos (TTP):

- T1071 - Protocolo de capa de aplicación estándar
- T1053 - Programación de tareas
- T1060 - Claves de ejecución en el registro
- T1057 - Descubrimiento de procesos
- T1129 - Ejecución mediante carga de módulos
- T1045 - Empaquetado de software

Comportamientos e IOCs:

- Empaquetado con UPX 2.90
- Crea ejecutable dwm.exe en %APPDATA%\Roaming
- Enumera procesos en ejecución
- Enumera adaptadores de red
- Posible inyección de código en procesos remotos
- Se instala en HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run para ejecución automática
- Realiza conexiones POST a someligeoas.com con datos codificados
- URLs de C&C apuntando a 208.98.63.228
- Posible exfiltración de datos de tarjetas de pago

Conclusión:

Se trata de un troyano espía bancario que roba datos de tarjetas de pago, se instala para persistencia, enumera procesos y redes, e intenta inyectarse en procesos legítimos para ocultar su actividad mientras exfiltra la información robada.

4.5.1. Hupigon

Tabla 20: Detalles de análisis de malware 5

Detalles			
Fecha de análisis	28 de abril de 2024, 02:43	**Duración**	128 segundos
Nombre	Hupigon.ex_		
Tamaño	705.0KB		
Tipo ejecutable	PE32 (GUI) Intel 80386, para MS Windows		
MD5	8f90057ab244bd8b612cd09f566eac0c		
SHA1	e8da95ff4801ff951fe8957bcafa31fb3c8251cb		
SHA256	730e1337cf9ecf842a965ea458ee241c2a1e5b0ef1daccde87cd628eb4b37057		
CRC32	21C7F194		
ssdeep	12288:yRycYktU4g/n/t0EW5A0zkOvJwQ5oalK+GxhAv6DIk6bQQ52LwRg08S5H91Zt:exnU4gf2EW5A2HJr1krhAvuIk6LXd		
Resultado Yara			
CookieTools - Chinese Hacktool Set - file CookieTools.exe			
Resultado de motores Antivirus en Virustotal			
Antivirus	**Malware detectado**	**Antivirus**	**Malware detectado**
Bkav	W32.AIDetectMalware	Alibaba	Backdoor:Win32/Hupigon.80dfbfcf
tehtris	Generic.Malware	NANO-Antivirus	Trojan.Win32.Hupigon.fhxxdo
Cynet	Malicious (score: 100)	MicroWorld-eScan	Trojan.Delf.Inject.Z
CMC	Generic.Win32.8f90057ab2!MD	Rising	Backdoor.Win32.Gpigeon.l (CLASSIC)
CAT-QuickHeal	Backdoor.Hupigon.DI8	Emsisoft	Trojan.Delf.Inject.Z (B)
Skyhigh	BehavesLike.Win32.Generic.bh	F-Secure	Backdoor.BDS/Hupigon.Gen
ALYac	Trojan.Delf.Inject.Z	DrWeb	BackDoor.Pigeon.21851
Cylance	unsafe	Zillya	Backdoor.Hupigon.Win32.1
VIPRE	Trojan.Delf.Inject.Z	TrendMicro	BKDR_HUPIGON.VEM
Sangfor	Virus.Win32.Save.a	Trapmine	malicious.high.ml.score
CrowdStrike	win/malicious_confidence_100% (W)	FireEye	Generic.mg.8f90057ab244bd8b
BitDefender	Trojan.Delf.Inject.Z	Sophos	Mal/Generic-S

K7GW	Trojan (7000000f1)	Ikarus	Backdoor.Win32.Hupigon
K7AntiVirus	Trojan (7000000f1)	Jiangmin	Backdoor/Huigezi.2007.rbu
Arcabit	Trojan.Delf.Inject.Z	Google	Detected
Baidu	Win32.Backdoor.Hupigon.e	Avira	BDS/Hupigon.Gen
VirIT	Backdoor.Win32.Hupigon.Y	MAX	malware (ai score=100)
Symantec	Backdoor.Graybird	Antiy-AVL	Trojan[Backdoor]/Win32.Hupigon.pv
Elastic	malicious (high confidence)	Kingsoft	Win32.Hack.HuigeziT.cz
ESET-NOD32	a variant of Win32/Hupigon	Gridinsoft	Backdoor.Win32.Hupigon.cc!s1
APEX	Malicious	Xcitium	TrojWare.Win32.Trojan.Agent.Gen@2j3wh
McAfee	BackDoor-AWQ.svr.gen.e	Microsoft	Backdoor:Win32/Hupigon.DI
Avast	Win32:GenMalicious-BND [Trj]	ViRobot	Backdoor.Win32.Hupigon.721408.AJ
ClamAV	Win.Trojan.Delf-1526	ZoneAlarm	Backdoor.Win32.Hupigon.axbr
Kaspersky	Backdoor.Win32.Hupigon.axbr	GData	Win32.Trojan.PSE.12OKESO

Fuente: Elaboración propia

Tabla 21: Comportamientos sospechosos de malware 5

Comportamientos sospechosos	
TTP (Táctica, Técnica y Procedimiento)	
T1053, short: Scheduled Task T1060, short: Registry Run Keys / Startup Folder T1031, short: Modify Existing Service T1129, short: Execution through Module Load	
Se observó la salida de la consola de línea de comando	
WriteConsoleW	buffer: The system cannot accept the date entered. buffer: Enter the new date: (mm-dd-yy) buffer: C:\Users\windows10\AppData\Local\Temp> buffer: DEL buffer: "C:\Users\windows10\AppData\Local\Temp\Hupigon.ex_" buffer: C:\Users\windows10\AppData\Local\Temp> buffer: if buffer: exist "C:\Users\windows10\AppData\Local\Temp\Hupigon.ex_" buffer: GGTO buffer: try buffer: C:\Users\windows10\AppData\Local\Temp> buffer: DEL buffer: C:\Windows\UNINSTAL.BAT

	buffer: The system cannot find message text for message number 0x236c in the message file for Application.
Asigna memoria de lectura, escritura y ejecución (normalmente para descomprimirse)	
NtAllocateVirtualMemory, NtProtectVirtualMemory	
Lengua extranjera identificada en el recurso PE	
name RT_ICON language LANG_CHINESE filetype Device independent bitmap graphic, 32 x 64 x 24, image size 3072 sublanguage SUBLANG_CHINESE_SIMPLIFIED name RT_GROUP_ICON language LANG_CHINESE filetype data sublanguage SUBLANG_CHINESE_SIMPLIFIED name RT_VERSION language LANG_CHINESE filetype data sublanguage SUBLANG_CHINESE_SIMPLIFIED	
Crea archivos ejecutables en el sistema de archivos (T1129)	
file C:\Windows\UNINSTAL.BAT	
Crea un servicio (T1031)	
CreateServiceA	start_type: 2 display_name: GrayPigeon_Hacker.com.cn filepath: C:\Windows\Hacker.com.cn.exe service_name: GrayPigeon_Hacker.com.cn filepath_r: C:\Windows\Hacker.com.cn.exe desired_access: 983551 service_handle: 0x0069b7e0 service_type: 272 service_manager_handle: 0x0067b4d0
Comprueba el identificador único local en el sistema en busca de un privilegio sospechoso	
LookupPrivilegeValueW	privilege_name: SeDebugPrivilege
Se instala automáticamente para ejecución automática al iniciar Windows(T1060, T1053)	
service_name GrayPigeon_Hacker.com.cn service_path C:\Windows\Hacker.com.cn.exe	
Crea archivos Hupigon, claves de registro y/o mutex conocidos	
file C:\Windows\bootstat.dat	

Fuente: Elaboración propia

Descripción:

Entre las tácticas y técnicas utilizadas por este malware se incluye la creación de tareas programadas y claves de registro para lograr una ejecución persistente al iniciar el sistema. Además, se observa la asignación de memoria con permisos de lectura, escritura y ejecución, lo que generalmente se asocia con acciones de descompresión de archivos.

Se identifica la presencia de lenguaje extranjero en los recursos del archivo ejecutable, lo que puede indicar una estrategia de evasión de detección. Además, se crea un servicio malicioso con un nombre y una ruta de archivo sospechosos.

La salida de la consola de línea de comandos muestra operaciones inusuales, como intentos de eliminación de archivos y verificación de la existencia de otros archivos. Esto sugiere una actividad anómala en el sistema que puede ser indicativa de un intento de encubrimiento por parte del malware.

Por último, la creación de archivos y claves de registro asociadas con el malware Hupigon es otro indicador claro de actividad maliciosa en el sistema.

Tácticas, Técnicas y Procedimientos (TTP):

- T1053 - Programación de tareas
- T1060 - Claves de ejecución en el registro
- T1031 - Modificar servicio existente
- T1129 - Ejecución mediante carga de módulos

Comportamientos e IOCs:

- Recursos del PE en idioma chino (iconos, metadata)
- Asigna memoria RWX, posible descompresión
- Crea archivo ejecutable C:\Windows\UNINSTAL.BAT
- Crea servicio "GrayPigeon_Hacker.com.cn" que apunta a C:\Windows\Hacker.com.cn.exe
- Busca presencia de privilegio SeDebugPrivilege
- Se instala el servicio para ejecución automática
- Crea archivo C:\Windows\bootstat.dat relacionado con Hupigon
- Intentos de borrar su propio ejecutable de %TEMP%
- Strings en chino relacionadas con comandos y fechas

Conclusión:

Hupigon.ex_ es una muestra del backdoor Hupigon o Graybird asociado a grupos de amenaza chinos. Se instala como un servicio, crea componentes adicionales, busca privilegios de depuración e intenta borrar rastros de sí mismo, comportamientos típicos de un backdoor malicioso.

4.6.1. Stabuniq

Tabla 22: Detalles de análisis de malware 6

Detalles			
Fecha de análisis	28 de abril de 2024, 02:52	**Duración**	21 segundos
Nombre	stabuniq_F31B797831B36A4877AA0FD173A7A4A2		
Tamaño	77.5KB		
Tipo ejecutable	PE32 (GUI) Intel 80386, para MS Windows		
MD5	f31b797831b36a4877aa0fd173a7a4a2		
SHA1	17db1bbaa1bf1b920e47b28c3050cbff83ab16de		
SHA256	5a0d64cc41bb8455f38b4b31c6e69af9e7fd022b0ea9ea0c32c371def24d67fb		
CRC32	9B7457E5		
ssdeep	1536:3XBp/wqLHinJ8i7zY8QiLBTaM4gTKSb4JjTKT7SEKla:3zIqLHG8GzV9laMz4h+SZl		
Resultado Yara			
Ninguno coincide			
Resultado de motores Antivirus en Virustotal			
Antivirus	**Malware detectado**	**Antivirus**	**Malware detectado**
Bkav	W32.AIDetectMalware	F-Secure	Heuristic.HEUR/AGEN.1338799
Lionic	Trojan.Win32.Ruskill.4!c	DrWeb	Trojan.Packed.22607
Cynet	Malicious (score: 100)	Zillya	Backdoor.Ruskill.Win32.1331
Skyhigh	BehavesLike.Win32.Generic.lc	TrendMicro	TROJ_STABUNIQ.A
ALYac	Gen:Trojan.ExplorerHijack.eqW@aKQzEgi	Trapmine	malicious.moderate.ml.score
Cylance	unsafe	FireEye	Generic.mg.f31b797831b36a48
VIPRE	Gen:Trojan.ExplorerHijack.eqW@aKQzEgi	Sophos	Mal/FakeAV-QN
Sangfor	Suspicious.Win32.Save.ins	Ikarus	Trojan.Win32.Diple
CrowdStrike	win/malicious_confidence_100% (W)	Google	Detected
BitDefender	Gen:Trojan.ExplorerHijack.eqW@aKQzEgi	Avira	HEUR/AGEN.1338799
K7GW	Trojan (003960211)	MAX	malware (ai score=100)

K7AntiVirus	Trojan (003960211)	Antiy-AVL	Trojan/Win32.AGeneric
Arcabit	Trojan.ExplorerHijack.EC7FC2	Kingsoft	malware.kb.a.998
VirIT	Trojan.Win32.Rootkit.ED	Gridinsoft	Trojan.Win32.Agent.vb!s1
Symantec	Trojan.Stabuniq	Xcitium	Malware@#2cofmfoz6ca3
ESET-NOD32	a variant of Win32/Injector.RVT	Microsoft	Trojan:Win32/Buniq.A
APEX	Malicious	ViRobot	Trojan.Win32.Infostealer.79360
McAfee	GenericRXFF-ZF!F31B797831B3	ZoneAlarm	HEUR:Trojan.Win32.Generic
Avast	Win32:Ruskill-FQ [Trj]	GData	Gen:Trojan.ExplorerHijack.eqW@aKQzEgi
ClamAV	Win.Trojan.Stabuniq-1	Varist	W32/Graftor.BG.gen!Eldorado
Kaspersky	HEUR:Trojan.Win32.Generic	AhnLab-V3	Backdoor/Win32.Ruskill.R34552
Alibaba	Trojan:Win32/Buniq.1c5cb5be	BitDefenderTheta	Gen:NN.ZexaCO.36802.eqW@aKQzEgi
NANO-Antivirus	Trojan.Win32.Agent.duljsv	TACHYON	Backdoor/W32.Ruskill.79360
MicroWorld-eScan	Gen:Trojan.ExplorerHijack.eqW@aKQzEgi	DeepInstinct	MALICIOUS
Emsisoft	Gen:Trojan.ExplorerHijack.eqW@aKQzEgi (B)	VBA32	BScope.Malware-Cryptor.2812

Fuente: Elaboración propia

Tabla 23: Comportamientos sospechosos de malware 6

Comportamientos sospechosos.	
TTP (Táctica, Técnica y Procedimiento)	
T1055, short: Process Injection T1057, short: Process Discovery T1045, short: Software Packing	
Consultas para el nombre de la computadora	
GetComputerNameA	computer_name: WIN10
El ejecutable utiliza un empaquetador conocido (T1045)	
packer	Armadillo v1.71
Busca procesos en ejecución potencialmente para identificar procesos de evasión de sandbox, inyección de código o volcado de memoria (T1057)	
Process32NextW	System, Registry, smss.exe, csrss.exe, wininit.exe, winlogon.exe, services.exe, lsass.exe, svchost.exe, fontdrvhost.exe, dwm.exe, upfc.exe, spoolsv.exe
Asigna permiso de ejecución a otro proceso indicativo de una posible inyección de código	

NtAllocateVirtualMemory	process_identifier: 6248 region_size: 73728 stack_dep_bypass: 0 stack_pivoted: 0 heap_dep_bypass: 0 protection: 64 (PAGE_EXECUTE_READWRITE) base_address: 0x00400000 allocation_type: 12288 (MEM_COMMIT|MEM_RESERVE) process_handle: 0x0000010c
Crea un hilo usando CreateRemoteThread en un proceso no secundario indicativo de inyección de proceso (T1055)	
CreateRemoteThread	Process 6248 created a remote thread in non-child process 4324 thread_identifier: 0 process_identifier: 4324 function_address: 0x00650000 flags: 0 stack_size: 0 parameter: 0x00640000 process_handle: 0x000002e4
Manipula la memoria de un proceso no secundario indicativo de inyección de proceso	
NtAllocateVirtualMemory	Process 6248 manipulating memory of non-child process 4324
Posible inyección de código escribiendo en la memoria de otro proceso	
WriteProcessMemory	Process 6248 injected into non-child 4324 buffer: id=KiksNSs1KzUq&varname=bHJ1f3RsaCor&comp=TFJVKis=&ver=&src=LS9Zcm8=&sec=33600&view=SH56aXhzXXJ3b35pU3RobzV+Y347NjskEUh+eml4c0tpdG90eHR3U3RobzV+Y347NjskEWhvenludXJqRF0oKlksIiwjKCpZKC1aLyMsLFpaK11fKiwoWixaL1yQRaG14c3RobzV+Y347NjskEWht… base_address: 0x00660000 process_identifier: 4324 process_handle: 0x000002e4
Inyección de código escribiendo un ejecutable o DLL en la memoria de otro proceso	
WriteProcessMemory	buffer: MZ□ÿÿ¸@Ø°´ Í!¸LÍ!This program cannot be run in DOS mode. $□Ó¬ßÈ²Â□È²Â□È²Â□ É□É²Â□K®Ì□Á²Â□K°□□Û²Â□È²Ã□□²Â□ - È□Ò²Â□p´Ä□É²Â□RichÈ²Â□PEL¼¥kOà ´4o□Ð@ ¬ÔÈÐH.text~²´ `.rdataê Ð¸@@.data<#à Ä@À.rsrc ä@@ base_address: 0x00400000 process_identifier: 6248 process_handle: 0x0000010c

Se utilizó NtSetContextThread para modificar un hilo en un proceso remoto indicativo de inyección de proceso.	
NtSetContextThread	Process 7104 called NtSetContextThread to modify thread in remote process 6248 registers.eip: 2006601808 registers.esp: 1703920 registers.edi: 0 registers.eax: 4234607 registers.ebp: 0 registers.edx: 0 registers.ebx: 4091904 registers.esi: 0 registers.ecx: 0 thread_handle: 0x00000108 process_identifier: 6248
Se reanudó un hilo suspendido en un proceso remoto potencialmente indicativo de inyección de proceso	
NtResumeThread	Process 7104 resumed a thread in remote process 6248 thread_handle: 0x00000108 suspend_count: 1 process_identifier: 6248
Ejecutó un proceso e inyectó código en él, probablemente mientras desempaquetaba	
NtUnmapViewOfSection, NtAllocateVirtualMemory, WriteProcessMemory, NtGetContextThread, NtSetContextThread, NtResumeThread	

Fuente: Elaboración propia

Descripción:

El malware emplea un empaquetador conocido, en este caso, Armadillo v1.71, para ocultar su carga útil y dificultar su detección. Además, realiza una serie de acciones para inyectar código malicioso en otros procesos en el sistema. Estas acciones incluyen la asignación de permisos de ejecución a procesos externos, la creación de hilos remotos en procesos no secundarios y la manipulación de la memoria de otros procesos.

Se observa la inyección de código malicioso en la memoria de otros procesos, lo que puede indicar un intento de evadir la detección y ejecutar acciones maliciosas en un entorno ajeno. También se identifica el uso de NtSetContextThread para modificar un hilo en un proceso remoto, lo que sugiere una actividad de inyección de proceso.

Además, se destaca la reanudación de hilos suspendidos en procesos remotos, un comportamiento comúnmente asociado con la inyección de proceso para activar el malware. Por

último, se detecta la ejecución de un proceso junto con la inyección de código en él, lo que sugiere actividades de desempaquetado y ejecución de la carga útil del malware.

Tácticas, Técnicas y Procedimientos (TTP):

- T1055 - Inyección de procesos.
- T1057 - Descubrimiento de procesos.
- T1045 - Empaquetado de software

Comportamientos e IOCs:

- Recopila el nombre del equipo infectado (WIN10) mediante GetComputerNameA.
- Utiliza un empaquetador conocido llamado Armadillo v1.71 (T1045).
- Enumera los procesos en ejecución, posiblemente para identificar sandboxes, procesos a inyectar o realizar volcados de memoria (T1057).
- Asigna permisos de ejecución a la memoria de otro proceso, indicativo de inyección de código.
- Crea un hilo remoto en un proceso no secundario, técnica de inyección de proceso (T1055).
- Manipula la memoria de un proceso no secundario.
- Escribe código ejecutable o una DLL en la memoria de otro proceso, inyectando efectivamente su carga maliciosa (T1055).
- Utiliza NtSetContextThread para modificar el contexto de un hilo en un proceso remoto, otra técnica de inyección (T1055).
- Reanuda un hilo suspendido en un proceso remoto, posible inyección.
- Parece ejecutar un proceso e inyectar código en él mientras se desempaqueta, probablemente para evadir la detección.

Conclusión:

stabuniq_F31B797831B36A4877AA0FD173A7A4A2 es un malware empaquetado que utiliza diversas técnicas para inyectar su carga maliciosa en otros procesos, posiblemente con el objetivo de mantener acceso remoto persistente al sistema comprometido. Su comportamiento ha activado numerosas detecciones de troyanos, malware genérico y backdoors por parte de los motores antivirus.

4.7.1. Dofoil

Tabla 24: Detalles de análisis de malware 7

Detalles			
Fecha de análisis	28 de abril de 2024, 05:31	**Duración**	129 segundos
Nombre	US_Airways_E-Ticket_Print_Doc.exe		
Tamaño	92.0KB		
Tipo ejecutable	PE32 (GUI) Intel 80386, para MS Windows		
MD5	53fec0c29fb30de88c9a6a369e8cae62		
SHA1	597b9aab24a0bffce34407f5ea8c5082dc4bb3b2		
SHA256	4bb1d2130bb7ac35a03eb2f1eb483fc74103cea2086f3fc6984cb8724bcbcbfc		
CRC32	1414374A		
ssdeep	1536:IKt4CZ0XlXogPZEtEFNxVJz4sqAbyYPNwHBTUeC3R0FrPQxnLiaO:34CZ4YghAqVFX3PahTULmkn		
Resultado Yara			
Ninguno coincide			
Resultado de motores Antivirus en Virustotal			
Antivirus	**Malware detectado**	**Antivirus**	**Malware detectado**
Bkav	W32.AIDetectMalware	F-Secure	Trojan.TR/Kryptik.bqup.8
Lionic	Trojan.Win32.Dofoil.a!c	DrWeb	BackDoor.Kuluoz.4
Cynet	Malicious (score: 100)	Zillya	Downloader.Dofoil.Win32.509
CAT-QuickHeal	TrojanDownloader.Kuluoz.D4	TrendMicro	BKDR_KULOUZ.SME
Skyhigh	Generic.rl	Trapmine	malicious.high.ml.score
ALYac	Trojan.Dropper.WEK	FireEye	Generic.mg.53fec0c29fb30de8
Cylance	unsafe	Sophos	Mal/Generic-S
VIPRE	Trojan.Dropper.WEK	Ikarus	Trojan-Spy.Zbot
Sangfor	Trojan.Win32.Save.a	Jiangmin	TrojanDownloader.Dofoil.mq
BitDefender	Trojan.Dropper.WEK	Webroot	Trojan.Dropper.Gen
K7GW	Trojan-Downloader (003a8f751)	Google	Detected
K7AntiVirus	Trojan-Downloader (003a8f751)	Avira	TR/Kryptik.bqup.8
Arcabit	Trojan.Dropper.WEK	MAX	malware (ai score=100)
VirIT	Trojan.Win32.Crypt2.CBWT	Antiy-AVL	Trojan[Downloader]/Win32.Dofoil
Symantec	ML.Attribute.HighConfidence	Kingsoft	Win32.Troj.Undef.a
ESET-NOD32	Win32/TrojanDownloader.Zortob.B	Xcitium	Malware@#249tg1dagbuid
APEX	Malicious	Microsoft	TrojanDownloader:Win32/Kuluoz.D
McAfee	Generic.rl	ViRobot	Trojan.Win32.Agent.94208.EM
Avast	Win32:Evo-gen [Trj]	ZoneAlarm	Trojan-Downloader.Win32.Dofoil.rnh

Kaspersky	Trojan-Downloader.Win32.Dofoil.rnh	GData	Win32.Trojan.Agent.R34MBZ
Alibaba	TrojanDownloader:Win32/Dofoil.cd2e5a60	Varist	W32/Trojan.NTXP-1039
NANO-Antivirus	Trojan.Win32.Packed.cqvqzu	AhnLab-V3	Downloader/Win32.Dofoil.R96012
MicroWorld-eScan	Trojan.Dropper.WEK	BitDefenderTheta	Gen:NN.ZexaF.36804.fqW@aCekXgdi
Rising	Malware.FakeXLS/ICON!1.9C3D (CLASSIC)	DeepInstinct	MALICIOUS
Emsisoft	Trojan.Dropper.WEK (B)	VBA32	TrojanDownloader.Dofoil

Fuente: Elaboración propia

Tabla 25: Comportamientos sospechosos de malware 7

Comportamientos sospechosos.	
TTP (Táctica, Técnica y Procedimiento)	
T1129, short: Execution through Module Load	
Comprueba si un depurador está depurando el proceso	
IsDebuggerPresent	
Recopila información para tomar huellas dactilares del sistema (MachineGuid, DigitalProductId, SystemBiosDate)	
registry	HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Microsoft\Windows NT\CurrentVersion\InstallDate
Asigna memoria de lectura, escritura y ejecución (normalmente para descomprimirse)	
NtAllocateVirtualMemory	
Consulta el tamaño del disco que podría usarse para detectar máquinas virtuales con un tamaño fijo pequeño o asignación dinámica	
GetDiskFreeSpaceExW	root_path: C:\Windows, root_path: C:\Windows\system32
Crea archivos ejecutables en el sistema de archivos (T1129)	
file C:\Users\windows10\AppData\Local\qcdiinvq.exe	
Comprueba las direcciones de los adaptadores que se pueden utilizar para detectar interfaces de red virtuales	
GetAdaptersAddresses	
Uno o más de los buffers contienen un archivo PE incrustado	
Buffer with sha1: 56a509700aa2fec029e04346df910a60e7fe3d35	
Asigna permiso de ejecución a otro proceso indicativo de una posible inyección de código	
NtAllocateVirtualMemory	process_identifier: 6656 region_size: 81920 stack_dep_bypass: 0 stack_pivoted: 0

	heap_dep_bypass: 0 protection: 64 (PAGE_EXECUTE_READWRITE) base_address: 0x00400000 allocation_type: 12288 (MEM_COMMIT|MEM_RESERVE) process_handle: 0x000001a0
Posible inyección de código escribiendo en la memoria de otro proceso	
WriteProcessMemory	buffer: MZ□ÿÿ¸@È°´ Í!¸LÍ!This program cannot be run in DOS mode. $a¯Ã%Î\|□%Î\|□%Î\|□¸¶ï□&Î\|□%Î}□-Î\|□J¸×□$Î|□J¸á□$Î\|□Rich%Î\|□PELr5¨Rà □0@@@□È"(0X$.text¬ à.relocú0@B base_address: 0x00400000 process_identifier: 6656 process_handle: 0x000001a0
Inyección de código escribiendo un ejecutable o DLL en la memoria de otro proceso	
WriteProcessMemory	buffer: MZ□ÿÿ¸@È°´ Í!¸LÍ!This program cannot be run in DOS mode. $a¯Ã%Î\|□%Î\|□%Î\|□¸¶ï□&Î\|□%Î}□-Î\|□J¸×□$Î|□J¸á□$Î\|□Rich%Î\|□PELr5¨Rà □0@@@□È"(0X$.text¬ à.relocú0@B base_address: 0x00400000 process_identifier: 6656 process_handle: 0x000001a0
Se utilizó NtSetContextThread para modificar un hilo en un proceso remoto indicativo de inyección de proceso.	
NtSetContextThread	Process 6992 called NtSetContextThread to modify thread in remote process 6656 registers.eip: 0 registers.esp: 0 registers.edi: 0 registers.eax: 4267408 registers.ebp: 0 registers.edx: 0 registers.ebx: 3334144 registers.esi: 0 registers.ecx: 0 thread_handle: 0x000001b8 process_identifier: 6656
Se reanudó un hilo suspendido en un proceso remoto potencialmente indicativo de inyección de proceso	
NtResumeThread	thread_handle: 0x000001b8 suspend_count: 1 process_identifier: 6656
Ejecutó un proceso e inyectó código en él, probablemente mientras desempaquetaba	
NtGetContextThread, NtUnmapViewOfSection, NtAllocateVirtualMemory, WriteProcessMemory, NtSetContextThread, NtResumeThread, NtMapViewOfSection	

Fuente: Elaboración propia

Descripción:

Se identifican múltiples acciones consistentes con tácticas de inyección de código, como la asignación de permisos de ejecución a otros procesos, la manipulación de la memoria de procesos externos y la inyección de código malicioso en la memoria de otros procesos.

Además, el malware recopila información del sistema, como la huella dactilar del sistema, mediante la consulta del registro y la verificación de adaptadores de red. Esto indica un intento de recopilar información sobre el entorno del sistema para adaptar sus actividades maliciosas.

Se observa también la modificación de hilos en procesos remotos, lo que sugiere una actividad de inyección de proceso. La reanudación de hilos suspendidos en procesos remotos también es indicativa de posibles intentos de inyección de código.

Finalmente, se detecta la ejecución de un proceso junto con la inyección de código en él, lo que sugiere actividades de desempaquetado y ejecución de la carga útil del malware.

Tácticas, Técnicas y Procedimientos (TTP):

T1129 - Ejecución mediante carga de módulos

Comportamientos e IOCs:

- Comprueba si hay un depurador adjunto al proceso mediante IsDebuggerPresent, posible evasión de análisis.
- Recopila información de identificación del sistema como MachineGuid, DigitalProductId, SystemBiosDate.
- Asigna memoria RWX (lectura, escritura, ejecución), típico de malware empaquetado para descomprimirse.
- Consulta el espacio libre en disco, posible detección de entornos virtualizados.
- Crea un archivo ejecutable en C:\Users\windows10\AppData\Local\qcdiinvq.exe (T1129).
- Comprueba las direcciones de los adaptadores de red, posible detección de interfaces virtuales.
- Contiene un archivo PE incrustado, probablemente su carga maliciosa empaquetada.
- Asigna permisos de ejecución a la memoria de otro proceso, indicativo de inyección de código.
- Escribe un ejecutable o DLL en la memoria de otro proceso, inyectando su carga.

- Utiliza NtSetContextThread para modificar el contexto de hilos en procesos remotos, técnica de inyección.
- Reanuda hilos suspendidos en procesos remotos, posible inyección.
- Ejecuta un proceso e inyecta código en él, probablemente mientras se desempaqueta para evadir detección.

Conclusión:

US_Airways_E-Ticket_Print_Doc.exe es un malware empaquetado que implementa técnicas de inyección de código en otros procesos, creando archivos ejecutables y recopilando información del sistema. Su comportamiento sugiere que se trata de un troyano downloader o puerta trasera diseñado para entregar una carga maliciosa adicional mientras intenta evadir la detección.

5. Discusión de resultados de la metodología propuesta

En esta sección se presentan los resultados obtenidos a partir de la aplicación de la metodología propuesta para el análisis de malware. A lo largo del proyecto, se implementaron diversas técnicas y herramientas para identificar, clasificar y comprender el comportamiento del malware, abarcando tanto análisis estático como dinámico. Se analizaron múltiples muestras de malware representativas, como Cerber, WannaCry, NSIS, Alina, Hupigon, Stabuniq y Dofoil, con el fin de evaluar la eficacia de la metodología y su capacidad para detectar y mitigar amenazas.

5.1. Lecciones aprendidas

La implementación de una metodología para el análisis de malware ha demostrado una serie de lecciones que son esenciales para mejorar continuamente las prácticas de seguridad cibernética. A lo largo del desarrollo y las pruebas de la metodología propuesta, se han identificado varios aspectos importantes que deben ser considerados en futuras implementaciones y estudios.

Importancia de la virtualización y aislamiento: La virtualización se ha demostrado como una herramienta invaluable en el análisis de malware. El uso de máquinas virtuales (VMs) y sandboxes proporciona un entorno seguro para ejecutar y observar el comportamiento del malware sin riesgo para los sistemas anfitriones. El aislamiento adecuado de estas máquinas es fundamental para prevenir cualquier posible escape del malware hacia otros sistemas de la red.

Diversificación de técnicas de análisis: Combinar análisis estático y dinámico permite una visión más completa del malware. Mientras que el análisis estático facilita la identificación de patrones y firmas en el código del malware, el análisis dinámico permite observar su comportamiento en tiempo real. La integración de ambas técnicas es esencial para una detección y mitigación efectiva.

Adaptabilidad a nuevas amenazas: Las amenazas evolucionan rápidamente, y la metodología de análisis debe ser flexible para adaptarse a nuevos tipos de malware y técnicas de evasión. La observación de casos como WannaCry y Cerber ha demostrado que los atacantes continuamente mejoran sus métodos, haciendo necesario que las técnicas de análisis también evolucionen constantemente.

Eficacia de herramientas especializadas: Las herramientas específicas para análisis de malware, como aquellas que proporcionan funciones de des compilación y monitoreo en tiempo real, han mostrado ser extremadamente útiles. El uso de software especializado en combinación con MISP (Malware Information Sharing Platform) mejora significativamente la capacidad de detectar, analizar y compartir información sobre amenazas avanzadas.

Necesidad de actualización continua: La información sobre amenazas debe actualizarse continuamente. Las bases de datos de indicadores de compromiso (IoC) y las firmas de virus deben mantenerse al día para asegurar que las nuevas variantes de malware sean detectadas eficientemente. La colaboración y el intercambio de información a través de plataformas como MISP son esenciales para mantenerse al día con las últimas amenazas.

Desafíos en la evasión de detección: El estudio ha demostrado que las técnicas de evasión de análisis y detección utilizadas por el malware son cada vez más sofisticadas. Esto incluye la detección de entornos virtualizados y la manipulación de APIs del sistema para evitar su análisis. Desarrollar contramedidas efectivas contra estas técnicas es un desafío constante que requiere investigación y desarrollo continuos.

El análisis de diversas muestras de malware mediante una metodología bien estructurada ha resaltado la necesidad de una infraestructura de análisis adaptable, la actualización continua de herramientas y conocimientos, y la importancia de la colaboración entre organizaciones para mejorar la seguridad cibernética global. Estas lecciones aprendidas son fundamentales para el desarrollo de estrategias más efectivas en la lucha contra las amenazas cibernéticas emergentes.

5.2. Recomendaciones prácticas

Basado en el análisis detallado del malware y las lecciones aprendidas durante este estudio, se presentan las siguientes recomendaciones prácticas para mejorar la detección, análisis y mitigación de amenazas de malware.

Colaboración y compartición de información: Es importante colaborar con otras organizaciones y compartir información sobre amenazas a través de plataformas como MISP que pueden integrarse en entornos controlados y compartir datos en tiempo real. Esto permite a las organizaciones beneficiarse de la inteligencia colectiva y estar mejor preparadas para enfrentar nuevas amenazas.

Políticas de seguridad rigurosas: Desarrollar y mantener políticas de seguridad rigurosas que incluyan la gestión de parches, control de acceso, y la aplicación de principios de mínimo privilegio en entornos controlados puede reducir significativamente la superficie de ataque. Asegurarse de que todas las aplicaciones y sistemas operativos estén actualizados reduce las vulnerabilidades explotables por el malware.

Pruebas regulares y auditorías de seguridad: Realizar pruebas regulares y auditorías de seguridad en la infraestructura tecnológica del entorno controlado ayuda a identificar y corregir vulnerabilidades antes de que puedan ser explotadas por muestras de malware. Esto incluye pruebas de penetración y simulaciones de ataques para evaluar la eficacia de las medidas de seguridad implementadas.

Preparación y respuesta a incidentes: Establecer un plan de respuesta a incidentes bien definido que incluya procedimientos claros para la identificación, contención, erradicación y recuperación ante incidentes de malware. La preparación adecuada y la práctica regular de estos planes pueden minimizar el impacto de un ataque de malware si este lograse escapar del entorno controlado.

Estas recomendaciones prácticas buscan fortalecer la postura de seguridad de las organizaciones frente a las amenazas de malware, promoviendo una cultura de seguridad proactiva y resiliente, mediante el uso de entornos controlados para el análisis estático y dinámico de malware.

5.3. Futuras líneas de investigación

La evolución constante del malware y las técnicas utilizadas por los atacantes requieren que la investigación en este campo continúe desarrollándose. A continuación, se presentan algunas líneas de investigación que podrían ofrecer avances significativos en la detección, análisis y mitigación de amenazas de malware.

Inteligencia Artificial y Machine Learning: La aplicación de técnicas avanzadas de inteligencia artificial (IA) y machine learning (ML) tiene un potencial enorme para mejorar la detección y clasificación de malware. Investigaciones futuras podrían enfocarse en desarrollar algoritmos que identifiquen patrones complejos y comportamientos anómalos que los métodos tradicionales no detectan. Además, la combinación de IA con análisis de Big Data puede proporcionar un entendimiento más profundo sobre las tácticas, técnicas y procedimientos (TTP) utilizados por los atacantes.

Deep Learning y Redes Neuronales: El uso de Deep Learning y redes neuronales para la detección y clasificación de malware es una línea prometedora. Estas tecnologías pueden aprender y adaptarse a nuevas amenazas en tiempo real, mejorando significativamente las tasas de detección. Futuras investigaciones podrían explorar arquitecturas de redes neuronales específicas para el análisis de malware y técnicas de aprendizaje supervisado y no supervisado.

Seguridad en la nube y malware: Dado el aumento en la adopción de servicios en la nube, investigar cómo se puede detectar y mitigar el malware en estos entornos es esencial. Esto incluye el desarrollo de técnicas para proteger infraestructuras de nube híbrida y pública, así como la creación de herramientas de análisis específicas para entornos virtualizados.

Malware como servicio: Una de las áreas emergentes que merece una atención especial en futuras investigaciones es el concepto de Malware como servicio o Malware as a Service (MaaS). Este modelo, inspirado en la creciente popularidad de los servicios en la nube, permite a los ciberdelincuentes alquilar y utilizar malware sofisticado sin necesidad de tener conocimientos técnicos avanzados. MaaS representa una amenaza significativa debido a su accesibilidad y al bajo costo de entrada para los atacantes potenciales, lo que facilita la proliferación de ataques cibernéticos.

Estas futuras líneas de investigación buscan anticipar y mitigar las amenazas emergentes, garantizando que las defensas contra el malware sean robustas y adaptativas frente a un panorama de ciber amenazas en constante evolución.

6. Bibliografía

Anderson, H. S., Woodbridge, J., & Filar, B. (06 de 10 de 2016). *DeepDGA: Adversarially-Tuned Domain Generation and Detection.* Obtenido de https://doi.org/10.48550/arXiv.1610.01969

Arntz, P. (22 de 07 de 2015). *Introduction to Alternate Data Streams.* Obtenido de https://www.malwarebytes.com/blog/news/2015/07/introduction-to-alternate-data-streams

Bayer, U., Comparetti, P. M., Hlauschek, C., Kruegel, C., & Kirda, E. (2009). *Scalable, Behavior-Based Malware Clustering.* Obtenido de http://hdl.handle.net/20.500.12708/52860

Bhojani, N. (10 de 2014). *Malware Analysis.* Obtenido de http://dx.doi.org/10.13140/2.1.4750.6889

Brinkmann, M. (19 de 09 de 2017). *First Chrome extension with JavaScript Crypto Miner detected.* Obtenido de https://www.ghacks.net/2017/09/19/first-chrome-extension-with-javascript-crypto-miner-detected/

Chen, J. C. (16 de 07 de 2018). *New Andariel Reconnaissance Tactics Uncovered.* Obtenido de https://www.trendmicro.com/en_us/research/18/g/new-andariel-reconnaissance-tactics-hint-at-next-targets.html

Chen, J., Sasson, A., & Zelivansky, A. (03 de 02 de 2021). *Hildegard: New TeamTNT Cryptojacking Malware Targeting Kubernetes.* Obtenido de https://unit42.paloaltonetworks.com/hildegard-malware-teamtnt/

Cohen, S., Bitton, R., & Nassi, B. (05 de 03 de 2024). *Here Comes The AI Worm: Unleashing Zero-click Worms that Target GenAI-Powered Applications.* Obtenido de https://doi.org/10.48550/arXiv.2403.02817

Delpy, e., & TOUX, V. L. (2018). *DCShadow.* Obtenido de https://www.dcshadow.com

Dumont, R., M.Léveillé, M.-E., & Porcher, H. (12 de 2018). *The Dark Side of the ForSSHe A landscape of OpenSSH backdoors.* Obtenido de https://web-assets.esetstatic.com/wls/2018/12/ESET-The_Dark_Side_of_the_ForSSHe.pdf

Erica Eng, D. C. (15 de 06 de 2015). *Operation Clandestine Wolf – Adobe Flash Zero-Day in APT3 Phishing Campaign.* Obtenido de https://www.mandiant.com/resources/blog/operation-clandestine-wolf-adobe-flash-zero-day

ESET. (2022). *ESET Threat Report T1 2022.* Obtenido de https://www.eset.com/int/business/resource-center/reports/eset-threat-report-t1-2022/

ESET. (02 de 2022). *Industry Report on Retail: Evolving threats to data and payments.* Obtenido de https://www.eset.com/int/business/resource-center/reports/eset-retail-industry-report/

ESET. (2023). *APT Activity Report Q2 2023-Q3 2023: Government Espionage and Unpatched Vulnerabilities.* Obtenido de https://www.eset.com/us/business/resource-center/reports/eset-apt-activity-report-q2-2023-q3-2023/

ESET. (2023). *ESET Threat Report H2 2023.* Obtenido de https://web-assets.esetstatic.com/wls/en/papers/threat-reports/eset-threat-report-h22023.pdf

ESET. (2023). *Security Report Latinoamérica.* Obtenido de https://web-assets.esetstatic.com/wls/es/articulos/reportes/eset-security-report-latam2023.pdf

Faou, M. (10 de 09 de 2023). *MoustachedBouncer: Espionage against foreign diplomats in Belarus.* Obtenido de https://www.welivesecurity.com/en/eset-research/moustachedbouncer-espionage-against-foreign-diplomats-in-belarus/

Fernandez, R. J. (2015). *Comparación de performance de Hipervisores.* Obtenido de https://rdu.iua.edu.ar/handle/123456789/629

Gandotra, E., Bansal, D., & Sofat, S. (04 de 2014). *Malware Analysis and Classification: A Survey.* Obtenido de http://dx.doi.org/10.4236/jis.2014.52006

García Monje, R. A. (10 de 10 de 2017). *Seguridad Informatica y el malware.* Obtenido de http://repository.unipiloto.edu.co/handle/20.500.12277/2641

Gatlan, S. (03 de 07 de 2019). *New Godlua Malware Evades Traffic Monitoring via DNS over HTTPS.* Obtenido de https://www.bleepingcomputer.com/news/security/new-godlua-malware-evades-traffic-monitoring-via-dns-over-https/

Golovanov, S. (06 de 12 de 2018). *DarkVishnya: Banks attacked through direct connection to local network.* Obtenido de https://securelist.com/darkvishnya/89169/

Harbison, M. (09 de 02 de 2021). *BendyBear: Novel Chinese Shellcode Linked With Cyber Espionage Group BlackTech.* Obtenido de https://unit42.paloaltonetworks.com/bendybear-shellcode-blacktech/

Hell., S. P. (03 de 2023). *Using GPT to encode and mutate computer virusesentirely in natural language.* Obtenido de https://github.com/SPTHvx/SPTH/blob/master/articles/files/LLMorpher.txt

Holmes, G. (8 de 10 de 2015). *Evolution of attacks on Cisco IOS devices.* Obtenido de https://blogs.cisco.com/security/evolution-of-attacks-on-cisco-ios-devices

Hosseini, A. (18 de 07 de 2017). *Ten process injection techniques: A technical survey of common and trending process injection techniques.* Obtenido de https://www.elastic.co/es/blog/ten-process-injection-techniques-technical-survey-common-and-trending-process

Hrčka, V. (01 de 01 de 2021). *FontOnLake.* Obtenido de https://web-assets.esetstatic.com/wls/2021/10/eset_fontonlake.pdf

Hromcová, Z. (06 de 08 de 2021). *IIStealer: malware en servidores IIS que afecta a sitios web de comercio electrónico.* Obtenido de https://www.welivesecurity.com/la-es/2021/08/06/iistealer-malware-servidores-iis-afecta-sitios-web-comercio-electronico/

Hyppönen, M. (03 de 04 de 2023). *Malware and machine learning: A match made in hell.* Obtenido de https://www.helpnetsecurity.com/2023/04/03/machine-learning-malware/

IBM. (26 de 04 de 2017). *Storwize USB Initialization Tool may contain malicious code.* Obtenido de https://www.ibm.com/support/pages/storwize-usb-initialization-tool-may-contain-malicious-code

Intelligence, F. T. (01 de 12 de 2015). *China-based Cyber Threat Group Uses Dropbox for Malware Communications and Targets Hong Kong Media Outlets.* Obtenido de https://www.mandiant.com/resources/blog/china-based-threat

Konoth, R. K., Wegberg, R. v., Moonsamy, V., & Bos, H. (29 de 01 de 2019). *Malicious cryptocurrency miners: Status and Outlook.* Obtenido de https://doi.org/10.48550/arXiv.1901.10794

Kubovič, O. (08 de 2021). *Ransomware: A look at the criminal art of malicious code, pressure, and manipulation.* Obtenido de https://www.eset.com/int/business/resource-center/white-papers/criminal-art-of-ransomware-white-paper/

Marvi, A., Slaybaugh, B., Ebreo, D., Ahmed, T., Umair, M., & Johnson, T. (16 de 03 de 2023). *Fortinet Zero-Day and Custom Malware Used by Suspected Chinese Actor in Espionage*

Operation. Obtenido de https://www.mandiant.com/resources/blog/fortinet-malware-ecosystem

Miao, Y. (17 de 07 de 2015). *Understanding Heuristic-based Scanning vs. Sandboxing.* Obtenido de https://www.opswat.com/blog/understanding-heuristic-based-scanning-vs-sandboxing

Microsoft. (09 de 02 de 2021). *Windows Win32k Elevation of Privilege Vulnerability.* Obtenido de https://msrc.microsoft.com/update-guide/vulnerability/CVE-2021-1732

Muir, M. (06 de 04 de 2022). *Cado Discovers Denonia: The First Malware Specifically Targeting Lambda.* Obtenido de https://www.cadosecurity.com/blog/cado-discovers-denonia-the-first-malware-specifically-targeting-lambda

NIST. (2019). *Glossary of Key Information.* Obtenido de https://doi.org/10.6028/NIST.IR.7298r3

Ongun, T., Stokes, J. W., & Or, J. B. (2021). *Living-Off-The-Land Command Detection.* Obtenido de https://doi.org/10.1145/3471621.3471858

Or-Meir, O., Nissim, N., Elovici, Y., & Rokach, L. (13 de 09 de 2019). *Dynamic Malware Analysis in the Modern Era—A State of the Art Survey.* Obtenido de https://doi.org/10.1145/3329786

Pantazopoulos, N. (17 de 06 de 2018). *Decoding network data from a Gh0st RAT variant.* Obtenido de https://research.nccgroup.com/2018/04/17/decoding-network-data-from-a-gh0st-rat-variant/

Ponce Larreategui, J. G. (2021). *Indicadores de compromiso (IOC) para detección de amenazas en la seguridad informática con enfoque en el código malicioso.* Obtenido de http://dspace.ups.edu.ec/handle/123456789/20937

Porolli, M. (12 de 01 de 2021). *Operation Spalax: Targeted malware attacks in Colombia.* Obtenido de https://www.welivesecurity.com/2021/01/12/operation-spalax-targeted-malware-attacks-colombia/

Poslušný, M. (11 de 01 de 2022). *Signed kernel drivers – Unguarded gateway to Windows' core.* Obtenido de https://www.welivesecurity.com/2022/01/11/signed-kernel-drivers-unguarded-gateway-windows-core/

Pradhan, A. (08 de 02 de 2022). *LolZarus: Lazarus Group Incorporating Lolbins into Campaigns.* Obtenido de https://blog.qualys.com/vulnerabilities-threat-research/2022/02/08/lolzarus-lazarus-group-incorporating-lolbins-into-campaigns

Raggi, M. (01 de 12 de 2021). *Injection is the New Black: Novel RTF Template Inject Technique Poised for Widespread Adoption Beyond APT Actors.* Obtenido de https://www.proofpoint.com/us/blog/threat-insight/injection-new-black-novel-rtf-template-inject-technique-poised-widespread

Ramírez Vega, J. C. (2015). *Mejora de la detección de malware mediante la modificación profunda de sistemas de sandboxing.* Obtenido de https://zaguan.unizar.es/record/48229

Rea, C. P. (04 de 2023). *Análisis de los datos enviados por una sandbox para monitorear malware en tiempo real.* Obtenido de https://repositorio.pucesa.edu.ec/handle/123456789/4138

Report, T. D. (28 de 11 de 2022). *Emotet Strikes Again – LNK File Leads to Domain Wide Ransomware.* Obtenido de https://thedfirreport.com/2022/11/28/emotet-strikes-again-lnk-file-leads-to-domain-wide-ransomware/

Research, A. T. (06 de 12 de 2018). *Pulling Linux Rabbit/Rabbot Malware Out of a Hat.* Obtenido de https://www.anomali.com/blog/pulling-linux-rabbit-rabbot-malware-out-of-a-hat

Rivera Guevara, R. (06 de 2014). *Análisis de características estáticas de ficheros ejecutables para la clasificación de Malware.* Obtenido de https://oa.upm.es/34343/

Rivera Guevara, R. P. (09 de 2018). *Detección y Clasificación de Malware con el Sistema de Análisis de Malware Cuckoo.* Obtenido de https://reunir.unir.net/handle/123456789/7444

Salem, E. (27 de 04 de 2022). *The chronicles of Bumblebee: The Hook, the Bee, and the Trickbot connection.* Obtenido de https://elis531989.medium.com/the-chronicles-of-bumblebee-the-hook-the-bee-and-the-trickbot-connection-686379311056

Sidi, L., Nadler, A., & Shabtai, A. (24 de 02 de 2019). *MaskDGA: A Black-box Evasion Technique Against DGA Classifiers and Adversarial Defenses.* Obtenido de https://doi.org/10.48550/arXiv.1902.08909

Stegger, P. G. (30 de 04 de 2021). *Malware cluster analysis.* Obtenido de https://projekter.aau.dk/projekter/files/411279330/ITEV_Master_thesis_Peter_Stegger_2021.pdf

Team, C. T. (24 de 09 de 2019). *REvil/Sodinokibi Ransomware.*

Team, C. T. (17 de 08 de 2022). *DarkTortilla Malware Analysis.* Obtenido de https://www.secureworks.com/research/darktortilla-malware-analysis

Team, S. T. (16 de 12 de 2021). *Noberus: Technical Analysis Shows Sophistication of New Rust-based Ransomware.* Obtenido de https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/noberus-blackcat-alphv-rust-ransomware

Torello, A., & Guibernau, F. (18 de 05 de 2021). *Environment Awareness.* Obtenido de https://drive.google.com/file/d/1t0jn3xr4ff2fR30oQAUn_RsWSnMpOAQc/view?usp=sharing

Uppal, D., Mehra, V., & Verma, V. (02 de 2014). *Basic survey on Malware Analysis, Tools and Techniques.* Obtenido de https://wireilla.com/papers/ijcsa/V4N1/4114ijcsa10.pdf

Villar, E., & Gómez, J. (s.f.). *Introducción a la virtualización.* Obtenido de http://biblioteca.udgvirtual.udg.mx/jspui/handle/123456789/2273

Walter, J. (10 de 08 de 2020). *Agent Tesla Old RAT Uses New Tricks to Stay on Top.* Obtenido de https://www.sentinelone.com/labs/agent-tesla-old-rat-uses-new-tricks-to-stay-on-top/

Zapata Pareja, C. A., Cubides Corrales, I. D., & Murcia Guzmán, M. O. (2015). *Técnicas de detección y análisis de malware en entornos corporativos con sistemas operativos Windows.* Obtenido de http://hdl.handle.net/10819/4208

7. Anexos

Anexo A: Tabla de referencia de las Técnicas MITRE (MITRE Techniques Reference)

ID	Nombre	Detalles de la técnica
T1012	Query Registry	Los adversarios pueden interactuar con el Registro de Windows para recopilar información sobre el sistema, la configuración y el software instalado. El Registro contiene una gran cantidad de información sobre el sistema operativo, su configuración, software y seguridad. Esta información puede ser consultada fácilmente utilizando la utilidad Reg, aunque existen otros métodos para acceder al Registro. La información obtenida a través de consultas al Registro puede ayudar a los adversarios a avanzar en sus operaciones dentro de una red. Durante la fase de descubrimiento automatizado, los adversarios pueden utilizar esta información para determinar comportamientos posteriores, incluyendo la decisión de infectar completamente el objetivo y/o llevar a cabo acciones específicas.
T1023	Shortcut Modification	Los adversarios pueden crear o modificar accesos directos para ejecutar programas maliciosos durante el arranque del sistema o el inicio de sesión del usuario, abusando de los accesos directos en la carpeta de inicio para lograr persistencia. Estos accesos directos, a menudo utilizados como cargas útiles en una cadena de infección, pueden ser manipulados para que el malware se ejecute en lugar del programa legítimo al editar la ruta de destino o reemplazar un acceso directo existente. Además, los adversarios pueden emplear técnicas de enmascaramiento para que el acceso directo malicioso parezca un programa legítimo, y también pueden modificar extensiones de navegador para lanzar persistente y automáticamente el malware.
T1031	Modify Existing Service	Los adversarios pueden crear o modificar servicios de Windows para ejecutar repetidamente cargas útiles maliciosas como parte de su persistencia. Cuando Windows se inicia, arranca programas o aplicaciones llamadas servicios que realizan funciones del sistema en segundo plano. Los adversarios pueden instalar un nuevo servicio o modificar uno existente para que se ejecute al inicio, utilizando utilidades del sistema como sc.exe, modificando directamente el Registro de Windows, o interactuando con la API de Windows. Además, pueden utilizar servicios para instalar y ejecutar controladores maliciosos, que pueden actuar como rootkits para ocultar actividades maliciosas. Los servicios maliciosos pueden incorporar técnicas de enmascaramiento para parecer legítimos y pueden ser creados con privilegios de administrador pero ejecutarse con privilegios de SISTEMA, lo que también permite la escalada de privilegios.
T1045	Software Packing	Los adversarios pueden utilizar técnicas de empaquetado de software o protección de software mediante máquinas virtuales para ocultar su código. El empaquetado de software es un método que comprime o encripta un ejecutable, alterando su firma para evitar la detección basada en firmas. La mayoría de las técnicas de

		descompresión descomprimen el código ejecutable en memoria. La protección mediante máquinas virtuales traduce el código original de un ejecutable a un formato especial que solo una máquina virtual específica puede ejecutar. Las utilidades usadas para el empaquetado se llaman empaquetadores, como MPRESS y UPX, aunque los adversarios pueden crear sus propias técnicas de empaquetado para evadir las defensas.
T1047	Windows Management Instrumentation	Los adversarios pueden abusar de la Instrumentación de Administración de Windows (WMI) para ejecutar comandos y cargas útiles maliciosas, aprovechando su capacidad para gestionar datos y operaciones en sistemas Windows. WMI permite el acceso tanto local como remoto a componentes del sistema, facilitado por servicios como DCOM y WinRM. Mediante WMI, los atacantes pueden realizar diversas acciones, como recopilar información y ejecutar comandos. Por ejemplo, pueden utilizar wmic.exe para eliminar copias sombra con el comando "wmic.exe Shadowcopy Delete". Cabe destacar que wmic.exe está en desuso desde enero de 2024 y será reemplazado por PowerShell en futuras versiones de Windows.
T1053	Scheduled Task or Job	Los adversarios pueden abusar de la funcionalidad de programación de tareas para facilitar la ejecución inicial o recurrente de código malicioso. Todos los sistemas operativos principales incluyen utilidades para programar la ejecución de programas o scripts en una fecha y hora especificadas. Es posible programar una tarea en un sistema remoto si se cuenta con la autenticación adecuada. Los atacantes pueden usar la programación de tareas para ejecutar programas al inicio del sistema o de manera periódica para mantener la persistencia. Además, pueden ejecutar procesos bajo el contexto de una cuenta con permisos elevados, enmascarando la ejecución bajo un proceso del sistema de confianza.
T1055	Process Injection	Los adversarios pueden inyectar código en procesos para evadir defensas basadas en procesos y posiblemente elevar privilegios. La inyección de procesos es un método para ejecutar código arbitrario en el espacio de direcciones de un proceso en ejecución. Ejecutar código en el contexto de otro proceso puede permitir el acceso a la memoria del proceso, a los recursos del sistema/red y posiblemente a privilegios elevados. La ejecución mediante inyección de procesos también puede evadir la detección de productos de seguridad, ya que la ejecución se enmascara bajo un proceso legítimo. Existen muchas formas de inyectar código en un proceso, muchas de las cuales abusan de funcionalidades legítimas y son específicas de cada plataforma. Los ejemplos más sofisticados pueden realizar múltiples inyecciones de procesos para segmentar módulos y evadir aún más la detección, utilizando tuberías con nombre u otros mecanismos de comunicación entre procesos (IPC) como canal de comunicación.
T1057	Process Discovery	Los adversarios pueden intentar obtener información sobre los procesos en ejecución en un sistema. La información obtenida podría utilizarse para comprender mejor el software y las aplicaciones comunes que se ejecutan en los sistemas dentro de la red. El acceso de administrador o con privilegios elevados puede proporcionar

		detalles más completos de los procesos. Los adversarios pueden usar esta información durante el descubrimiento automatizado para ajustar sus comportamientos posteriores, incluyendo la decisión de infectar completamente el objetivo o intentar acciones específicas. En entornos Windows, los adversarios podrían obtener detalles de los procesos en ejecución utilizando la utilidad Tasklist a través de cmd o Get-Process a través de PowerShell. En Mac y Linux, esto se logra con el comando ps, y en dispositivos de red, comandos CLI como show processes pueden usarse para mostrar los procesos en ejecución.
T1060	Registry Run Keys / Startup Folder	Los adversarios pueden lograr persistencia añadiendo un programa a una carpeta de inicio o referenciándolo con una clave de ejecución en el Registro. Agregar una entrada a las "claves de ejecución" en el Registro o a la carpeta de inicio hará que el programa referenciado se ejecute cuando un usuario inicie sesión. Estos programas se ejecutarán bajo el contexto del usuario y tendrán el nivel de permisos asociado a la cuenta. Las claves de ejecución del Registro pueden existir en múltiples colmenas, y las ubicaciones de las carpetas de inicio pueden ser específicas del usuario o del sistema. Estas ubicaciones se pueden utilizar para ejecutar malware, como herramientas de acceso remoto, para mantener la persistencia a través de reinicios del sistema. Los adversarios también pueden usar el enmascaramiento para que las entradas del Registro parezcan asociadas con programas legítimos.
T1071	Standard Application Layer Protocol	Los adversarios pueden comunicarse utilizando protocolos de capa de aplicación OSI para evitar la detección y el filtrado de red al mezclarse con el tráfico existente. Los comandos al sistema remoto, y a menudo los resultados de esos comandos, se incrustarán dentro del tráfico del protocolo entre el cliente y el servidor. Los adversarios pueden utilizar diferentes protocolos, como los utilizados para navegación web, transferencia de archivos, correo electrónico o DNS. Para conexiones que ocurren internamente dentro de un enclave, como aquellas entre un nodo proxy o de pivote y otros nodos, se utilizan comúnmente protocolos como SMB, SSH o RDP.
T1082	System Information Discovery	El Descubrimiento de información del sistema es una técnica utilizada por los adversarios para obtener detalles sobre el sistema operativo y hardware de una máquina objetivo, incluyendo versión, parches, actualizaciones y arquitectura. Esta información puede ser recopilada mediante herramientas como Systeminfo o comandos como df -aH, y permite a los atacantes adaptar sus acciones y cargas maliciosas en base a las características específicas del sistema comprometido. Al combinarse con otros datos de reconocimiento, el Descubrimiento de información del sistema facilita el desarrollo de ataques más efectivos y el ocultamiento de actividades maliciosas en la máquina víctima.
T1083	File and Directory Discovery	El Descubrimiento de archivos y directorios es una técnica mediante la cual los adversarios enumeran y buscan archivos y directorios específicos en el sistema de archivos de una máquina o recurso de red objetivo. Utilizan utilidades de la línea de comandos como dir, tree, ls, find y locate, además de herramientas personalizadas,

		para obtener esta información. Los atacantes pueden aprovechar el Descubrimiento de archivos y directorios durante el reconocimiento automatizado para determinar sus próximos pasos, como si deben infectar completamente el objetivo o intentar acciones específicas. Algunos archivos y directorios pueden requerir permisos elevados o especiales para acceder a ellos.
T1089	Disabling Security Tools	Los adversarios pueden modificar o deshabilitar herramientas de seguridad para evitar la detección de su malware/herramientas y actividades. Esto puede implicar matar procesos o servicios de seguridad, modificar claves del Registro o archivos de configuración para que las herramientas no funcionen correctamente, u otros métodos para interferir con el escaneo o reporte de información por parte de las herramientas de seguridad. Los atacantes también pueden deshabilitar actualizaciones para evitar que los últimos parches de seguridad lleguen a las herramientas en los sistemas víctima. Además, pueden alterar artefactos desplegados por las herramientas de seguridad, como desconectar o modificar características agregadas por estas herramientas para evadir la detección. Esta técnica apunta a deteriorar las defensas de ciberseguridad.
T1112	Modify Registry	Los adversarios pueden interactuar con el Registro de Windows para ocultar información de configuración dentro de las claves del Registro, eliminar información como parte de la limpieza, o como parte de otras técnicas para facilitar la persistencia y ejecución. Esto se realiza mediante utilidades como reg.exe o herramientas de acceso remoto que interactúan con la API de Windows. Las modificaciones pueden incluir ocultar claves anteponiendo caracteres nulos para causar errores, abusar de estas claves pseudo-ocultas para ocultar cargas persistentes, o modificar el Registro remoto de otros sistemas para facilitar la ejecución lateral de archivos. Dependiendo de los permisos de cuenta, algunas áreas del Registro requieren acceso de administrador. Esta técnica permite a los atacantes mantener acceso encubierto y persistencia en los sistemas comprometidos.
T1129	Shared Modules	Los adversarios pueden ejecutar cargas útiles maliciosas cargando módulos compartidos. Los módulos compartidos son archivos ejecutables que se cargan en procesos para proporcionar acceso a código reutilizable, como funciones personalizadas o invocar funciones de la API del sistema operativo. Los atacantes pueden modularizar la funcionalidad de su malware en objetos compartidos que realizan varias tareas, como administrar las comunicaciones de C2 o ejecutar acciones específicas. En Linux y macOS, el cargador de módulos puede cargar y ejecutar objetos compartidos desde rutas locales arbitrarias utilizando funciones como dlopen y dlsym. En Windows, el cargador de módulos puede cargar DLLs desde rutas locales y rutas de red UNC arbitrarias mediante la API nativa de Windows y funciones como LoadLibrary. Esta técnica permite a los adversarios ejecutar cargas útiles maliciosas de manera encubierta.
T1158	Hidden Files and Directories	Los adversarios pueden configurar archivos y directorios como ocultos para evadir mecanismos de detección. La mayoría de los sistemas operativos tienen el concepto

		de "archivo oculto" para evitar que los usuarios modifiquen accidentalmente archivos especiales. Estos archivos no se muestran cuando un usuario navega por el sistema de archivos con una GUI o comandos normales. Los usuarios deben solicitar explícitamente mostrar los archivos ocultos a través de la interfaz gráfica o con switches de línea de comandos. En Linux y macOS, los archivos que comienzan con un "." están ocultos de forma predeterminada. En Windows, se pueden marcar archivos como ocultos con attrib.exe. Los adversarios pueden aprovechar esta funcionalidad para ocultar archivos y directorios en cualquier parte del sistema, evadiendo un análisis típico que no investigue archivos ocultos.
T1188	Multi-hop Proxy	Los adversarios pueden encadenar múltiples proxies para disfrazar el origen del tráfico malicioso. Esto dificulta que los defensores identifiquen la fuente original, ya que deben rastrear el tráfico a través de varios proxies. Por ejemplo, los atacantes pueden usar redes de enrutamiento de cebolla como Tor para transportar tráfico de C2 encriptado a través de una población comprometida. En infraestructuras de red, es posible que un adversario aproveche múltiples dispositivos comprometidos para crear una cadena de proxy multinivel, implementando el enrutamiento de cebolla entre esos nodos. De manera similar, los adversarios pueden abusar de infraestructuras peer-to-peer y blockchain para implementar el enrutamiento entre una red descentralizada de pares. Esta técnica oculta aún más el origen del tráfico malicioso.
T1204	User Execution	La Ejecución por el Usuario ocurre cuando un adversario depende de acciones específicas de un usuario para obtener ejecución de código malicioso. Los usuarios pueden ser objeto de ingeniería social para que ejecuten código dañino, como abrir un archivo o enlace malicioso. Estas acciones del usuario suelen observarse como un comportamiento posterior al Phishing inicial. Los atacantes también pueden engañar a los usuarios para que habiliten software de acceso remoto, ejecuten JavaScript malicioso en su navegador o descarguen y ejecuten malware. Por ejemplo, las estafas de soporte técnico pueden facilitarse a través de suplantación de identidades y números promocionados que dirigen a las víctimas a sitios web maliciosos para entregar cargas útiles. Esta técnica explota las acciones inseguras de los usuarios para comprometer sus sistemas.