Análisis Forense de una APT

Trabajo Fin de Máster

Máster Universitario en Ciberseguridad

Realizado por: Jesús Ramos García
Dirigido por: María de los Llanos Tobarra Abad
Co-tutor: Antonio Robles Gómez

Resumen

Desde 2004, Mandiant ha examinado problemas de seguridad informática en muchas organizaciones de todo el mundo. La mayoría de estas infracciones de seguridad se deben a actores de amenazas avanzadas denominados Advanced Persistent Threat (APT), amenaza permanente avanzada, en español. Los datos que han sido analizados durante cientos de investigaciones nos convencen de que los grupos que realizan estas actividades se realizan en China.

En cuanto a la descripción del trabajo realizado, existe un APT conocido como "APT1", y es uno de más de 20 grupos APT con orígenes en China. APT1 es una organización única de atacantes que ha llevado a cabo una campaña de ciberespionaje contra muchas víctimas desde por lo menos 2006. Según las observaciones, es uno de los grupos de espionaje cibernético con más éxito en términos de la gran cantidad de información robada.

Aunque la visibilidad de las actividades de APT1 no es completa, han sido analizadas las intrusiones del grupo contra más o menos 150 víctimas durante siete años. Fue descubierta una gran cantidad de la infraestructura de ataque de APT1 (herramientas, tácticas y procedimientos).

Creemos que APT1 es capaz de emprender una campaña de ciberespionaje de muy larga duración. En gran parte porque recibe apoyo directo del gobierno. Al tratar de identificar la organización detrás de esta actividad, la investigación encontró que la Unidad 61398 del Ejército Popular de Liberación (EPL) es similar a APT1 en su misión, capacidades y recursos. La Unidad PLA 61398 también está precisamente en la misma área desde la que parece originarse la actividad APT1.

La unidad 61398 está situada en Datong Road (大同路) en Gaoqiaozhen (高桥镇), que se encuentra en Pudong (浦东新区) de Shanghai (上海). El edificio principal de esta infraestructura es una instalación de 130,663 pies cuadrados con 12 pisos de altura y fue construido a principios de 2007.

En cuanto a la descripción del trabajo realizado, es un trabajo de análisis y de aplicación de una serie de aspectos relacionados con el APT1, abarcando tanto aspectos teóricos como prácticos. Dichos aspectos se explicarán de forma ordenada a lo largo de esta memoria, dividida en bloques bastante diferenciados entre sí y con temas muy concretos explicados en dichos bloques.

Palabras clave

Mandiant, APT1, JIB, SilK, Internet Census 2012, fingerprint, synscan, service probe, WEBC2-DIV.

Abstract

Since 2004, Mandiant has examined information security issues in many organizations around the world. Most of these security breaches are due to advanced threat actors called Advanced Persistent Threat (APT). The data that has been analyzed during hundreds of investigations convinces us that the groups that carry out these activities are in China.

There is an APT known as "APT1", and it is one of more than 20 APT groups with origins in China. APT1 is a unique organization of attackers that has carried out a cyber espionage campaign against many victims since at least 2006. According to observations, it is one of the most successful cyber espionage groups in terms of the large amount of information stolen.

Although the visibility of APT1's activities is not complete, the group's intrusions against roughly 150 victims over seven years have been analyzed. A large amount of APT1's attack infrastructure (tools, tactics, and procedures) was discovered.

We believe that APT1 is capable of undertaking a very long-term cyber espionage campaign. Largely because it receives direct support from the government. In trying to identify the organization behind this activity, the investigation found that Unit 61398 of the Popular Liberation Army (EPL) is similar to APT1 in its mission, capabilities, and resources. PLA Unit 61398 is also in precisely the same area from which the APT1 activity appears to originate.

Unit 61398 is located at Datong Road (大同路) in Gaoqiaozhen (高桥镇), which is in Pudong (浦东新区) of Shanghai (上海). The main building of this infrastructure is a 130,663-square-foot facility which has 12 floors and was built in early 2007.

Regarding the description of the work carried out, it is a work of analysis and application of a series of aspects related to the APT1, covering both theoretical and practical aspects. These aspects will be explained in an orderly manner throughout this memory, divided into blocks that are quite different from each other and with very specific topics explained in the blocks.

Keywords

Mandiant, APT1, JIB, SilK, Internet Census 2012, fingerprint, synscan, service probe, WEBC2-DIV.

Contenido

INDICE DE FIGURAS

1 Introducción

1.1 Concepto sobre APT y su impacto en la ciberseguridad

Algo que no gusta a los profesionales en el ámbito de la ciberseguridad corporativa es tener en mente un ataque que utilice una serie de técnicas de carácter avanzado creadas para usurpar información importante de una empresa.
Con el calificativo de avanzado, las amenazas avanzadas persistentes (AAP en castellano y APT en inglés, que significa Advance Persistent Thread) se basan en métodos de hackeo prolongado, fuera de visión y avanzado para el acceso a un equipo y estar allí mucho tiempo, con resultados de carácter destructivo, como es mostrado en [12].
El "porqué" del nivel de trabajo necesario para realizar ataques de tipo APT, estas se asocian con objetivos altamente valiosos, como países y empresas importantes. Todo ello para usurpar información durante mucho tiempo, en vez de simplemente "entrar y salir" de forma rápida, actividad realizada por hackers de sombrero negro en ciberataques de bajo nivel.
Las APT suponen una ofensiva que debería estar vigilada por las empresas internacionales. Aun así, las medianas y pequeñas empresas, pueden no darse cuenta de ese ataque.
Los que realizan la APT usan con más volumen empresas pequeñas que forman parte de la cadena de suministros como objetivo final tener un medio de acceso hacia las grandes organizaciones. Por ejemplo, las empresas son utilizadas como trampolines al contar con menos protección.

El objetivo de un ataque de tipo APT es acceder de forma continua al sistema. Esto es realizado mediante una serie de etapas.

- **Etapa 1: obtener acceso.** Para introducir malware en una red, los criminales tienen una vía de acceso mediante una red, un archivo con infección, un email basura o una vulnerabilidad.
- **Etapa 2: infiltrarse.** Los criminales introducen malware para fabricar una red con "puertas traseras" y, otro tipo de "conductos", para meterse en los equipos desapercibidamente. El malware usa métodos como reinscripción de código para que sea de ayuda a la hora de ocultar datos.
- **Etapa 3: intensificar el acceso.** Una vez que los hackers están dentro, desencriptan contraseñas, aumentan el manejo sobre el sistema y tiene mejores fases de acceso.
- **Etapa 4: desplazamiento horizontal**. Debido a la incursión planeada por los hackers, estos pueden desplazarse libremente por el sistema. También pueden acceder a otros sitios como por ejemplo servidores.
- **Etapa 5: mirar, aprender y permanecer**. Desde el interior del sistema, los hackers obtienen una completa comprensión de su funcionamiento y sus vulnerabilidades, lo que les permite hacer uso de la información que desean.

Los hackers pueden intentar mantener este proceso en funcionamiento, posiblemente de manera indefinida, o retirarse después de cumplir un objetivo específico. A menudo, dejan una puerta abierta para acceder al sistema de nuevo en el futuro. Dentro del sistema, es obtenido mediante los hackers el entendimiento de cómo funciona y cuáles

son sus puntos débiles, para usar la información deseada. De esta manera los hackers pueden seguir funcionando.

La ciberseguridad corporativa es más evolucionada que la de los usuarios, debido a eso se necesita de alguien del interior para realizar el ataque. Esto no siempre es debido a que el personal sea partícipe a conciencia del ataque. El atacante usa técnicas de ingeniería social, como el “whaling” o el “spear phsising”.

Whaling consiste en una manera usada por los cibercriminales en las que ellos simulan ocupar cargos de alto nivel en una organización y, de esa manera, atacar de forma directa a altos ejecutivos y otros individuos importantes. Su objetivo es robar dinero, obtener información confidencial o conseguir acceso a sus equipos informáticos con finalidad delictiva.

El principal peligro de los ataques de APT es que incluso cuando se descubren y la amenaza inmediata pareciera haber desaparecido, los hackers podrían tener varias puertas traseras abiertas que les permitan regresar cuando lo deseen. Además, muchas ciberdefensas tradicionales, como los firewalls y antivirus, no siempre pueden proteger contra estos tipos de ataques.

Uno de los grandes daños que producen las APT es el uso de las puertas traseras, aunque el ataque parece haber desaparecido. Los métodos como firewall y antivirus no suelen ser medida de protección ante esos ataques. Toda esta información viene indicada en [12].

1.2 Análisis Forense y APTs

El análisis forense es el proceso de identificar, preservar, analizar y presentar las evidencias de forma legal y aceptable, esto se muestra en los apuntes de la asignatura análisis forense [24].
Gracias a esto se pueden buscar datos desconocidos previamente, con el objetivo de encontrar unas señas determinadas como patrones o información oculta, como se puede ver en [1].
Su importancia para combatir las APTs se basa en una monitorización de ataque mediante registros ubicados en el acceso, su conexión con internet, tráfico de red, procesos en la memoria y ficheros creados (ver [2]).

En nuestro caso concreto:

> El análisis forense es sobre el estudio detallados de las IPs relacionadas con el ataque y el estudio de un malware usado por APT1 llamado WEBC2-DIV.

En cuanto al estudio de las IPs:

> Cuánto más sepamos de las IPs que nos atacan, mejor podemos defendernos, por ejemplo, si nos ataca una dirección IP que no figuran en el estudio pues no es un ataque de tipo APT1.
>
> A lo mejor podemos saber el sistema operativo de la dirección atacante, si esta viene “escondida”, pero sabemos el sistema operativo del cuál es ejecutado ya sabemos que seguramente se trate de una dirección de APT1.

Saber los puertos abiertos de un ordenador atacante sabiendo también su dirección IP puede ser útil para realizar un ataque a las aplicaciones del ordenador enemigo.

Tenemos que tener en cuenta también el número de servicio asociado (service probe), si recibimos un ataque con una determinada dirección IP y tiene un número de servicio, por ejemplo, de 4, sabemos qué tipo de ataque realiza por lo que podemos defendernos mejor.

Saber el ASN de una determinada dirección IP permite filtrar información y averiguar de donde viene el ataque y saber las redes que están bajo una misma gestión.

En cuanto al estudio de WEBC2-DIV:

El núcleo fundamental de este malware se basa en desencriptar información y actuar según se desencripte la información en el equipo víctima. Una posible manera de combatir este malware puede ser evitar que se produzca dicha desencriptación mediante algún software especializado.

Otra manera puede ser ver el código fuente de la página enviada y ver si hay caracteres extraños en ella, en caso de que sí, no ejecutarla bajo ningún concepto o ejecutarla en un laboratorio virtual.

1.3 Objetivos del TFM

Por todo lo mostrado en secciones anteriores, los principales objetivos planteados en este trabajo fin de máster se pueden resumir en los siguientes puntos:

1. Analizar unas determinadas direcciones IP relacionadas con APT1. Si analizamos el “como” conseguimos esas direcciones, serán extraídas de varios documentos, dos denominados JIB (Joint Indicator Bulletins, contenidos en [21] y [22]) y otro mediante un documento apéndice del informe Mandiant, dicho informe es el más extenso en cuanto a la descripción del ataque APT1, pero, apenas tiene contenido “práctico”. Atendiendo al “para que”**,** se obtiene mucha información de las direcciones IP (marcas, puertos, números de servicio…), luego se obtienen otras direcciones IP para su análisis según su ASN. Todo esto puede ser útil para entender las características de dichas direcciones IP, lo que nos da ventaja a la hora de actuar en caso de ataque.
2. Estudio del malware WEBC2-DIV. En cuanto al **“**como**”** extraemos ese malware, este fue obtenido de una página web que almacenaba un montón de malware en formato comprimido, para descomprimirlo tuve que pedir la contraseña al “dueño” de la página. En cuanto al **“**para que**”**, dicho estudio permite esclarecer cómo funciona el malware desde un punto de vista detallado mediante programas como: Process Hacker, InetSim, Wireshark, lenguaje ensamblador mediante IDA y Cutter...
3. Descripción superficial del malware Poison Ivy. Para ello definiremos lo que es una RAT y el manejo de dicho malware.

1.4 Planificación inicial

1. Almacenar las direcciones IP. He tardado 4 días. Su complejidad ha sido fácil.

2. Examinar la distinta información que se obtiene, usando los programas de Internet Census, he tenido que descargarles y me ha llevado bastante tiempo tanto encontrarlo en la página web como en su descarga, además de haber tenido que descargar solo aquella información específica. En el diagrama de Gantt describo lo "tratado" en Internet Census (fingerprint, synscan…). He tardado 1 semana. Su complejidad ha sido difícil.
3. Estudio del malware WEBC2-DIV mediante Wireshark, IDA y Cutter. He tardado 8 días. Su complejidad ha sido difícil.
4. Descripción superficial del malware Posion Ivy. He tardado 1 día. Su complejidad ha sido fácil.

1.4.1 Planificación temporal (Diagrama de Gantt)

En la figura 1 se puede observar el Diagrama de Gantt. Este se ajusta a la descripción definida en el paso anterior.

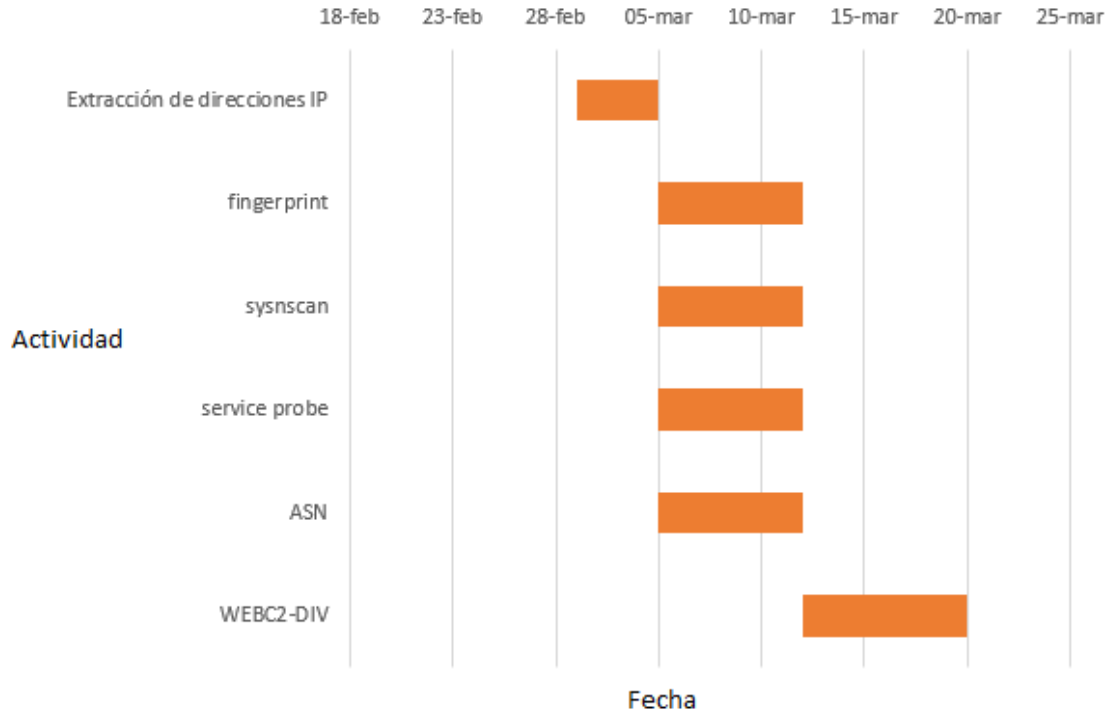


Figura 1: Diagrama de Gantt

1.4.2 Presupuesto y estimación de costes

En la tabla 1 se reflejan las diferentes fases del proyecto con el número de horas por fase y el coste de cada una, más el coste total.

Tabla 1: Presupuesto y estimación de costes

Actividad	Horas	Coste

Almacenar las direcciones IP	96	1440 euros
Examinar la información	168	2520 euros
Estudio de WEBC2-DIV	192	2880 euros
Descripción de Poison Ivy	24	360 euros
	15 euros/hora	7200 euros totales

1.5 Organización del trabajo realizado

El siguiente trabajo de fin de máster se organiza de la siguiente manera. En los apartados anteriores se puede encontrar una introducción que se centra en un enfoque más teórico relacionado con los APT y mostrar la información más básica del propio proyecto (objetivos y planificación).

Tras esta sección se muestra una sección dónde se detalla la APT seleccionada para su estudio, en este caso la APT1. Posteriormente se pasará a describir una metodología de forma más detallada de los pasos que se van a seguir en el proyecto, dividido en diferentes fases. A continuación, se va a describir las herramientas que conforman el laboratorio forense, así como las distintas actividades que han tenido lugar dentro del análisis forense propiamente realizado. Por último, se abordarán los resultados alcanzados a través de todo el proyecto. Y finalmente el apartado conclusiones y posibles trabajos futuros relacionados con este trabajo fin de máster.

Dicha organización ha sido establecida atendiendo a un orden lógico. Empezando por la teoría para dar un toque introductorio y general de lo que es el proyecto, para que el lector se empiece a familiarizar con él. Después se atiende a aspectos más concretos y “técnicos” que abarcan que es lo que se espera en cuanto a la realización del proyecto (objetivos), como dichos objetivos van a ser conseguidos de forma general y ordenada (planificación, metodología) y que herramientas nos harán falta, ya de cara al análisis práctico. Una vez dados los datos “introductorios” se procederá al análisis práctico. Para acabar, será necesario hacer un balance final mediante la introducción de resultados obtenidos y conclusiones, estos aspectos solo pueden ir al final, cuando ya se hayan realizado los pasos relacionados con la parte práctica.

2 Una aproximación teórica a APT 1

En cuanto al descubrimiento, impacto y relevancia, APT1 ha usurpado cientos de datos de al menos 141 organizaciones en un grupo diversificado de empresas, su comienzo empezó en 2006. APT1 señala a muchas organizaciones a la vez. Una vez realizado el acceso a los datos de la víctima, su acceso es continuo con acceso periódico durante mucho tiempo con el objetivo de usurpar datos de carácter industrial como planos, métodos de fabricación, pruebas, información sobre negocios, precios, asociaciones, emails y contactos. Todo lo observado representa una pequeña porción de ciberespionaje existente.

En 2006, APT1 ha expandido sin parar la “abertura” a nuevas personas potenciales a ser atacadas. En la figura 2 se puede ver como tenemos 141 compromisos conocidos; cada marca en la figura muestra una víctima con separación y señala la fecha inicial de actividad APT1 en la red organizativa. Apenas hay evidencias electrónicas. Se subestima el tiempo de cuanto está la APT1 en la red, en la figura 2 se observa un panorama general de los ataques por fecha. Toda esta información se muestra en [13].

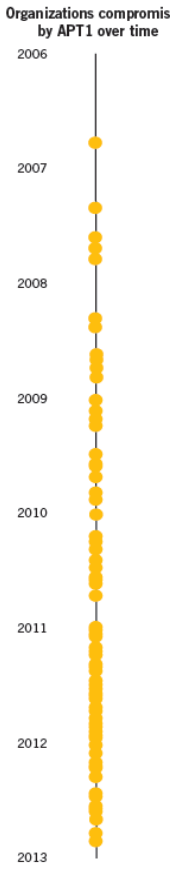


Figura 2: Figura de ataques en el tiempo [13]

Cuando APT1 ha atacado una red, se monitoriza y usurpan de forma continuada información de propiedad y comunicaciones del perjudicado durante mucho tiempo. APT1 estuvo con acceso a la red del perjudicado de media un año. El periodo más largo

en que APT1 accedió a la víctima fue de unos 1764 días. Su actividad no era continua, como nos informa el documento contenido en [13].

También nos informa de que, si tenemos en cuenta el enfoque geográfico e industrial de APT1, las empresas que APT1 tiene como objetivo utilizan el inglés. Hay como objetivo un numero pequeño de víctimas que no tienen el inglés como lengua. Alrededor de un 87% de los perjudicados por el APT1 son sitios con el inglés como idioma nativo. 115 víctimas en EE. UU, 7 en Canadá y Reino Unido. Del resto de los 19 perjudicados restantes, 17 tienen el inglés como idioma principal para su actividad organizativa. Agencias relacionadas con el desarrollo y cooperación de gobiernos extranjeros en el que el inglés es uno de sus idiomas principales es incluido. Grandes organizaciones también usan el inglés. Solo dos perjudicados utilizan un idioma distinto al inglés. En el PLA Unit 61398 se usa el idioma inglés. En la figura 3 se puede observar los países afectados por APT1.

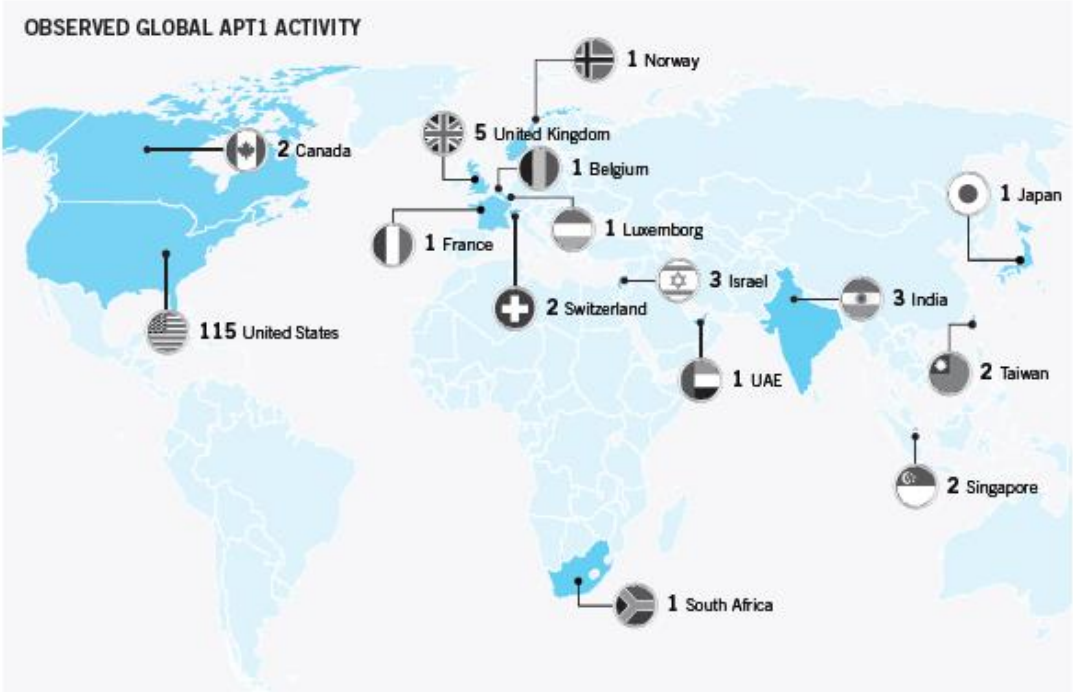


Figura 3: Observación global de la actividad APT 1 [13]

Como continuación del apartado anterior, se informa de que APT1 ha demostrado la capacidad y la intención de robar a docenas de organizaciones Se ve que APT1 tiene el poder y quiere substraer a muchas organizaciones en una gran variedad de industrias virtualmente a la vez. La figura de arriba provee la visión del tiempo más antiguo que se conoce del "uso" del APT1 en contraste a las 141 víctimas, que se organizan en 20 industrias que los representan. Las conclusiones dicen que el objetivo de APT1 es extensa; el conjunto no señala a las industrias, pero las ataca de forma seguida.
Las figuras de la página anterior solo suponen una pequeña parte. Las industrias que señalan APT1 puede ser más extensa. La visión de los ejercicios que son paralelos sugiere que el grupo cuenta que una gran cantidad de grupos humanos y de perfil técnico. La figura señala que APT1 muestra un compromiso con 17 desfavorecidos usuarios que trabajan en 10 industrias. Como hemos observado que el conjunto se mantiene constante en el tiempo de cada víctima en la red durante más o menos un año

después del tiempo inicial, hacemos énfasis que APT1 cometió esas 17 acciones perjudiciales y sigue extrayendo datos al mismo momento que conserva el acceso y continúa extrayendo datos de una serie de desfavorecidos. En la figura 4 se puede ver la evolución de los ataques en diversos sectores.

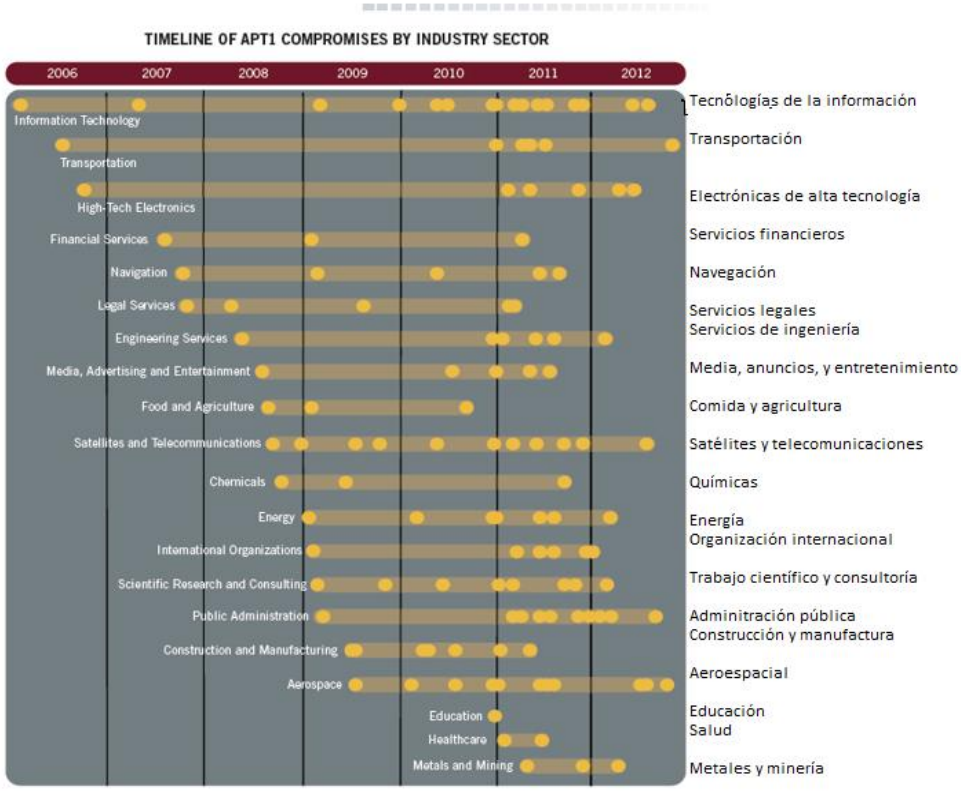


Figura 4 :Línea de tiempo que compromete el sector de la industria [13]

Según [13], se cree que los grupos en las industrias relacionadas con negocios en China son potenciales víctimas. Se puede ver que APT1 ha señalado a 4 de las 7 industrias al "alza", que China señaló en su doceavo plan quincenal. En la figura 5 se puede observar el número de víctimas del APT1 separadas por sectores "básicos".

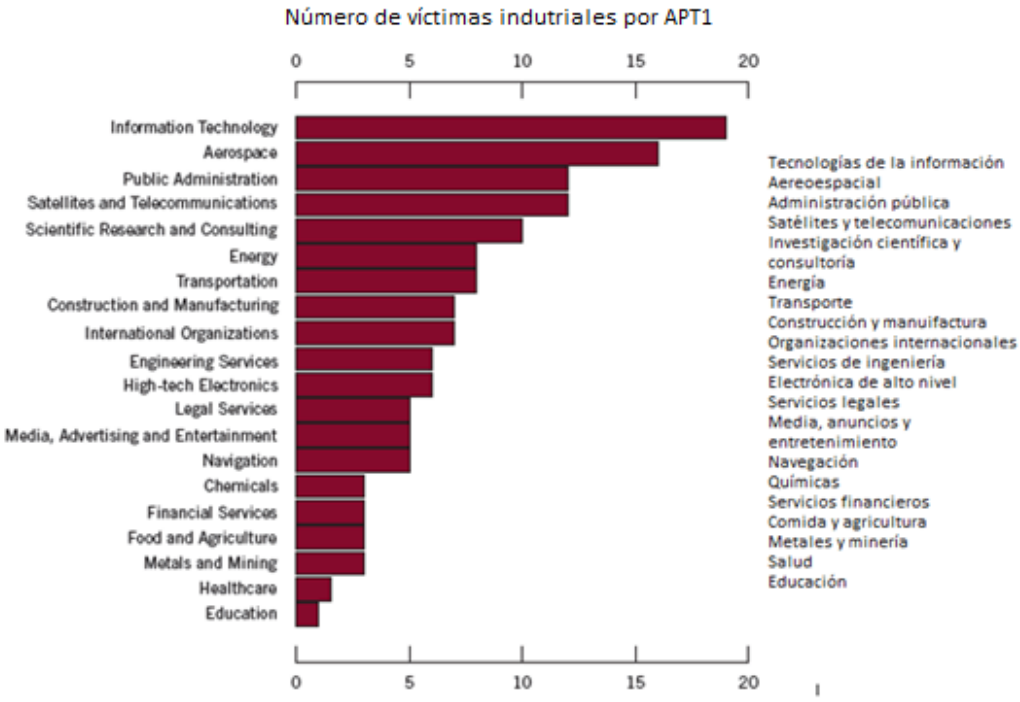


Figura 5: Sectores de ataque [13]

También, según [13], haciendo hincapié en el robo de datos, APT1 extrae mucha información de sus víctimas. El grupo de conocimientos robados es el siguiente:
Uso y desarrollo del material, en la que se incluyen resultados, diseños, manuales, listas, simulaciones, procedimientos, descripciones, normas, procesos, planes, negociaciones, precios, eventos, posiciones, análisis, correos electrónicos, credenciales e información de red.
Es costoso saber cuánto ha robado APT1 debido a: la eliminación de archivos que están comprimidos, dejando evidencia normalmente comprimida durante actividades de comercio. Monitoreo ineficaz. Duración entre robo e investigación durante el curso del negocio. Ciertas víctimas prefieren invertir recursos para enfocarse en la seguridad de la red, en vez de investigar el problema de seguridad.
Se observa que APT1 ha extraído unos 6,5 terabytes de datos que están comprimidos de un solo grupo durante un tiempo de 10 meses. Según estás evidencias podemos decir que APT1 ha robado muchísimos terabytes de sus víctimas.
No tenemos información de quién recibe los datos que APT1 roba o como es procesada la información extraída, se cree que esa información se usa como ventaja para la República Popular China y empresas del estado. En 2008 APT1 puso en compromiso la red de una empresa relacionada con la industria de tipo mayorista. APT1 usa herramientas que se instalan para la creación de archivos que están comprimidos y la extracción de correos electrónicos que archivos que están adjuntos. En los siguientes 2 años y medio APT1 extrajo un gran número de información de archivos y tuvo acceso repetidamente a las cuentas de correo de algunos ejecutivos incluido la gente con una profesión importante.
Sería una sorpresa que APT1 continuase realizando un esfuerzo de espionaje y extracción de datos si el esfuerzo de los resultados del grupo no va a parar a sistemas capaces de convertirlo en capital, toda esta información también está en [13].

En cuanto a la difusión de APT1, hay informes que establecen y forman ciertas observaciones sobre el recorrido de espionaje de APT1. Hay muchas causas que complican el acceso de información de APT1, por ejemplo, como está la seguridad de la red.
Hay muchos nombres referidos a APT1 según gente especializada. Muchos expertos en ciberseguridad se enfocan en la escritura sobre herramientas que son compartidas entre muchos conjuntos de tipo APT chino sin establecer una diferencia entre los usuarios que lo utilizan. Para que sirva de ayuda a los que investigan a la identificación de tipo APT1, la tabla nos ofrece un lisado de apodos de conjuntos de APT aparecidos frecuentemente en los medios y se diferencia entre los que deben hacer una descripción de APT1 y los que no. Existe una lista de caracteres públicos sobre amenazas de tipo chino que se confirma que aparece en APT1.
El informe primero de carácter público es la publicación en 2006 de una división de tipo japonesa conocida por Symantec. En el informe se menciona el nombre de un host llamado sb.hugesoft.org cuyo registro se asocia a una persona que se hace llamar UglyGorilla.
En el año 2012, en el mes de septiembre, Brian Krebs, propietario de un blog llamado "Krebs on security" comunicó sobre un problema de ciberseguridad en Telvent Canada ltd que se atribuye a APT1 con relación de las herramientas de la infraestructura que los ciberdelincuentes usaron para la explotación y obtención de acceso al sistema. Enlazando con el apartado anterior, la información aquí descrita también está en [13].

Se puede observar en la siguiente tabla (Fig. 6) ciertos apodos relacionados con el APT1.

Nickname	Verdict
"Comment Crew"	APT1 confirmado
"Comment Group"	APT1 confirmado
"Shady Rat"	Posible APT1, no confirmado
"Nitro Attacks"	No APT1, atribuido a otro APT
"Elderwood"	No APT1, atribuido a otro APT
"Sikipot"	No APT1, atribuido a otro APT
"Aurora"	No APT1, atribuido a otro APT
"Night Dragon"	No APT1, atribuido a otro APT

Figura 6: Listas de apodos de grupos APT aparecidos en los medios [13]

Una de las características de APT1 es que tiene una metodología de ataque bien definida, que ha sido perfeccionada a lo largo de los años y ha sido diseñada para el robo de gran cantidad de propiedad intelectual. Empiezan con un phishing de tipo electivo y agresivo, se procede a el despliegue de armas digitales con personalización y acaban con la exportación de paquetes de archivos que están comprimidos a China, antes de que el ciclo vuelva a empezar nuevamente. El idioma que se maneja es el inglés, usado en los correos electrónicos de ingeniería social. Se ha evolucionado las armas digitales durante más de siete años, a lo que deriva en continuas actualizaciones que forman parte de su propio ciclo de "lanzamiento" de software. Tiene capacidad de adaptarse a su entorno, esto significa que los ataques se hacen efectivos en entornos

empresariales con “relaciones de confianza”. Estos ataques forman parte de un patrón cíclico de actividad, como se puede ver en la figura 7, contenida en [13].

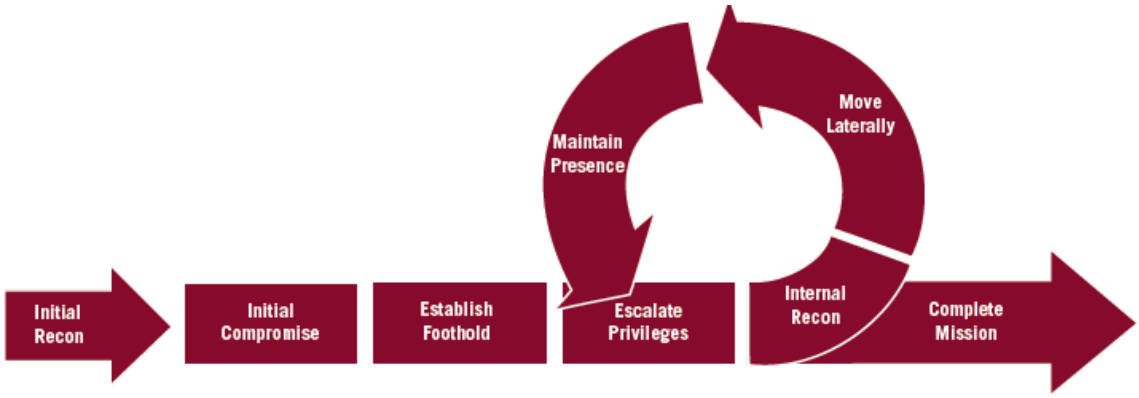


Figura 7: Ciclo de vida del ataque [13]

En la figura 8 se puede observar la familia del malware de APT1.

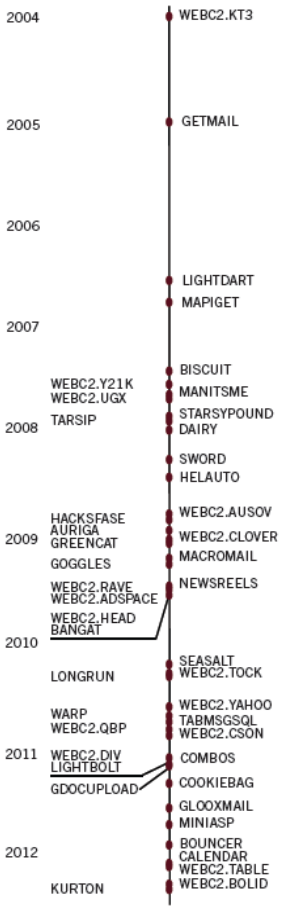


Figura 8: Familia

3 Metodología

3.1 Fases del trabajo

De todas las APTs existentes, se ha elegido la APT1. Se ha seleccionado esta porque debido a que está bastante estudiada y detallada en diferentes artículos ubicados en internet.

En el siguiente apartado se explica la metodología de trabajo, es decir, el método que he usado yo para realizar el desarrollo del proyecto. Ha sido dividido en tres fases atendiendo a su función/es concreta/s.

Fase 1: se ha recolectado datos (IPs), extraídos en los JIBs INC260425, INC260425-2 e IPs de Mandiant. Siempre hay que tener una copia de las muestras para que no perdamos su contenido original al ser modificadas. Esta fase es importante y debe ser tratada con un determinado orden, sabiendo diferenciar entre las direcciones IP de los JIBs y los de Mandiant, para que no haya una confusión entre las diferentes direcciones IP. Esta fase es la primera porque se extrae “en bruto” los “materiales” (direcciones IP) a utilizar en el resto del trabajo (la “primera parte”).

Fase 2: se ha estudiado el “funcionamiento” de Internet Census, apoyándonos en sus respectivas funciones, agrupadas en “colecciones” de datos de diversa índole. Las usadas en el proyecto son: fingerprint, synscan y service probe. También se ha realizado un análisis de las direcciones IP según su ASN. Se ha intentado realizar un análisis basado en Routing Data, Country Code y Open Resolver. El orden elegido de la realización de los análisis derivados de Internet Census es debido a su orden (valga la redundancia) en dicha página. Esto se hace en primer lugar para concatenar directamente la obtención de “materiales/herramientas” con el apartado “práctico”, descrito en esta fase. Después se realiza la parte ASN, como parte independiente del trabajo. Esta fase es la segunda debido a que es la práctica justo después de la obtención de los “materiales/herramientas” y, quedaría de forma errónea y desordenada, empezar otra fase que no esté ligada íntimamente con la anterior.

Fase 3: se ha conseguido una muestra del malware WEBC2-DIV y se ha estudiado usando InetSim, Wireshark, IDA y Cutter. La muestra no será alterada de ninguna manera, al tratarse de una muestra concreta de malware. La arquitectura es mostrada en pasos posteriores. El orden de esta fase ha sido aleatorio, primero se ha instalado InetSim debido a que es necesario para para trabajar con Wireshark; se podría haber empezado por el análisis de IDA o Cutter. Sin embargo, el análisis con esas herramientas (IDA y Cutter) es más exhaustivo que el análisis con Wireshark, por lo que se ha optado por un orden de (relativamente) fácil a más complejo. A pesar de todo, la información ubicada en Cutter es más fácil de interpretar (en este trabajo) que la mostrada en IDA. Esta fase es la tercera por un criterio personal, pudiéndose haber hecho la primera de todas. También, al final, describo un apartado relacionado con un malware llamado “Poison Ivy”.

3.2 Laboratorio forense creado para realizar el análisis

El laboratorio forense se compone de los siguientes entornos y herramientas:

Entornos: Máquinas virtuales usadas en nuestro análisis, en concreto estas son Ubuntu 20.04.1 y Windows server 2016, dichas herramientas se “comunicarán entre ellas” en uno de los pasos siguientes.

Herramientas que se utilizan en esos entornos:

3.2.1 SilK

SilK (System for internet knowledge) es un grupo de herramientas para analizar tráfico desarrolladas por el CERT con el objetivo de facilitar el acto de analizar la seguridad de grandes redes. El grupo de herramientas SilK permite el recopilatorio, el almacenado y el análisis de forma eficiente de los datos relacionados con el flujo de datos que van por la red, lo que facilitan los que analizan la seguridad de la red la consulta rápida de gran grupo de tráfico de tipo histórico. SilK es bueno para el análisis de tráfico en una red de tipo frontal o en la zona borde de, por ejemplo, granes empresas o ISP de tamaño menor. Se utiliza esta herramienta para la conversión de, por ejemplo, archivos *.list* en archivos *.set*. Se utiliza como cualquier comando de Ubuntu. Aunque realmente, atendiendo al uso en el desarrollo y análisis, dicha herramienta, en este trabajo, se usa para conseguir archivos *.set* en función de los *.list*, para realizar una intersección y para generar un pmap. En los diferentes apartados donde se usa es, por ejemplo, en los apartados service probe o ASN. Su uso es bastante breve, una explicación completa de SilK se puede encontrar en [3]. No se conoce alternativas a esta herramienta. La razón de usar esta herramienta es porque se necesita para crear archivos nuevos a partir de otros para trabajar con ellos.

3.2.2 Comandos de Linux

Los comandos de Linux son un conjunto de palabras de tipo reservado que usa el sistema operativo para la ejecución de determinados actos usando una línea de comandos o un terminal. Un terminal Linux es una “pantalla” o un programa en la que es posible ejecutar los comandos. Un comando es una “orden” cuyo sistema operativo es indicada para ejecutar una tarea. Los comandos existentes permiten modificar, crear o mover archivos y carpetas, su uso viene indicado en [4]. Teniendo en cuenta su uso en el desarrollo y análisis, esos comandos sirven para manipular los diferentes datos que poseemos, por ejemplo, filtrar, cortar, comandos SilK…Una alternativa posible serían comandos Windows, pero, no tienen tanto alcance como los comandos Linux. Otras alternativas más útiles pueden ser usar los sistemas operativos Kali Linux u OS Parrot para introducir en esos sistemas operativos los comandos apropiados. La razón de utilización de esta herramienta es porque se usa para manipular (filtrar, cortar…) archivos.

3.2.3 Internet Census 2012:

Mientras se trabajaba con nmap scrpit engine, fue mencionado que se debería probar el inicio de sesión clásico telnet root:root en direcciones IP de tipo aleatorio. Se comenzó el escaneo y nos damos cuenta de que había muchos dispositivos sin protección en internet.

Después de la finalización del escaneo de unas cien mil direcciones IP, la cantidad de dispositivos sin seguridad deben ser unos 10000. Se comienza con un dispositivo y se asume una velocidad de exploración de 10 direcciones IP por segundo. Si implementamos un sistema de escaneo en el dispositivo recién encontrado, la velocidad

de escaneo se duplicaría. De diez sondeos por segundo que son escaneados por diez mil dispositivos, tendríamos un escáner de puertos con distribución para escanear puertos de todas las direcciones IPv4 en el intervalo de tiempo de 1 hora.

Para la minimización de la interferencia con el funcionamiento del tipo normal del sistema, el binario fue configurado para su ejecución como un perro guardián y con la prioridad más pequeña que puede tener el sistema. Su instalación no fue permanente y se paró pasados unos días. También fue implementado un archivo Readme, que tiene una descripción del trabajo y un correo electrónico del contacto.

El binario tiene dos partes. El primero es un binario de tipo telnet que realiza combinaciones de iniciar sesión de forma diferente, por ejemplo root:root y admin:admin y los dos sin contraseña, mediante las conexiones necesarias. El escáner es administrado por la segunda parte, le ofrece rangos de IP para la carga y el escaneo a una dirección IP de tipo específico. Nuestro binario es implementado en las direcciones IP recopiladas de nuestros datos de ejemplo y se comienza el escaneo en el puerto 23 (telnet) en las respectivas IPv4. El escáner telnet empezó en los dispositivos encontrados, una noche duró el escaneo completo. Se detuvo la implementación después de que el binario empezó en unos 30000 dispositivos.

No había interés en la interferencia con el funcionamiento “de serie”, no se cambió la contraseña ni se realizaron cambios permanentes. Una vez se ha reiniciado, el dispositivo volvió a su estado inicial, incluido una contraseña defectuosa, sin ninguno de nuestros datos actualizados. Los binarios eran ejecutados en una prioridad muy baja y tienen un perro guardián para detener el ejecutable en caso de fallo. El escáner tenía 128 conexiones de tipo simultáneo y un tiempo de espera de tipo de conexión de 12 segundos. Esto produce la limitación de escaneo a 10 IP por segundo por cliente. También fue cargado un archivo Léame, cuyo contenido es una pequeña explicación del proyecto, también un email de contacto para dar comentarios a los que investigan la seguridad, las fuerzas del orden y el ISP.

Muchos dispositivos sin protección son decodificadores o enrutadores que pueden ser encontrados en muchísimos dispositivos. Un conjunto está formado por máquinas cuya CPU y RAM es igual. Existen pequeños grupos de máquinas cuya disponibilidad es pequeña (unos cientos de veces). Vemos en detalle esos dispositivos para saber cuál podría ser ese propósito y, de forma rápida, y encontramos enrutadores IPSec, enrutadores BGP, equipos x86, sistemas de control, sistemas de seguridad y grandes equipos. No nos enfocamos del tráfico que pasa por los dispositivos y todo lo que se encuentra detrás de los enrutadores. Debido a esto que no existen estadísticas de tipo arc, dhcp, conteo o monitoreo de tráfico, escaneo de puertos de dispositivos LAN.

Se usan estos dispositivos como un “utensilio” para trabajar en internet, atendiendo a la privacidad y teniendo un factor “invasivo” lo menos posible. Como se observa en los datos de muestra, los dispositivos no seguros se ubican en todas partes de Internet. No tienen especificación de país o ISP. Debido a esto el problema de predeterminadas contraseñas o vacías están relacionadas con Internet.

Se usa un conjunto de reglas para la identificación de la CPU y RAM de dispositivos finales para la garantía de que el binario solo fuera implementado en equipos donde se sabía que funcionaba. También se excluyen todos los dispositivos de grupos pequeños, porque no queríamos tener interferencias con hardware crítico o controles industriales.

El binario fue ejecutado en 420000 dispositivos, eso supone que solo alrededor del 25% de todos los dispositivos encontrados desprotegidos. Hay muchos dispositivos que carecen de Shell real, por lo que no se puede cargar un binario. Se puede usar la herramienta ifconfig para obtener la información MAC de gran parte de los dispositivos. Se recopilan las direcciones MAC durante cierto tiempo y se identifica alrededor de 1,2 millones de dispositivos desprotegidos. Esto no incluye los dispositivos que no tienen ifconfig. Ha habido que descargar aquella información específica para cada análisis, eso ha sido costoso. Toda esta información procede de [5].
En cuanto al desarrollo y análisis, se extrae la información necesaria de la página web oficial y se almacena en carpetas lo más ordenadas posibles.
La razón de uso de esta herramienta es porque maneja una gran cantidad de datos relacionados con sistemas operativos, puertos abiertos o cerrados…Todos asociados a diferentes direcciones IP. No ha sido encontrada ninguna "herramienta" alternativa. Atendiendo al motivo de usar esta herramienta, se usa porque necesitamos muchos datos para manipularlos con comandos Linux.

3.2.4 IDA:

IDA puede, como desensamblador, mostrar las instrucciones en lenguaje ensamblador, es decir, su representación simbólica. Se han usado técnicas de tipo avanzado para que el código sea legible a los usuarios. La depuración característica favoreció el análisis dinámico. Provee múltiples objetivos de tipo depuración y para manejar aplicaciones remotas. IDA permite la depuración instantánea, fácil conexión a procesos remotos y locales, compatibilidad con sistemas de 64 bits y nuevas posibilidades de conexión, todo ello gracias a su depuración multiplataforma.
Si examinamos el uso en el desarrollo y análisis, el uso de IDA es debido a que ofrece una interfaz gráfica muy potente que sirve para examinar el malware. Si tenemos en cuenta la razón de usar dicha herramienta es porque ofrece una interfaz del código ensamblador, análisis hexadecimal…para el examen del malware. Todo esto se define en [6], su página principal. Unas alternativas pueden ser x64dbg o TitanMist.

3.2.5 Process Hacker:

El administrador de tareas de Windows es una herramienta muy útil. Debido a ello podemos saber todos los programas en memoria y los procesos ejecutados en tiempo real. Sin embargo, puede querer saber más conocimiento sobre dichos procesos. El administrador de tareas puede quedarse corto. Process Hacker nos permite conocer en tiempo real como es el estado de nuestro ordenador.

En resumen: es una herramienta de Windows, gratis, y "abierta", con el objetivo de reemplazar al administrador de tareas de Windows para aquellas personas que necesitan más control sobre los procesos del PC. Cómo característica se puede resaltar la posibilidad de mostrar gráficas sobre el tiempo real del estado del hardware.

Teniendo en cuenta el uso en el desarrollo y análisis, se examinará una interfaz compuesta por muchos procesos, estando en pantalla el malware en ejecución. El motivo del uso de esta herramienta es que es útil para ver el malware en ejecución. Unas alternativas pueden ser Process Explorer o System Explore. Para ver en profundidad la utilidad de la herramienta mirar [7].

3.2.6 Cutter:

Es una herramienta cuyo objetivo es simular algunos protocolos de comunicación como UDP o TCP, recibe paquetes y produce respuestas falsas, pero con una forma acorde con lo que espera recibir del malware.

Rizin es el motor principal de Cutter. Se puede acceder a muchas funciones a través de una GUI o terminal integrado.

Para que la experiencia de ingeniería inversa sea lo mejor posible, Cutter proporciona una entorme cantidad de funciones y widgets diferentes.

De forma predeterminada, Cutter viene con varios temas modernos: claro, nativo, oscuro y medianoche. Las versiones de Cutter están totalmente integradas con el descompilador nativo de Ghidra. No hay Java involucrado.Todo este informe está detallado en [8].

Atendiendo al uso en el desarrollo y análisis, el uso de Cutter permite examinar el malware en detalle, ya sea su versión hexadecimal o su estudio mediante lenguaje ensamblador. Si tenemos en cuenta la razón de uso de esta herramienta es porque permite el análisis del malware de una manera bastante explícita. Una alternativa posible podía ser el propio IDA.

3.2.7 Wireshark:

Wireshark es una herramienta muy utilizada para cualquier administrador de sistemas o profesional de seguridad. Es gratuito y permite, en tiempo real, analizar tráfico. Es una herramienta muy útil para detectar hackers o saber la latencia de la red. Requiere conocimientos no muy avanzados de redes, eso incluye: comprender la pila de protocolos TCP/IP, saber interpretar los encabezados de los paquetes y cómo funcionan en el enrutamiento, DHCP o reenvío de puertos, etc.

La herramienta permite observar de forma legible el tráfico interceptado, esto permite identificar el tráfico que pasa por la red. Wireshark admite más de 2000 protocolos de red, como nos indica [9].
En cuanto al uso en el desarrollo y análisis, con Wireshark se analizará información importante relacionada con el malware. Atendiendo a la razón de uso de esta herramienta es porque ofrece una interfaz intuitiva mediante la cual se puede examinar el malware.

3.2.8 InetSim:

Cuando es realizado un análisis dinámico de una posible amenaza, podemos encontrarnos con la situación de que se quiere establecer una comunicación con algún equipo de fuera mediante internet. Si monitorizamos la comunicación entre computadora-servidores obtenemos mucha información de valor al accionar el malware. Existen casos en que es preferible evitar la comunicación real con servidores maliciosos. Podría pasar que los servidores no estén disponibles, esta situación puede producir que el malware pueda cambiar su comportamiento y no seríamos capaces de entender las acciones que realiza el mismo o los objetivos que persigue. Podría suceder que necesitemos ejecutar el malware muchas veces antes de poder entenderlo. Si se

comunica con los servidores maliciosos podemos alertar a los criminales de nuestra presencia.

En ocasiones, es una alternativa favorable simulas ciertas situaciones. Por ejemplo, podemos hacer creer al malware que se está conectando con los servidores maliciosos, cuando, en verdad, está enviando mensajes a nuestra máquina. Eso es lo que ocurre con InetSim.

Comprendiendo el uso en el desarrollo y análisis, utilizar esta herramienta permite crear un “laboratorio” en el cual “atacamos” a una máquina, pero se ataca realmente al “sustituto”, siendo este InsetSim, mostrando su uso en [10]. La razón de uso de esta herramienta es porque es necesario crear un “servidor falso” para atacarle y examinar dicho ataque.

3.2.9 Posion Ivy

Posion Ivy no es una herramienta en sí misma, pero la hemos definido como un software maligno relacionado con APT1. Una explicación sobre el funcionamiento de este troyano es que Poison Ivy se encuentra dentro de la categoría RAT (Remote Access Trojan). Este utiliza un método de conexión llamado “conexión inversa”, es decir, cuando el troyano infecta a un usuario y el pirata informático no necesita conocer, sino que dicho troyano infecta a un usuario y el hacker no necesita saber la IP de la víctima, este virus ya está configurado con la IP del pirata y será este el que tenga conexión en busca del servidor con el que controlarlo. Posion Ivy es muy buena en términos de control remoto. Permite tomar capturas de pantalla, robar las pulsaciones de teclado, acceso a los archivos de la víctima. Cualquier ataque se puede llevar a cabo con Poison Ivy. Se pueden ver sus características en [11].
Atendiendo el uso en el desarrollo y análisis, este malware solo será descrito de forma superficial, sin entrar en detalles en su funcionamiento. El motivo de explicar este malware es para dar otro ejemplo de malware relacionado con APT1, sin entrar en detalle de funcionamiento.

3.1.10 Tabla Comparativa

En la siguiente tabla se puede observar las ventajas y desventajas de las herramientas descritas anteriormente.

Tabla 2: Tabla comparativa

Herramienta	Ventajas	Desventajas
SilK	Instrucciones fáciles de entender	Poco uso práctico
Comandos de Linux	Muchas funciones para manejar información	Hay que tener conocimientos mínimos de comandos usados en terminal
Internet Census	Un gran montón de información relacionada con sistemas operativos,	Demasiada información, lo que es difícil de discriminar

	puertos abiertos…asociados a direcciones IP	u observar de manera clara dicha información
IDA	Potente interfaz gráfica, bien ordenada y detallada	Complejo manejo por los menús, la información puede ser difícil de interpretar
Process Hacker	Buen sustituto del editor de registro	Compleja interpretación de la información para gente sin conocimientos informáticos
Cutter	Potente interfaz gráfica	Complejo manejo por los menús, la información puede ser difícil de interpretar
Wireshark	Interfaz gráfica con mucha información ordenada de forma cómoda	Complejo manejo por los menús, la información puede ser difícil de interpretar para alguien sin conocimiento de redes.
InetSim	Ofrece un excelente servicio de servidor de pruebas para ser infectado	Problemas técnicos de difícil solución en cuanto a su configuración
Posion Ivy	Funciones muy interesantes relacionadas con el uso del malware	Requiere conocimiento avanzado de malware

4 Análisis

4.1 Recolección de datos IP

Este estudio utiliza direcciones IP asociados con APT1, recopilado de los Joint Indicator Bulletins (JIB INC260425 y INC260425-2). El acceso a dichos documentos está en los documentos [21] y [22]. También se trabajará con nombres de dominio proporcionados por Mandiant, ubicados en [18]. En cuanto al juego de datos y su estructura se explica posteriormente.

4.2 JIBs INC260425 y INC260425-2

En el apartado anterior se define que recolectaremos los datos de “dos” fuentes distintas, dichos datos son JIBs. En este apartado se estudian.

Dichos documentos (JIB INC260425 y INC260425-2) se estructuran de la siguiente manera:

-El título: Nombre e identificación.
-La fecha.
-Notificación, Introduction y document overview: Habla en líneas generales de cómo es el documento.
-Indicator description: Definición de una serie de términos relacionados con las redes.
-Una serie de datos de contacto.
-Un documento FAQ.
-Un technical Data: Aquí se ubican las direcciones IP a estudiar.

Los JIB, cuyas fechas son el 18 de febrero de 2013 y 26 de febrero de 2013, tenían 855 direcciones IP asociadas con la actividad de carácter malicioso APT1. Para que nuestro análisis sea más preciso, se combina la propiedad intelectual de las direcciones encontradas en los dos JIB en un archivo llamado *Ij.list*, guiándonos por el guion ubicado en [16].

La figura 9 hace referencia a los documentos JIB INC260425 - INC260425-2, en concreto se puede visualizar el título, la fecha y la Notification.

La figura 10 muestra el apartado donde están las direcciones IP.

Joint Indicator Bulletin (JIB) – INC260425

February 18, 2013

Notification

This report is provided "as is" for informational purposes only. The Department of Homeland Security (DHS) does not provide any warranties of any kind regarding any information contained within. In no event shall the United States Government or its contractors or subcontractors be liable for any damages, including but not limited to, direct, indirect, special or consequential damages, arising out of, resulting from, or in any way connected with this report, whether or not based upon warranty, contract, tort, or otherwise, whether or not arising out of negligence, and whether or not injury was sustained from, or arose out of the results of, or reliance upon the report.

Figura 9: Ejemplo JIB1 [21]

IP Address Awareness List

****

107[.]6[.]38[.]55
108[.]171[.]207[.]62
108[.]171[.]244[.]138
108[.]171[.]246[.]87
108[.]171[.]248[.]182
108[.]171[.]248[.]83
108[.]171[.]248[.]86
108[.]171[.]252[.]41
108[.]171[.]254[.]76
112[.]121[.]164[.]2
112[.]133[.]203[.]215
112[.]133[.]203[.]250
115[.]119[.]92[.]178
115[.]178[.]60[.]19
116[.]212[.]100[.]94
117[.]121[.]241[.]186
119[.]75[.]5[.]132
119[.]75[.]5[.]134

Figura 10: IPs del JIB1 [21]

La figura 11 muestra el documento del segundo JIB, tiene la misma estructura que el anterior, exceptuando la fecha. La figura 12 muestra las direcciones IP asociadas al segundo JIB.

Joint Indicator Bulletin (JIB) – INC260425-2

February 26, 2013

Notification

This report is provided "as is" for informational purposes only. The Department of Homeland Security (DHS) does not provide any warranties of any kind regarding any information contained within. In no event shall the United States Government or its contractors or subcontractors be liable for any damages, including but not limited to, direct, indirect, special or consequential damages, arising out of, resulting from, or in any way connected with this report, whether or not based upon warranty, contract, tort, or otherwise, whether or not arising out of negligence, and whether or not injury was sustained from, or arose out of the results of, or reliance upon the report.

Figura 11: Ejemplos de JIB2 [22]

****
IP Address Awareness List
****
107[.]6[.]38[.]55
108[.]171[.]207[.]62
108[.]171[.]244[.]138
108[.]171[.]246[.]87
108[.]171[.]248[.]182
108[.]171[.]248[.]83
108[.]171[.]248[.]86
108[.]171[.]252[.]41
108[.]171[.]254[.]76
112[.]121[.]164[.]2

Figura 12: IPs de JIB2 [22]

Para almacenar correctamente dichas direcciones IP se les quitará el corchete de los puntos mediante la herramienta Reemplazar…**[.]** por "**.**" Esto se hace mediante la herramienta reemplazar por… Y escribir [.] en la parte de arriba y "**.**" en la parte de abajo.

Dichos datos se analizan de la siguiente manera:

Primero se combinan las direcciones encontradas en ambos JIB en un documento de texto llamado *Ij.list* debido a que necesitamos que esté en ese formato para crear archivo .set de salida. Mediante SilK fue creado un documento con extensión *.set* mediante un *.list*

La figura 13 se muestra cómo funciona de SilK, mediante una instrucción de Linux.
Nos empezamos a familiarizar con la herramienta SilK. Se crea un archivo *.set* a partir de un *.list* y el comando de Linux.

```
jrg@ubuntu:~/Desktop/Ij$ rwsetbuild Ij.list Ij.set
jrg@ubuntu:~/Desktop/Ij$
```

Figura 13: Ejemplo SilK 1

El rwsetbuild crea un fichero binario de tipo IPset de una lista de IPs. Específicamente un archivo IPset binario es creado mediante un archivo IPset partiendo de la entrada del texto. El IPset es escrito en el segundo argumento, si se ha especificado, de lo contrario, el IPset es escrito en la salida estándar si dicha salida no es un terminal. Un archivo existente no será sobrescrito por rwsetbuild a menos que se establezca la variable de entorno SILK_CLOBBER. La entrada de texto es leída desde el primer argumento de la línea de comando si se ha dado el caso de ser especificado; de lo contrario, el texto se lee de la entrada de tipo estándar si dicha entrada no es un terminal. Un posible nombre de archivo de entrada de tipo estándar de stdin (significa la entrada estándar; un nombre de archivo de salida stdout) tiene como significado la salida estándar. rwsetbuild leerá las IP de tipo texto del terminal si la entrada de tipo estándar se proporciona como entrada. rwsetbuild proporciona un error si el archivo de entrada no se puede leer o no es posible escribir el archivo de salida. Dicha información ha sido extraída de [29].

Los archivos SET (archivo de tipo configuración) tienen información sobre la configuración y preferencias que son utilizadas por algunos programas. Cualquier configuración que realiza el usuario dentro de programas de tipos específico se guarda en los archivos de tipo SET. Dicha información procede de [30].

Al abrir el fichero *Ij.set* con el bloc de notas encontramos la siguiente información (figura 14).

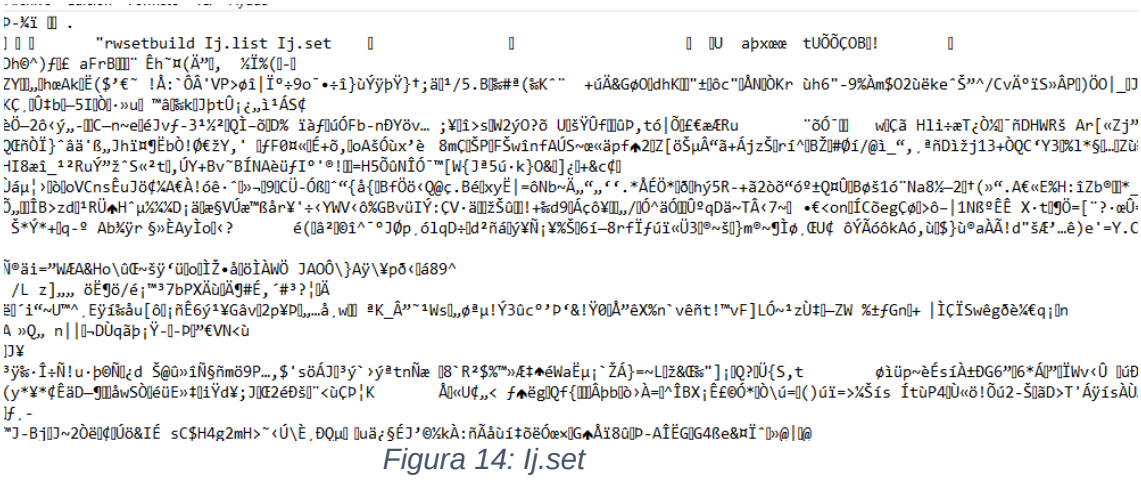


Figura 14: Ij.set

Se puede observar que la información es totalmente críptica.

A partir de este gran documento concatenado (*Ij.list*), creamos un segundo archivo, poniendo cada dirección IP en notación de expresión regular, ^127\.0\.0\.1[[:blank:]]. No está claro por qué hay que cambiarlo de formato, quizás sea porque la herramienta que explicaremos posteriormente (grep) no acepta otro formato. El archivo con los datos cambiados al formato de la expresión regular se llamó *Ij_reg.list.*

La figura 15 refleja las direcciones IP cambiadas de formato, son las direcciones IP que teníamos al principio extraídas de los JIB, con la diferencia de que están en otro tipo de “visualización” diferente a la norma.

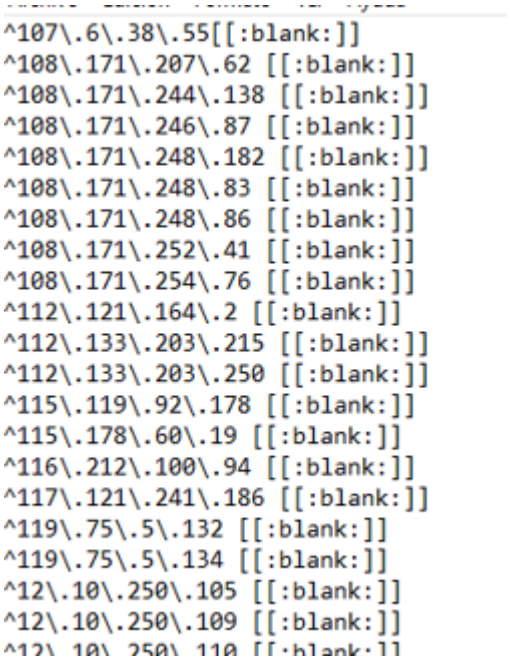


Figura 15: Ejemplo IPs con formato

4.3 Información de IPs de Mandiant

Mientras que en el apartado anterior se ha trabajado con las IPs de los JIB, ahora se manejarán las direcciones IP de la otra “fuente”: IPs de Mandiant.

Mandiant hizo pública una lista de FQDN asociados con la actividad de APT1 como apéndice de su informe. Se realizará una operación DNS para calcular sus direcciones IP. Para calcular las direcciones IP se usará una herramienta especializada que se explicará más adelante. Esto se hace por qué se necesita trabajar únicamente con las direcciones IP y no con sus nombres de dominio.

La figura 16 muestra algunas de las páginas webs asociadas a Mandiant. Dicha información no es fácil de encontrar debido a que hay que hacer una búsqueda exhaustiva en internet, la cual está bastante escondida. El fichero que contiene dichas páginas se encuentra en [18].

```
advanbusiness.com
aoldaily.com
aolon1ine.com
applesoftupdate.com
arrowservice.net
attnpower.com
aunewsonline.com
avvmail.com
bigdepression.net
bigish.net
blackberrycluter.com
blackcake.net
bluecoate.com
booksonlineclub.com
```

Figura 16: Ejemplo FQDN

La figura 17 muestra la herramienta que actúa como conversor de nombres de dominio en direcciones IP mediante la introducción de las direcciones de dominio. Dicha herramienta, de nombre “Herramienta de búsqueda masiva de dominios e IPs” ubicada en una página web. Permite calcular las direcciones IP en muestras de 100 en 100 direcciones de dominio. Está indicada en [23]. Dicha página web es bastante sencilla de utilizar.

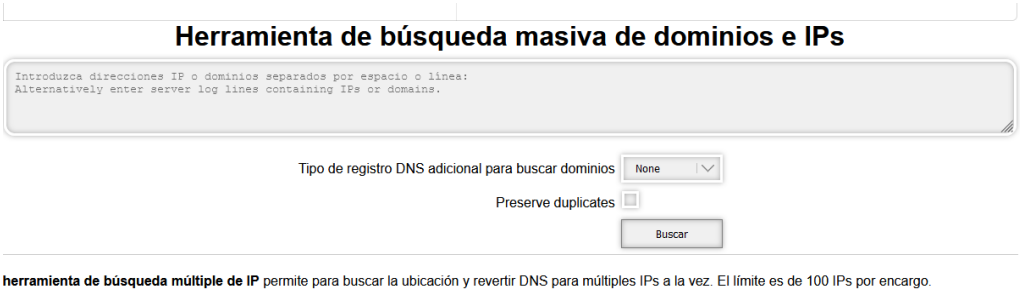


Figura 17: Ejemplo de conversor DNS [23]

La figura 18 muestra las direcciones IP traducidas de los nombres de dominio anteriores, las almacenaremos en *Im.list* .

```
204.11.56.48
154.36.150.146
13.93.163.20
149.255.58.43
208.91.197.27
76.164.206.196
208.91.197.46
104.21.16.223
34.102.136.180
185.230.63.171
3.19.116.195
67.222.16.131
35.186.238.101
```

Figura 18: Direcciones relacionadas

Se juntan los dos archivos en uno (Ij+Im.list), dichos archivos son los formados por las IPs de los JIB y los de la propia Mandiant. Fueron juntados mediante la "herramienta de copia y pega" la información de ambos documentos en un único documento. Después las direcciones IP fueron cambiadas de formato (de nombre "prototipo" ^127\.0\.0\.1[[:blank:]] para cada dirección IP), formando *Ij+Im_reg.list.* Dicho cambio de formato fue realizado de forma manual.

4.4 Internet census 2012

4.4.1 fingerprint

Se tiene una gran cantidad de información relacionada con las direcciones IP según Internet Census 2012, como, por ejemplo, el sistema operativo asociado a una dirección IP. Tenemos una gran cantidad de información a estudiar:

La figura 19 muestra los datos relacionados con el fingerprint del TCP/IP

1.zpaq	10/02/2022 19:10	Archivo ZPAQ	12.399 KB
2.zpaq	04/02/2022 16:43	Archivo ZPAQ	5.489 KB
3.zpaq	04/02/2022 16:32	Archivo ZPAQ	7 KB
4.zpaq	02/02/2022 18:10	Archivo ZPAQ	326 KB
5.zpaq	04/02/2022 16:42	Archivo ZPAQ	3.941 KB
6.zpaq	02/02/2022 18:02	Archivo ZPAQ	7 KB
7.zpaq	02/02/2022 18:08	Archivo ZPAQ	8 KB
8.zpaq	02/02/2022 18:18	Archivo ZPAQ	802 KB
9.zpaq	02/02/2022 18:23	Archivo ZPAQ	7 KB
11.zpaq	04/02/2022 16:38	Archivo ZPAQ	7 KB
12.zpaq	02/02/2022 18:22	Archivo ZPAQ	2.787 KB

Figura 19: Contenido de fingerprint

Se realizará una extracción. La herramienta zpaq es un software que permite descomprimir archivos .zpaq.

PAQ está formado por unos algoritmos de comprensión que no tienen pérdida de tipo open source que permiten lograr las máximas tasas de compresión de archivos, a cambio del mayor uso del CPU y memoria.

ZPAQ es un formato de tipo estándar basado en PAQ que permite el desarrollo de algoritmos de comprensión sin que eso conlleve romper la compatibilidad con anteriores versiones. El formato soporta archivers, la compresión de archivos simples y la compresión de memoria. Alcanza las mayores tasas de compresión en la mayoría de los benchmarks.

La herramienta que funciona mediante línea de comandos zpaq es una utilidad de compresión incremental que tiene funcionalidades que son necesarias para trabajar mediante el formato ZPAQ. Se encuentra disponible para Windows, Linux y OS/X, bajo open source (GPLv3). ZPAQ es más rápido y tiene una mayor tasa de compresión que la mayoría de herramientas de tipo popular, de forma especial cuando se trabaja con backups reales que contienen una gran cantidad de archivos duplicados y otros ya comprimidos, esta información ha sido extraída de [26].

Se ha tenido que usar dicha herramienta para extraer los datos necesarios. Su complejidad de uso es bastante baja. La figura 20 muestra un ejemplo de descompresión zpaq.

Figura 20: Descompresión

A diferencia del ejemplo, se escoge el valor 1 de entre todos los archivos (2,3…) que se hallaban en el zpaq, cuya herramienta para descomprimirlo tiene el mismo nombre.
Después se hace un "filtrado" (de nombre grep), en la que intervienen diversos elementos unidos a un comando Linux en la que extraeremos los valores que coincidan en el primer documento con el segundo. El primer documento tiene los valores de muestras direcciones IP mientras que el segundo tiene la información extraída del fingerprint. Dicha información (1) se extrae del archivo zpaq (podría ser 2, 3…), dependiendo de la información que se quiere estudiar. Esos archivos son muy grandes. Esto se hace debido a que solo queremos estudiar la información de nuestros ejemplos (en *Ij+Im_reg.list*). El ejemplo de uso está en [16]. La figura 21 muestra la instrucción del filtro.

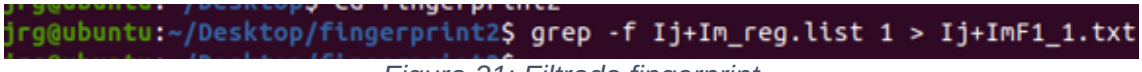


Figura 21: Filtrado fingerprint

La figura 22 muestra los datos asociados a sus direcciones IP. Hay que mirar en detalle, ya que dicha información se encuentra algo oculta. Al ver que aparece el nombre de Linux ya conseguimos ubicar dicha información. Hay que darse cuenta de que las direcciones empiezan por 1.0.4.... El sistema operativo identifica una dirección IP relacionada con él, es decir, en cada dirección IP hay un sistema operativo que les pertenece.

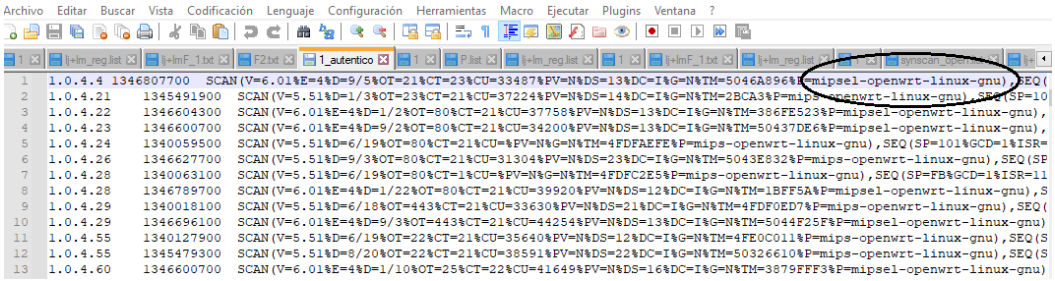


Figura 22: Ejemplo muestra 1

Supuestamente el resultado del filtrado debería contener aquellas direcciones ubicadas en 1 pero solo son coincidentes en *Ij+Im_reg.list,* pero, el archivo de salida está en blanco. Eso significa que no hay coincidencias del *Ij+Im_reg.list* en 1 o que se ha producido algún otro tipo de fallo.

La figura 23 muestra otro ejemplo de direcciones a filtrar Esta vez las direcciones IP empiezan por 2.0.0…Este apartado es igual que el anterior, con la diferencia de que esta vez las direcciones no empiezan por 1.0.4…

Figura 23: Ejemplo muestra 2

He probado a filtrar con el archivo de nombre “2”. Esto lo expreso en la figura 24.

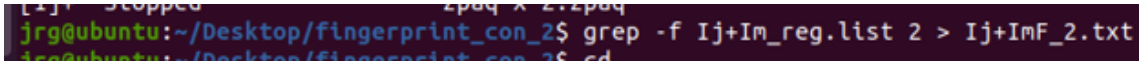
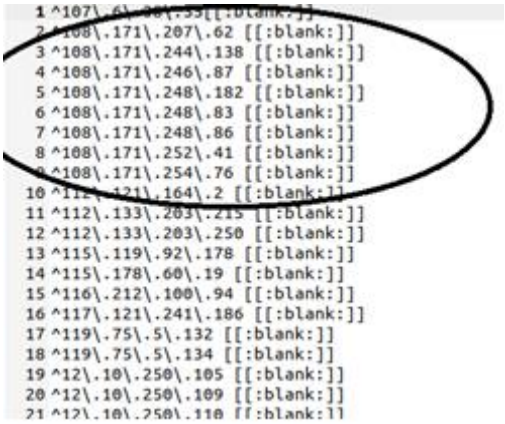


Figura 24: filtrado fingerprint 2

Se han probado muchas combinaciones: usar *Ij+Im.list* en lugar de *Ij+Im_reg.list*, quitar el -f del grep por si se producía un análisis reducido (más discriminatorio), poner todos los archivos en formato txt, cambiar el valor de “2” a uno que sepa que existan las direcciones en ambas partes (esto lo explico en el apartado siguiente)…Esto lo he hecho para poder ver si el filtro funciona, sin éxito.

Ahora se va a probar con otro ejemplo para ver si funciona el filtro, se basa en tener la misma información (o muy parecida) del valor justo delante del *Ij_Im.reg* que el archivo “2”, que esta vez tendrá el número “108”. Esto se hace debido a que puede que, al tratarse de información encontrada en *Ij+im_reg.list* específicamente, el filtro funcione, sabiendo los datos a filtrar a priori. La figura 25 muestra los datos de análisis a examinar, ya cambiado de formato.



```
 1 ^107\.6\.30\.55[[:blank:]]
 2 ^108\.171\.207\.62 [[:blank:]]
 3 ^108\.171\.244\.138 [[:blank:]]
 4 ^108\.171\.246\.87 [[:blank:]]
 5 ^108\.171\.248\.182 [[:blank:]]
 6 ^108\.171\.248\.83 [[:blank:]]
 7 ^108\.171\.248\.86 [[:blank:]]
 8 ^108\.171\.252\.41 [[:blank:]]
 9 ^108\.171\.254\.76 [[:blank:]]
10 ^112\.121\.164\.2 [[:blank:]]
11 ^112\.133\.203\.215 [[:blank:]]
12 ^112\.133\.203\.250 [[:blank:]]
13 ^115\.119\.92\.178 [[:blank:]]
14 ^115\.178\.60\.19 [[:blank:]]
15 ^116\.212\.100\.94 [[:blank:]]
16 ^117\.121\.241\.186 [[:blank:]]
17 ^119\.75\.5\.132 [[:blank:]]
18 ^119\.75\.5\.134 [[:blank:]]
19 ^12\.10\.250\.105 [[:blank:]]
20 ^12\.10\.250\.109 [[:blank:]]
21 ^12\.10\.250\.110 [[:blank:]]
```

Figura 25: Ejemplo con 108

La figura 26 muestran varios ejemplos de los datos completos extraídos de los datos zpaq, que contienen las direcciones IP asociadas al bloque de datos 108.

```
1 108.0.0.1       1346753700
  SCAN(V=6.01%E=4%D=9/4%OT=179%CT=21%CU=36617%PV=N%DS=8%DC=I%G=N%TM=5045D8
  openwrt-linux-
  gnu),SEQ(SP=100%GCD=1%ISR=10D%II=RI%TS=1),SEQ(SP=FB%GCD=1%ISR=108%TS=1),
  %F=AS%RD=0%Q=),T2(R=N),T3(R=N),T4(R=Y%DF=Y%T=40%W=2000%S=A%A=Z%F=R%O=%RD
  %F=AR%O=%RD=0%Q=),T6(R=Y%DF=N%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=),T7(R=Y%DF
2 108.0.1.1       1346395500
  SCAN(V=6.01%E=4%D=1/5%OT=80%CT=21%CU=38457%PV=N%DS=10%DC=I%G=N%TM=38733B
  openwrt-linux-
  gnu),SEQ(SP=102%GCD=1%ISR=10B%TS=1),SEQ(SP=108%GCD=1%ISR=109%II=RI%TS=1)
  %F=AS%RD=0%Q=),T2(R=N),T3(R=N),T4(R=Y%DF=Y%T=3D%W=2000%S=A%A=Z%F=R%O=%RD
  %F=AR%O=%RD=0%Q=),T6(R=Y%DF=N%T=3D%W=0%S=A%A=Z%F=R%O=%RD=0%Q=),T7(R=Y%DF
3 108.0.2.1       1346834700
  SCAN(V=6.01%E=4%D=1/6%OT=179%CT=21%CU=36410%PV=N%DS=10%DC=I%G=N%TM=3874A
  openwrt-linux-
  gnu),SEQ(SP=F6%GCD=1%ISR=107%II=RI%TS=1),SEQ(SP=106%GCD=2%ISR=10B%TI=RD%
  %F=AS%RD=0%Q=),T2(R=N),T3(R=N),T4(R=Y%DF=Y%T=39%W=2000%S=A%A=Z%F=R%O=%RD
  %F=AR%O=%RD=0%Q=),T6(R=Y%DF=N%T=39%W=0%S=A%A=Z%F=R%O=%RD=0%Q=),T7(R=Y%DF
4 108.0.2.189     1346732100
  SCAN(V=6.01%E=4%D=1/15%OT=22%CT=5900%CU=%PV=N%DC=I%G=N%TM=38801C30%P=mip
  openwrt-linux-
  gnu),SEQ(SP=102%GCD=1%ISR=109%TI=Z%TS=8),OPS(O1=M5B4ST11NW7%O2=M5B4ST11N
  %F=AS%RD=0%Q=),T2(R=N),T3(R=N),T4(R=N),T5(R=Y%DF=Y%TG=40%W=0%S=Z%A=S+-
  %F=AR%O=%RD=0%Q=),T6(R=N),T7(R=N),U1(R=N),IE(R=N)
5 108.0.2.1       1346231700
```

Figura 26: Solo 108

Figura 27: Solo 108 otro ejemplo

La figura 28 muestra otro ejemplo de filtrado relacionado con todos los datos comentados anteriormente, sin éxito:

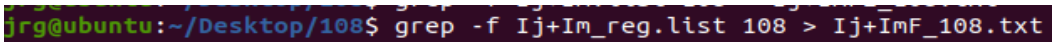


Figura 28: Filtrado con 108

Ningún método ha dado como salida los valores IP esperados. Esto quizás se deba a un fallo del guion.

En el guion tenemos la siguiente información:

La figura 29 y 30 muestran los datos que ofrece el guion.
Su traducción del inglés es la siguiente:

En la figura 28: Para este largo, concatenado argumento, creamos un segundo archivo a partir del archivo con las direcciones IP sin formato (*Ij+Im.list*), ponemos cada dirección IP en la expresión regular ^127\.0.\0\.1[[:blank:]], generándonos *Ij+Im_reg.list* para usarlo en grep -f (más adelante).

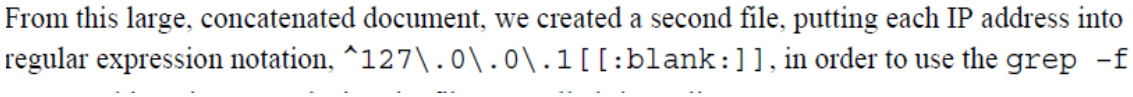
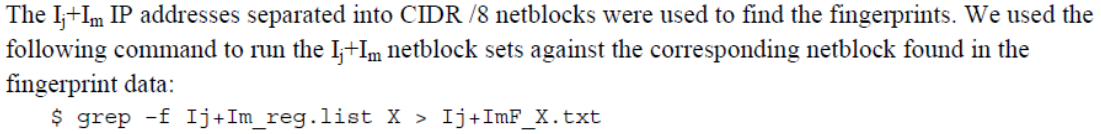

From this large, concatenated document, we created a second file, putting each IP address into regular expression notation, `^127\.0\.0\.1[[:blank:]]`, in order to use the `grep -f`

Figura 29: Ejemplo guión 1

En la figura 29: Las direcciones IPs de *Ij+Imt* han sido separados en /8 bloques de tipo CIDR usados para los fingerprint. Usamos el siguiente comando para “ejecutar” el bloque *Ij+Im_reg.list* con respecto al bloque ubicado en el fingerprint.


The I_j+I_m IP addresses separated into CIDR /8 netblocks were used to find the fingerprints. We used the following command to run the I_j+I_m netblock sets against the corresponding netblock found in the fingerprint data:

```
$ grep -f Ij+Im_reg.list X > Ij+ImF_X.txt
```

Figura 30: Ejemplo guion 2

Otro intento realizado fue sustituir el valor de X por “*”, para ver si así consigue identificar mejor los valores, pero también ha dado resultado una página en blanco.

También se ha intentado otro método para encontrar las características de los datos ubicados en *Ij+Im_reg.list.* Dicho intento se basa en buscar manualmente las direcciones IP contenidos en dicho archivo en el archivo de nombre X (que puede ser 1, 2, …108…) mediante la herramienta "Find".
Por ejemplo, tenemos en *Ij+Im_reg.list* la dirección IP 66.0.135.16, extraemos el archivo de nombre 66 (zpaq) que se encuentra en la carpeta de archivos tcp_fingerprint y, en el Find de dicho archivo (nombre 66) introducimos 66.0.135.16. Si existe en el archivo de nombre 66 extraemos sus características. Este método es muy exhaustivo y pesado, pues habría que examinar muchas direcciones IP. Se han intentado varios ejemplos, sin resultados positivos. Parece ser que tanto en 66 como en el resto de archivos con nombre igual a número el segundo "valor" de la dirección IP no es elevado, abarcando .0, .1 y .2, pero siendo más raro el .3, .4…Aunque el ejemplo mostrado el segundo valor es bajo (un 0), no se ha encontrado en 66.

Como último intento, otro ejemplo de solución puede ser encontrar patrones parecidos en la información de los archivos de fingerprint, abarcando todas sus direcciones IP. Hay que tener en cuenta que, si bien no es posible encontrar la información de una dirección IP concreta en el archivo de nombre 108, se sabe que la máscara de subred es /8 (255.0.0.0) por lo que, aunque no encontremos la dirección deseada de *Ij+Im_reg.list* en concreto, se podrá extraer valores del resto de direcciones que sabemos que pertenecen a la misma red. Por ejemplo, si nuestra dirección IP a estudiar es 108.171.207.62, sabemos que pertenece a la red que comienza por 108…y podemos extraer valores parecidos entre ellos. Encontrar patrones en las direcciones es una tarea demasiado exhaustiva debido a la gran información a analizar por lo que, como un ejemplo de estimación de encontrar una solución concreta y lo más correcta posible, nos quedamos con la información de la dirección IP más cercana (ubicada en los archivos de nombre 108, 2, …) a la queremos estudiar, la que se encuentra en *Ij+Im_reg.list*. Como posible ejemplo, atendiendo a la información anterior de *Ij+Im_reg.list* (108.171.207.62). Buscaremos las direcciones IP que más se acerquen en el archivo de nombre 108. Un ejemplo aleatorio puede ser que se encuentre en 108 una dirección cercana llamada 108.0.203.60, entonces extraeremos la información de una dirección IP cercana a esa dirección IP en *Ij+Im_reg.list* (lo más cercana posible). Todo ello haciendo la suposición que los equipos adyacentes separados por sus direcciones IP "cercanas" tengan el mismo rol. Si nos fijamos en, por ejemplo, el sistema operativo, este coincide en la mayoría (o todos) los casos, es de suponer que todos los equipos tengan la misma naturaleza.

Para una mejor visualización de los datos contenidos en 1 se va a crear una pequeña tabla demostrativa mediante Python y LibreOfficeCalc. El lugar en el que hacemos esto es en una máquina virtual Ubuntu. Las siguientes figuras (31 y 32) muestran tanto el código de introducción de los datos en la tabla de LibreOfficeCalc como los datos ya introducidos en las tablas. Saber cómo crear tablas es muy útil para visualizar mejor la información.

```
import pandas as pd
datos=pd.read_csv('F1.txt',sep='\t',header=None)
datos.columns=['IP','Match','Fingerprint']
datos.to_excel('F1.xlsx')
```

Figura 31: Código de Excel

La información ubicada en LibreOfficeCalc es la misma que la que vimos en anteriores apartados mediante el editor de texto de Ubuntu. Su única función es que se vea mejor la información.

	IP	Match	fingerprint
0	1.0.4.4	1.35E+09	SCAN(V=6.01%E=4%D=9/5%OT=21%CT=23%CU=33487%PV=N%DS=13%DC=I%G
1	1.0.4.21	1.35E+09	SCAN(V=5.51%D=1/3%OT=23%CT=21%CU=37224%PV=N%DS=14%DC=I%G=N%T
2	1.0.4.22	1.35E+09	SCAN(V=6.01%E=4%D=1/2%OT=80%CT=21%CU=37758%PV=N%DS=13%DC=I%G
3	1.0.4.23	1.35E+09	SCAN(V=6.01%E=4%D=9/2%OT=80%CT=21%CU=34200%PV=N%DS=13%DC=I%G
4	1.0.4.24	1.34E+09	SCAN(V=5.51%D=6/19%OT=80%CT=21%CU=%PV=N%G=N%TM=4FDFAEFE%P=mi
5	1.0.4.26	1.35E+09	SCAN(V=5.51%D=9/3%OT=80%CT=21%CU=31304%PV=N%DS=23%DC=I%G=N%T
6	1.0.4.28	1.34E+09	SCAN(V=5.51%D=6/19%OT=80%CT=1%CU=%PV=N%G=N%TM=4FDFC2E5%P=mip
7	1.0.4.28	1.35E+09	SCAN(V=6.01%E=4%D=1/22%OT=80%CT=21%CU=39920%PV=N%DS=12%DC=I%
8	1.0.4.29	1.34E+09	SCAN(V=5.51%D=6/18%OT=443%CT=21%CU=33630%PV=N%DS=21%DC=I%G=N

Figura 32: Tabla de Excel con datos relacionados con fingerprint

4.4.2 synscan

Los datos de synscan tienen información sobre puertos abiertos, filtrados y cerrados para todas las direcciones IP. Se creará un archivo que enumera las direcciones IP en el espacio de las direcciones IPv4 que contenga únicamente puertos abiertos para la optimización del análisis, siguiendo los pasos de [16]. Mientras que en el apartado anterior se ha estudiado la información relacionada con el "fingerprint" (IP, Match, fingerprint), ahora se estudian otros tipos de datos, los puertos asociados. La razón de hacer esto es, en concreto, estudiar dichos puertos abiertos asignados a cada dirección IP. Otra vez tenemos una gran cantidad de información a analizar.

La figura 33 muestra la información relacionada con synscan. Los archivos comprimidos son iguales a los que nos ofrecía "fingerprint" en cuanto al nombre, pero el contenido es diferente. También aparecen archivos comprimidos de tipo zpaq.

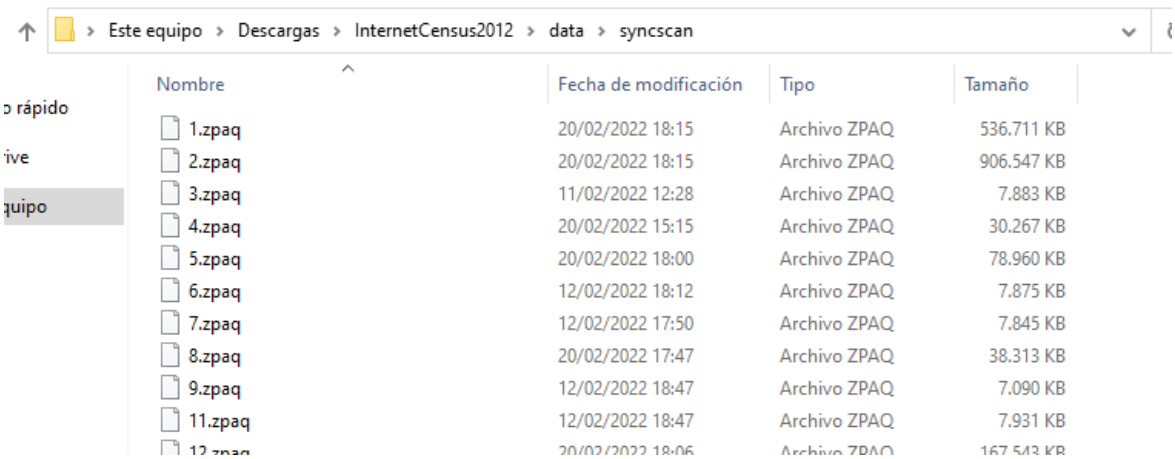


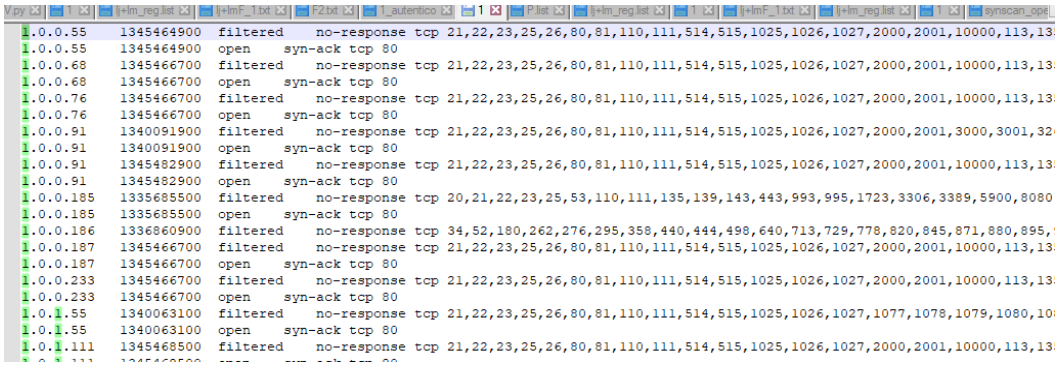
Figura 33: Contenido synscan

Se creará un archivo que extrae cada dirección IP en el espacio de direcciones IPv4 que contiene solo puertos abiertos. La figura 34 muestra un filtro, el criterio del filtro es el filtrado de los puertos abiertos. Llama la atención el formato usado en la información a ser filtrada ("[[:blank:]]open...). También seguimos los pasos de [16].

```
^Z
jrg@ubuntu:~/Desktop/synscan$ grep "[[:blank:]]open[[:blank:]]" * > synscan_open.list
```

Figura 34: Filtrado puertos abiertos

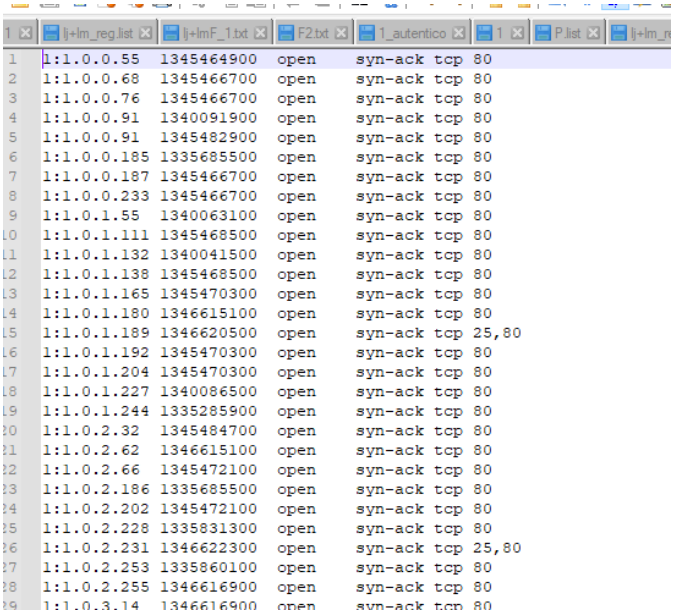
Si se observa la instrucción utilizada en la figura 35, "*" es:



```
1.0.0.55    1345464900  filtered    no-response tcp 21,22,23,25,26,80,81,110,111,514,515,1025,1026,1027,2000,2001,10000,113,13
1.0.0.55    1345464900  open    syn-ack tcp 80
1.0.0.68    1345466700  filtered    no-response tcp 21,22,23,25,26,80,81,110,111,514,515,1025,1026,1027,2000,2001,10000,113,13
1.0.0.68    1345466700  open    syn-ack tcp 80
1.0.0.76    1345466700  filtered    no-response tcp 21,22,23,25,26,80,81,110,111,514,515,1025,1026,1027,2000,2001,10000,113,13
1.0.0.76    1345466700  open    syn-ack tcp 80
1.0.0.91    1340091900  filtered    no-response tcp 21,22,23,25,26,80,81,110,111,514,515,1025,1026,1027,2000,2001,3000,3001,32
1.0.0.91    1340091900  open    syn-ack tcp 80
1.0.0.91    1345482900  filtered    no-response tcp 21,22,23,25,26,80,81,110,111,514,515,1025,1026,1027,2000,2001,10000,113,13
1.0.0.91    1345482900  open    syn-ack tcp 80
1.0.0.185   1335685500  filtered    no-response tcp 20,21,22,23,25,53,110,111,135,139,143,443,993,995,1723,3306,3389,5900,8080
1.0.0.185   1335685500  open    syn-ack tcp 80
1.0.0.186   1336860900  filtered    no-response tcp 34,52,180,262,276,295,358,440,444,498,640,713,729,778,820,845,871,880,895,
1.0.0.187   1345466700  filtered    no-response tcp 21,22,23,25,26,80,81,110,111,514,515,1025,1026,1027,2000,2001,10000,113,13
1.0.0.187   1345466700  open    syn-ack tcp 80
1.0.0.233   1345466700  filtered    no-response tcp 21,22,23,25,26,80,81,110,111,514,515,1025,1026,1027,2000,2001,10000,113,13
1.0.0.233   1345466700  open    syn-ack tcp 80
1.0.1.55    1340063100  filtered    no-response tcp 21,22,23,25,26,80,81,110,111,514,515,1025,1026,1027,1077,1078,1079,1080,10
1.0.1.55    1340063100  open    syn-ack tcp 80
1.0.1.111   1345468500  filtered    no-response tcp 21,22,23,25,26,80,81,110,111,514,515,1025,1026,1027,2000,2001,10000,113,13
```

Figura 35: Ejemplo sin filtrar

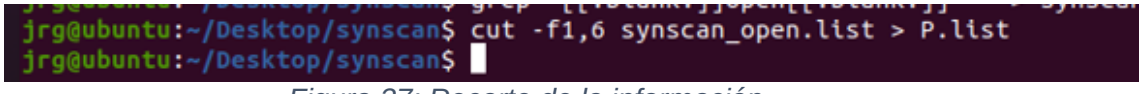
En la figura 35 se ve la información sin filtrar, es decir, se ven todos los puertos (abiertos, filtrados, cerrados), su tipo de trama y su protocolo asociado asignados a cada dirección IP. La figura 36 muestra el filtro a solo direcciones con puertos abiertos, si se observa la instrucción anterior, se refiere a synscan_open.list.



```
1  1:1.0.0.55   1345464900  open    syn-ack tcp 80
2  1:1.0.0.68   1345466700  open    syn-ack tcp 80
3  1:1.0.0.76   1345466700  open    syn-ack tcp 80
4  1:1.0.0.91   1340091900  open    syn-ack tcp 80
5  1:1.0.0.91   1345482900  open    syn-ack tcp 80
6  1:1.0.0.185  1335685500  open    syn-ack tcp 80
7  1:1.0.0.187  1345466700  open    syn-ack tcp 80
8  1:1.0.0.233  1345466700  open    syn-ack tcp 80
9  1:1.0.1.55   1340063100  open    syn-ack tcp 80
10 1:1.0.1.111  1345468500  open    syn-ack tcp 80
11 1:1.0.1.132  1340041500  open    syn-ack tcp 80
12 1:1.0.1.138  1345468500  open    syn-ack tcp 80
13 1:1.0.1.165  1345470300  open    syn-ack tcp 80
14 1:1.0.1.180  1346615100  open    syn-ack tcp 80
15 1:1.0.1.189  1346620500  open    syn-ack tcp 25,80
16 1:1.0.1.192  1345470300  open    syn-ack tcp 80
17 1:1.0.1.204  1345470300  open    syn-ack tcp 80
18 1:1.0.1.227  1340086500  open    syn-ack tcp 80
19 1:1.0.1.244  1335285900  open    syn-ack tcp 80
20 1:1.0.2.32   1345484700  open    syn-ack tcp 80
21 1:1.0.2.62   1346615100  open    syn-ack tcp 80
22 1:1.0.2.66   1345472100  open    syn-ack tcp 80
23 1:1.0.2.186  1335685500  open    syn-ack tcp 80
24 1:1.0.2.202  1345472100  open    syn-ack tcp 80
25 1:1.0.2.228  1335831300  open    syn-ack tcp 80
26 1:1.0.2.231  1346622300  open    syn-ack tcp 25,80
27 1:1.0.2.253  1335860100  open    syn-ack tcp 80
28 1:1.0.2.255  1346616900  open    syn-ack tcp 80
```

Figura 36: Ejemplo todos open

Se puede observar que el filtrado ha sido correcto y solo obtenemos las direcciones IP con los puertos abiertos borrando el resto de información. Luego, usamos el siguiente comando para eliminar el fingerprint y los campos TCP/UDP de sincronización asociado con la lista de puertos abiertos. En la figura 37 podemos ver la instrucción que quita datos del fichero anterior. Esta vez, en vez de filtrar, se corta, para solo quedarnos con parte de la información que nos interesa. Esto se hace con otra instrucción Linux llamada cut. Gracias a dicha herramienta, se eliminan todos los datos exceptuando la primera, la segunda y la quinta columna. Se sigue guiándo con [16].



```
jrg@ubuntu:~/Desktop/synscan$ cut -f1,6 synscan_open.list > P.list
jrg@ubuntu:~/Desktop/synscan$
```

Figura 37: Recorte de la información

P.list nos queda de la siguiente manera:

La figura 38 muestra el fichero completamente recortado. Destaca que solo parece que hay puertos cuyo número es 80. Su nombre es P.list.

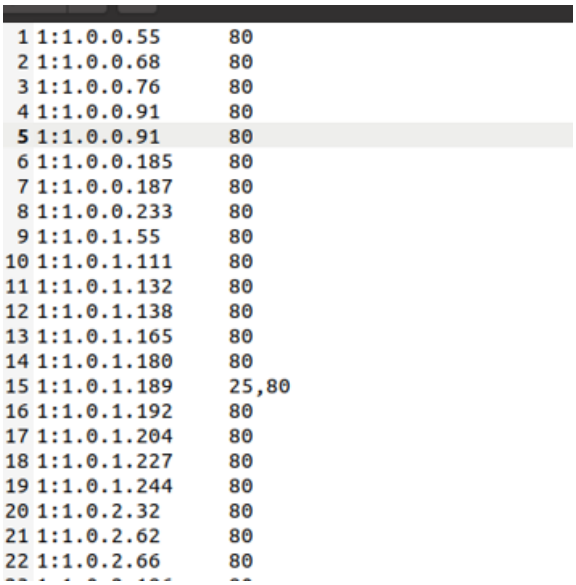


Figura 38: Fichero recortado

Ahora hacemos un filtrado para ver que direcciones IP de *Ij* (direcciones IP de JIB) e *Im* (direcciones IP de Mandiant), todas juntas en un fichero y con el formato cambiado (ver apartados anteriores), de nombre *Ij+Im_reg.list* tienen los puertos abiertos.

La figura 39 muestra el filtrado de las *Ij+Im_reg.list* con el fichero de los puertos, de nombre *P.list* generado en el paso anterior.

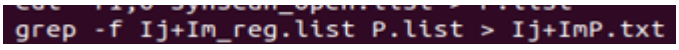
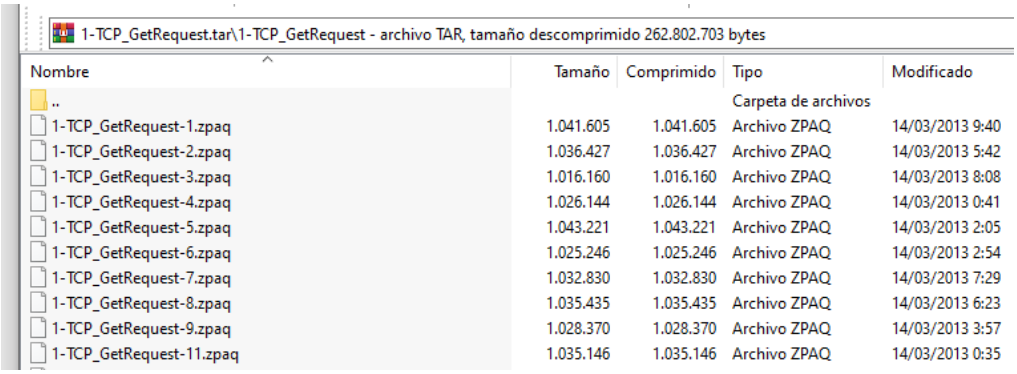


Figura 39: Filtrado puertos abiertos

Desgraciadamente, el archivo de salida está en blanco, por lo que no hay direcciones coincidentes o existe algún otro tipo de error.

4.4.3 service probe

Los datos del servicio (service probe) que se encuentran en Internet Census fueron muy interesantes para nuestro análisis APT1. Se esperaba encontrar datos interesantes enterrados en la respuesta a los servicios para el conjunto *Ij+Im*. Para encontrar esos datos fácilmente, se utilizará SiLK para ver una respuesta de 4 a una sonda de servicio, como viene indicado en [16]. La figura 40 muestra la información relacionada con el service probe sin extraer.



1-TCP_GetRequest.tar\1-TCP_GetRequest - archivo TAR, tamaño descomprimido 262.802.703 bytes

Nombre	Tamaño	Comprimido	Tipo	Modificado
..			Carpeta de archivos	
1-TCP_GetRequest-1.zpaq	1.041.605	1.041.605	Archivo ZPAQ	14/03/2013 9:40
1-TCP_GetRequest-2.zpaq	1.036.427	1.036.427	Archivo ZPAQ	14/03/2013 5:42
1-TCP_GetRequest-3.zpaq	1.016.160	1.016.160	Archivo ZPAQ	14/03/2013 8:08
1-TCP_GetRequest-4.zpaq	1.026.144	1.026.144	Archivo ZPAQ	14/03/2013 0:41
1-TCP_GetRequest-5.zpaq	1.043.221	1.043.221	Archivo ZPAQ	14/03/2013 2:05
1-TCP_GetRequest-6.zpaq	1.025.246	1.025.246	Archivo ZPAQ	14/03/2013 2:54
1-TCP_GetRequest-7.zpaq	1.032.830	1.032.830	Archivo ZPAQ	14/03/2013 7:29
1-TCP_GetRequest-8.zpaq	1.035.435	1.035.435	Archivo ZPAQ	14/03/2013 6:23
1-TCP_GetRequest-9.zpaq	1.028.370	1.028.370	Archivo ZPAQ	14/03/2013 3:57
1-TCP_GetRequest-11.zpaq	1.035.146	1.035.146	Archivo ZPAQ	14/03/2013 0:35

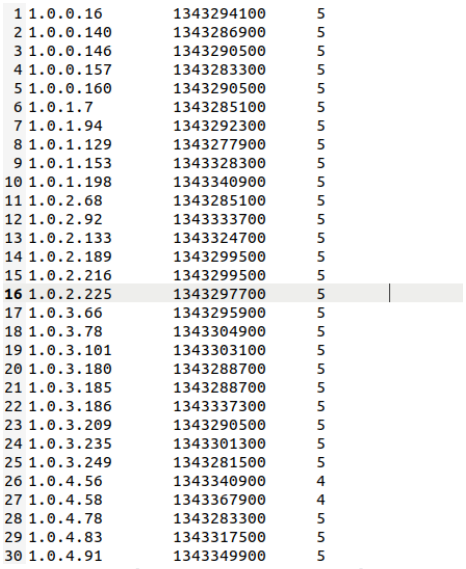
Figura 40: Información service probe

La figura 41 muestra las instrucciones para obtener los datos de “service probe” de valor 4, un recorte, un cambio con SilK y una intersección con SilK. Algunas herramientas usadas son iguales a las de apartados anteriores; otras no, como por ejemplo la intersección (rwsettool –interserct…). Las instrucciones son las siguientes, seguidamente se explica lo que hace cada instrucción, además de que es cada cosa dentro de la instrucción:

```
jrg@ubuntu:~/Desktop/service_probe2$ grep "[[:blank:]]4[[:blank:]]" * > servicep
robe_4.list
jrg@ubuntu:~/Desktop/service_probe2$ cut -f1 serviceprobe_4.list > serviceprobe_
4_IPS.list
jrg@ubuntu:~/Desktop/service_probe2$ rwsetbuild serviceprobe_4_IPS.list servicep
robe_4_IPS.set
jrg@ubuntu:~/Desktop/service_probe2$ rwsettool --intersect Ij.set serviceprobe_4
_IPS.set | rwsetcat | sort > Ij_sp1_intersection.txt
```

Figura 41: Filtrado según el número de servicio, recorte del fichero de salida, conversor con SilK, realización de intersección

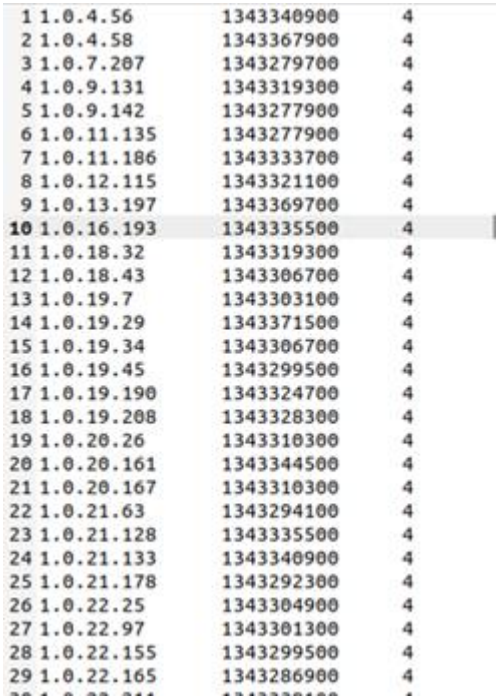
Primero se obtienen solo con todas las direcciones IP con número de servicio 4. Luego se corta la información para ver más en detalle. Se genera el *.set* del archivo anterior. Al final, se hace una intersección del *Ij.set* generado en apartados anteriores con el *.set* generado en la instrucción anterior (*serviceprobe_4_IPS.set*). La figura 42 muestra todos los archivos sin filtrar. En ella se contiene la información de todos los servicios, con muchos 5 y pocos 4.



1	1.0.0.16	1343294100	5
2	1.0.0.140	1343286900	5
3	1.0.0.146	1343290500	5
4	1.0.0.157	1343283300	5
5	1.0.0.160	1343290500	5
6	1.0.1.7	1343285100	5
7	1.0.1.94	1343292300	5
8	1.0.1.129	1343277900	5
9	1.0.1.153	1343328300	5
10	1.0.1.198	1343340900	5
11	1.0.2.68	1343285100	5
12	1.0.2.92	1343333700	5
13	1.0.2.133	1343324700	5
14	1.0.2.189	1343299500	5
15	1.0.2.216	1343299500	5
16	1.0.2.225	1343297700	5
17	1.0.3.66	1343295900	5
18	1.0.3.78	1343304900	5
19	1.0.3.101	1343303100	5
20	1.0.3.180	1343288700	5
21	1.0.3.185	1343288700	5
22	1.0.3.186	1343337300	5
23	1.0.3.209	1343290500	5
24	1.0.3.235	1343301300	5
25	1.0.3.249	1343281500	5
26	1.0.4.56	1343340900	4
27	1.0.4.58	1343367900	4
28	1.0.4.78	1343283300	5
29	1.0.4.83	1343317500	5
30	1.0.4.91	1343349900	5

Figura 42: Datos con la información de todos los números de servicios

La figura 43 muestra la información cortada y filtrada, de nombre *serviceprobe_4.list*, en la que solo aparecen aquellas direcciones IP de número de servicio 4.



1	1.0.4.56	1343340900	4
2	1.0.4.58	1343367900	4
3	1.0.7.207	1343279700	4
4	1.0.9.131	1343319300	4
5	1.0.9.142	1343277900	4
6	1.0.11.135	1343277900	4
7	1.0.11.186	1343333700	4
8	1.0.12.115	1343321100	4
9	1.0.13.197	1343369700	4
10	1.0.16.193	1343335500	4
11	1.0.18.32	1343319300	4
12	1.0.18.43	1343306700	4
13	1.0.19.7	1343303100	4
14	1.0.19.29	1343371500	4
15	1.0.19.34	1343306700	4
16	1.0.19.45	1343299500	4
17	1.0.19.190	1343324700	4
18	1.0.19.208	1343328300	4
19	1.0.20.26	1343310300	4
20	1.0.20.161	1343344500	4
21	1.0.20.167	1343310300	4
22	1.0.21.63	1343294100	4
23	1.0.21.128	1343335500	4
24	1.0.21.133	1343340900	4
25	1.0.21.178	1343292300	4
26	1.0.22.25	1343304900	4
27	1.0.22.97	1343301300	4
28	1.0.22.155	1343299500	4
29	1.0.22.165	1343286900	4

Figura 43: Información de las IP con número de servicio 4

La figura 44 muestra solo las direcciones IP asociadas al servicio número 4, de nombre serviceprobe_4_IPS.list

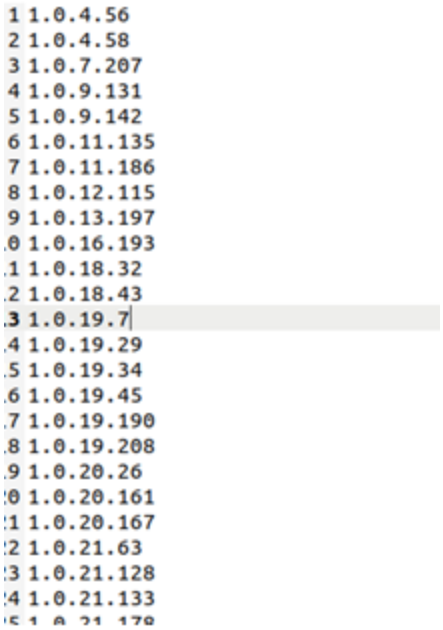


Figura 44: Solo IPs como servicio 4

Enlazando con lo explicado anteriormente, en el comando SilK llamado rwsettool, que se utiliza en una de las instrucciones ejecutadas anteriormente para realizar una intersección, se da uso al archivo lj.set que fue calculada en el primer apartado.
Desgraciadamente no hay ningún valor en lj_sp1_intersection.txt por lo que no existe tal intersección. Es decir, la herramienta ha fallado, el único motivo posible es que la instrucción sea errónea en el guion.

4.5 ASN

SilK nos dará acceso a la creación de archivos pmap, y después podremos ejecutar un comando de SilK llamado rwtuc para el encuentro de los ASN (Autonomous System Number) que se asocian con cada dirección IP, todo ello mediante instrucciones en Linux. Los archivos de tipo pmap tienen dentro solo los ASN que se asocian con las direcciones IP. Viene de forma específica reflejado en [16]. Esto se hace porque es interesante estudiar que ASN tienen exclusivamente las direcciones IP almacenadas anteriormente.

En la figura 45 se puede ver las direcciones IP asociadas a su correspondiente ASN (número al final de cada frase) y también se puede la “máscara” de subred. Dichas direcciones IP pertenecen a: España, Francia, Alemania, Finlandia, Canadá, China, Japón y Rusia.

```
192.137.126.0/16 33
45.228.236.0/24 6
193.174.0.0/16 28
90.147.0.0/16 137
45.182.198.0/23 1
136.236.0.0/23 33
72.163.0.0/22 109
72.163.5.0/24 109
45.182.198.0/23 1
```

Figura 45: IPs con países con su ASN asociado

En la figura 46 podemos ver la creación del .pmap derivada del archivo anterior, esta vez usaremos una instrucción nueva llamada rwpmapbuild.

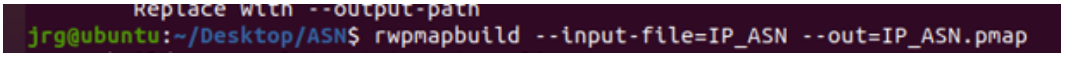


Figura 46: Obtención del pmap

rwpmapbuild lee una secuencia de texto delimitada por espacios en blanco y escribe una secuencia de salida binaria que representa un mapa de prefijo. La sintaxis de esta entrada se describe en la sección "FORMATO DE ARCHIVO DE ENTRADA" a continuación.

La entrada de texto es leída desde el archivo que es especificado por –input-path o desde la entrada estándar cuando no es proporcionado por el interruptor. La salida de tipo binario es escrita en la ubicación nombrada por –output-path o en la salida de tipo estándar cuando no es proporcionado el interruptor y no está conectada a un terminal.

Un archivo de asignación de prefijos es un archivo de tipo binario que asigna un valor (de forma específica una dirección IP o un par de puertos de tipo protocolo) a una etiqueta de cadena.

Cuando se haya creado un archivo de mapa de prefijos, se puede usar el archivo en rwfilter(1), rwstats(1), rwuniq(1), rwgroup(1), rwsort(1) o rwcut(1) para contar, ordenar, mostrar, particionar los registros de flujo de SilK en función de las etiquetas definidas en el mapa de prefijos. La página del manual de pmpapfilter(3) tiene más información. Para la visualización de un archivo de mapa de prefijos, utilice rwpmapcat(1). Para la consulta del contenido de un mapa de prefijos, utilice rwpmaplookup(1)

Toda esta información se ha extraído de [31]

La figura 47 muestra la página web la cuál hemos extraído la asociación ASN (33, 6, 28…). La herramienta es bastante intuitiva y esclarecedora, permitiéndonos extraer los ASN de cada dirección IP de forma rápida, cómoda y eficiente. En el ejemplo estudiado posteriormente vemos una serie de países con su “Flag”, “Country Code” y ASN asociado.

What Is An ASN?

An autonomous system number (ASN) is a unique number assigned to an autonomous system (AS) by the Internet Assigned Numbers Authority (IANA).

An AS consists of blocks of IP addresses which have a distinctly defined policy for accessing external networks and are administered by a single organization but may be made up of several operators.

Below you will find a list of public ASN's listed by country.

Country	Flag	Country Code	ASN's
Andorra		AD	52
United Arab Emirates		AE	1005
Afghanistan		AF	511
Antigua and Barbuda		AG	121
Anguilla		AI	42
Albania		AL	685

Figura 47: Fuente de datos ASN [20]

Una vez creado el pmap, se usará el siguiente comando para asociar la lista de direcciones IP APT1 con el ASN apropiado:

La figura 48 muestra el uso de SilK que relaciona el .list con un .pmap.
rwtuc lee archivos de texto que tienen un formato similar al producido por rwcut(1) e intenta crear un registro “SiLK Flow” para cada línea de entrada, esta información se encuentra en la página oficial de SilK, indicada en [27]. rwcut lee los registros binarios de SilK Flow e imprime aquellos atributos (o campos) que tienen un registro que han sido seleccionados por el usuario en un terminal con formato textual delimitado por barras.

```
jrg@ubuntu:~/Desktop/ASN$ rwtuc --field=sip Ij+Im.list| rwuniq --pmap-file=asn:IP_ASN.p
map --field=src-asn --no-col > Ij+ImA.txt
```

Figura 48: Asociación pmap

Se ha realizado utilizado el siguiente archivo (*Ij+Im.list*). Esto sirve de enlace con pasos anteriores, en el que también usamos dicho archivo.
El resultado Ij+ImA.txt contiene la siguiente información, en la figura 49:

```
1 src-asn|Records|
2 UNKNOWN|1253|
```

Figura 49: resultado rwtuc

Un apartado que se intentó sin éxito fue el siguiente:

4.6 Routing Data y Country Code

Como alternativa a ASN se decide usar los datos de Neustar para poder determinar de qué tipo es el enrutamiento y cuál es el país con más precisión. Los datos de Neustar incluyen un código de país, un rango de direcciones, ciudad y estado donde han estado

ese rango de direcciones y un enrutamiento y tipo de conexión. El tipo de enrutamiento se asocia al tipo de conexión (DSL, fibra...). Se hizo un tipo de conexión, se enumeró en el archivo para un rango de direcciones IP, entonces se ignoró el tipo de conexión en nuestro análisis, se cortaron los rangos de direcciones IP y los datos de enrutamiento. Luego se usó SilK para crear un pmap. Luego, se replica el proceso que contiene el estado y la ciudad y, una vez más usamos SilK para crear un pmap.
Usamos estos comandos:

```
$ gzip -dc neustar.csv.gz | cut -d"," -f1,2,5 | awk -F"," '{print $1 " " \
$2 " " $3}' | awk -F"\"" '{print $2 " " $4 " " $6}' | grep -v \
"start" | rwpmapbuild --in=- --out=routingConnection.pmap
$ gzip -dc neustar.csv.gz | cut -d"," -f1,2,5 | awk -F"," '{print $1 " " \
$2 " " $3}' | awk -F"\"" '{print $2 " " $4 " " $6}' | grep -v \
"start" | rwpmapbuild --in=- --out=countryState.pmap
$ rwtuc --field=sip Ij+Im.list | rwuniq --pmap- \
file=route:routingConnection.pmap --field=src-route --no-col > \
Ij+ImR.txt
```

4.7 Open Resolvers

Los datos de Open Resolver fueron proporcionados en un archivo gzip. Mediante el uso de estos comandos, se desempaquetará el archivo, se creará una lista de direcciones IP y luego fue creado un conjunto de tipo SilK

```
$ gzip -dc open_resolvers.out.gz | grep ":0:1:[0-9]" | cut –d":" –f3 \
> open_resolvers.list
$ rwsetbuild open_resolvers.list open_resolvers.set
```

A continuación, se usa SilK para poder encontrar una intersección entre el grupo Ij+Im de la Open Resolver List y las direcciones IP de marcado de tipo Open Resolver.

```
$ rwsettool --intersect open_resolvers.set Ij+Im.set | rwsetcat \
| sort > Ij+ImO.txt
```

Este apartado y el anterior no han sido realizados correctamente debido a que no se ha podido encontrar neustar.csv.gz ni open_resolvers.out.gz. La información ha sido extraída de [16].

Como añadido extra y también como posible ayuda para discriminar entre direcciones IP necesarias en nuestros ejemplos, ¿qué pasaría si tenemos muchas direcciones IP y solo queremos extraer las 100000 primeras? (Las direcciones se han extraído de una página que contenía direcciones de todos los países, pero, cuando he vuelto a acceder a ella me ha dado un problema de seguridad)

La figura 50 muestra un conjunto de direcciones IP masivo, formado por un conjunto muy grande de direcciones IP, un total de 206350.

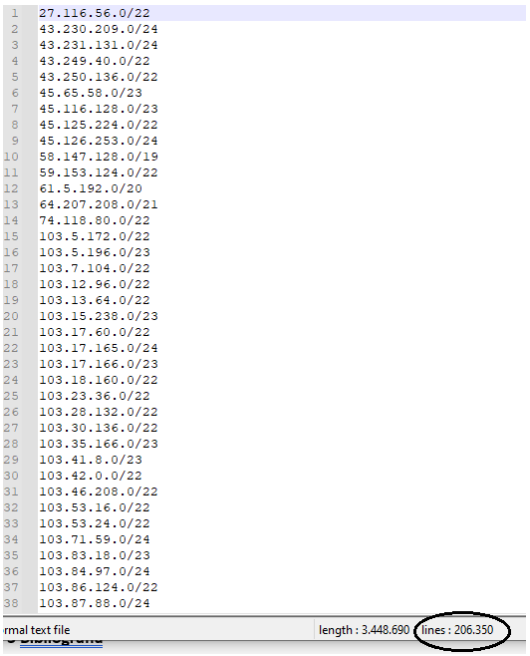


Figura 50: all_ips

La figura 51 muestra cómo, mediante SilK, recogemos 100000 muestras.

```
jrg@ubuntu:~/Desktop/SampleSet$ rwsettool --sample --size 100000 all_ips.set | r
wsetcat | sort > S.list
```

Figura 51: Instrucción all_ips.set

Fueron elegidas 100000 muestras al azar. La figura 52 muestra el total de 100000 direcciones IP generadas por la instrucción anterior. Todos estos pasos se han extraído de [16].

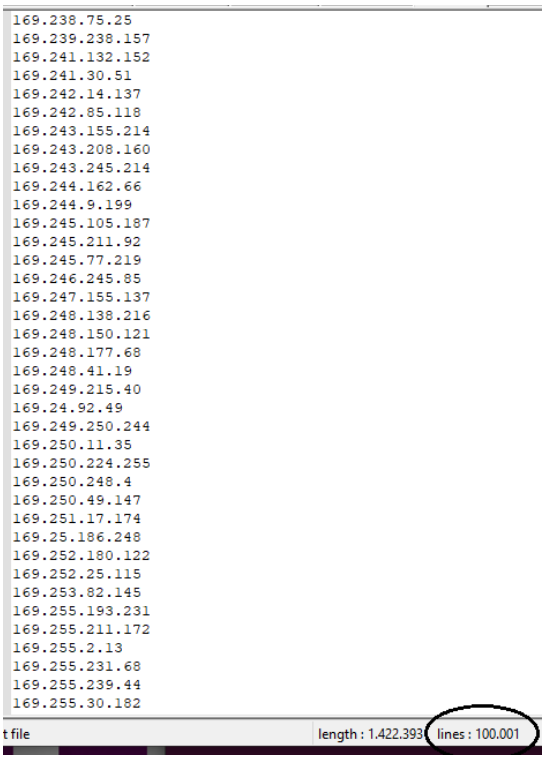


Figura 52: 100000 muestras

5 Estudio de WEBC2-DIV (y Poison Ivy)

En este apartado se va a estudiar el malware WEBC2-DIV debido a que es usado como parte del ataque APT1. Este apartado es completamente independiente de los apartados anteriores, en los que prácticamente solo se estudiaban direcciones IP con información asociada a ellas.

Los “backdoors beachhead” que usa APT1 parece que son lo que llamamos WEBC2. Dichos backdoors de tipo WEBC2 son con cierta probabilidad los más comunes de APT1. Como podemos ver en [13]

WEBC2-DIV realiza una búsqueda en las “instrucciones” “<div safe” y “balance”> para la delimitación de la información que está codificada de C2. Si la cadena que está decodificada comienza con “J”, el virus realizará un análisis de más argumentos en la cadena que está codificada para la especificación de cuál es el intervalo que está suspendido, que es para lo que sirve dicha “J”. Los detalles del funcionamiento de este malware se muestran en [18].

5.1 InetSim

Es necesario un “laboratorio virtual” que ejerza de “servidor” para que, en caso de ataque, se dañe solamente la interfaz virtual en vez de la real. Esto es muy útil para estudiar malware. Para instalar InetSim se realizan los siguientes pasos:

La figura 53 muestra la línea con el enlace del InetSim. Se necesita esta vez un conocimiento un poco más amplio del uso de Linux. Lo que hacemos en este apartado es escribir una dirección web relacionada con el programa InetSim en un archivo determinado mediante el uso de “nano”.

Figura 53: Instrucción necesaria

Luego se siguen las siguientes instrucciones:

Realizar sudo apt-get upgrade
Realizar sudo apt-get install inetsim

Para configurar InetSim se realizan los siguientes pasos:

Se calcula la dirección IP de nuestra máquina virtual:

La figura 54 muestra la dirección IP de nuestra máquina virtual:
Se necesita saber ciertas funciones de la máquina virtual para extraer esta información. En esta ocasión está en el apartado “red”, que se encuentra en la parte superior derecha del menú de dicha máquina.

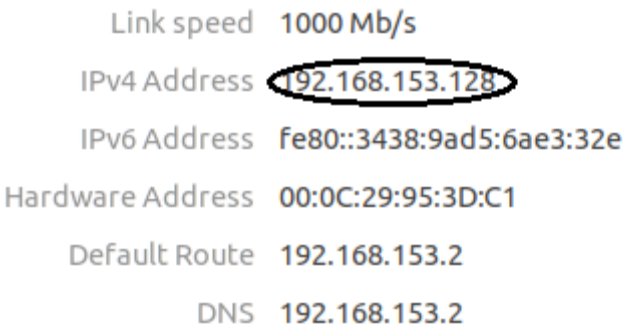


Figura 54: IP máquina virtual

En el fichero inetsim.conf se modifica el service_bin_address y el dns_default_ip. En las figuras 55 y 56 vemos esas modificaciones.
En un caso "se creará" una dirección IP asociada y en la otra un "dns" asociado, todo esto se hace para un correcto adecuado de la configuración de InetSim, es decir, será esta su dirección IP asignada.

```
60 ##########################################
61 # service_bind_address
62 #
63 # IP address to bind services to
64 #
65 # Syntax: service_bind_address <IP address>
66 #
67 # Default: 127.0.0.1
68 #
69 #service_bind_address 10.10.10.1
70 service_bind_address 192.168.153.128
71
72 ##########################################
```

Figura 55: Información service_bind_address

```
197
198 ########################################
199 # dns_default_ip
200 #
201 # Default IP address to return with DNS replies
202 #
203 # Syntax: dns_default_ip <IP address>
204 #
205 # Default: 127.0.0.1
206 #
207 #dns_default_ip 10.10.10.1
208 dns_default_ip 192.168.153.128
209
```

Figura 56: Información DNS

En la figura 57 se muestra la escritura de un archivo a otro en otra dirección. Este apartado es muy importante, pues la información real que se usa está en una carpeta externa. Otra vez usamos un comando Linux para copiar esa información. Sudo nos permite ejecutar comandos de Linux que necesiten permisos de administrador.

```
jrg@jrg-VirtualBox:~/Escritorio$ sudo cp inetsim.conf /etc/inetsim/inetsim.conf
[sudo] contraseña para jrg:
```

Figura 57: Copia de un inetsim a otro

Ahora el siguiente paso es ejecutar el InetSim. Miramos el valor en Inetsim.pid:

En la figura 58 vemos el valor a “matar”. Se muestra el valor inetsim.

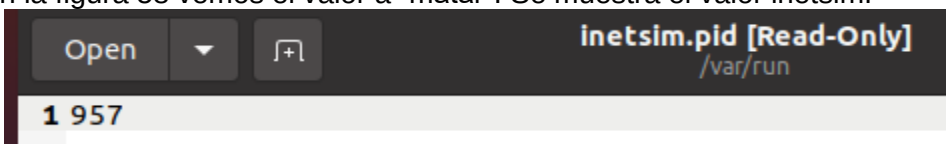


Figura 58: valor inetsim

Se mata el proceso. En la figura 59 se puede ver como se mata al proceso. Cómo viene siendo habitual, usamos un comando de Linux.

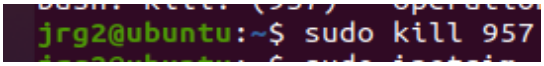


Figura 59: matar el proceso

Se inicia el InetSim. Se puede observar que la dirección IP donde está escuchando es la obtenida anteriormente. Las figuras 60 y 61 muestran el funcionamiento de InetSim.

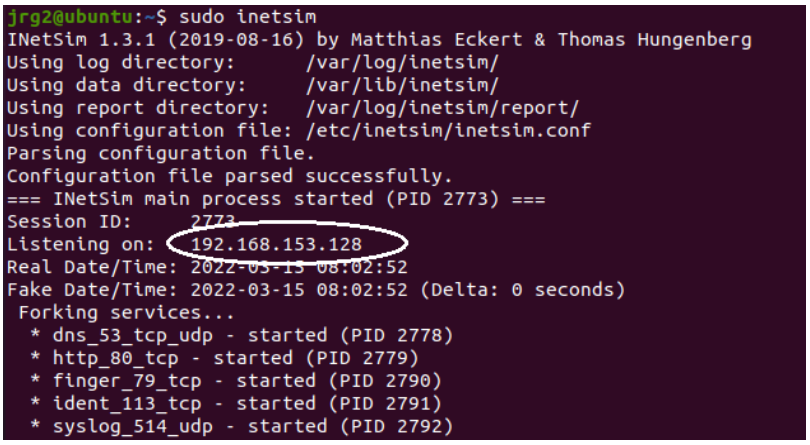


Figura 60: Información IP 1

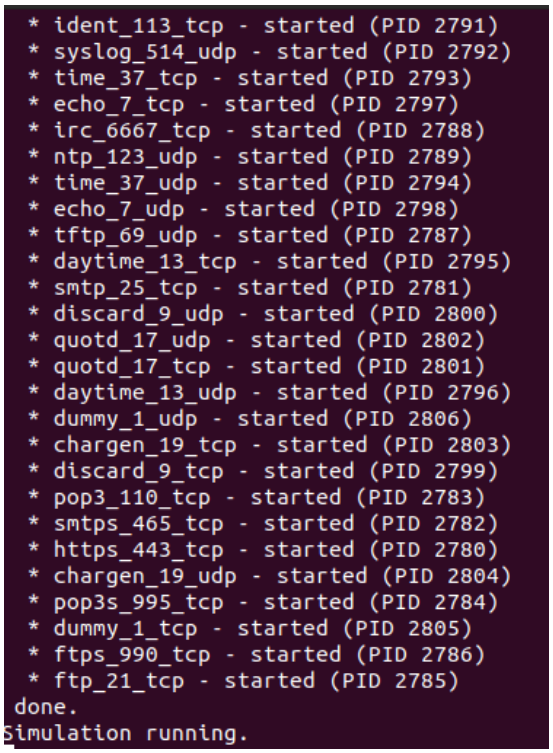


Figura 61: Información IP 2

Se cambian las direcciones IP de la máquina Windows para su posterior conexión con la máquina Ubuntu. La figura 62 muestra la configuración de red de Windows.

Para ello hay que conocer cómo acceder a dicha configuración en Windows. Para ello se accede al apartado de configuración de redes, elegimos la “interfaz” adecuada para ser modificada mediante las propiedades y, dentro de dichas propiedades, se selecciona “protocolo de internet versión 4” para cambiar las direcciones IP para su conexión con la otra máquina virtual. A modo de enlace con lo anterior, fijarse en que tanto la puerta de enlace predeterminada como el servidor DNS preferido tienen como dirección IP la asignada por InetSim.

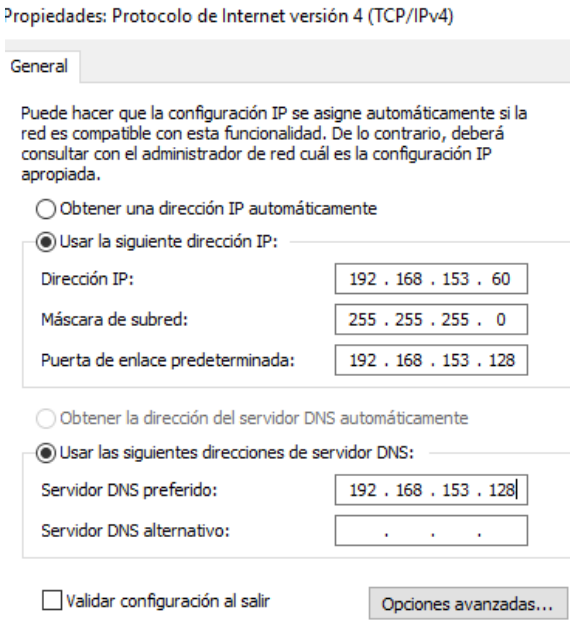


Figura 62: Configuración de red en Windows

Todos estos pasos se han seguido mediante el video indicado en [17].

5.2 Process Hacker y Wireshark

Todos los pasos posteriores se extraen de [15].

Gracias a la configuración de InetSim en la máquina Ubuntu y la configuración de interfaz de red de Windows, realizada en el apartado anterior, tenemos el “laboratorio” creado para intercambiar información de una máquina hacia otra. Para conseguir una muestra del malware, se accederá a la página web ubicada en [28], para ello, se descargó y se pidió la contraseña al responsable de la página. Se va a examinar nuestra muestra de WEBC2-DIV. Primero cambiamos el formato a un formato .exe en la máquina Windows y la ejecutamos simplemente haciendo “doble click”. La figura 63 muestra los procesos en activo, haciendo hincapié en el malware WEBC2-DIV. Dichos procesos se muestran en la herramienta Process Hacker.

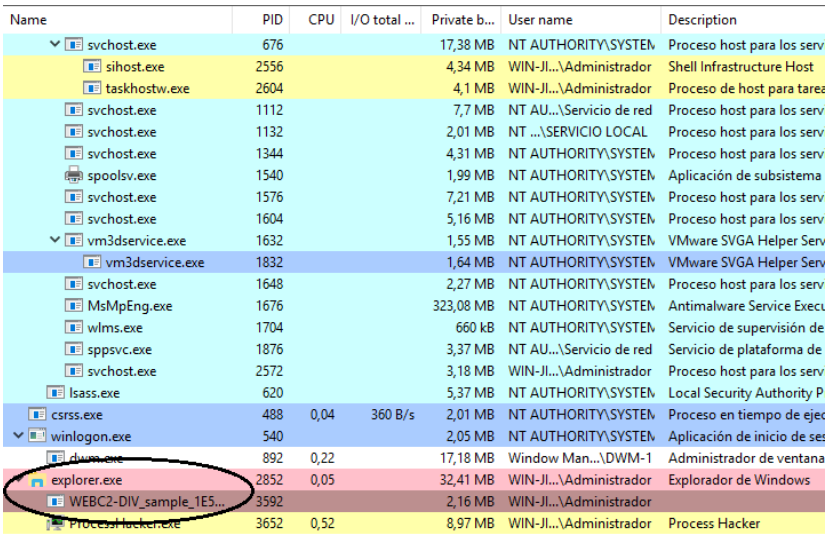


Figura 63: Información WEBC2-DIV

Mediante Wireshark, ubicada en la máquina Ubuntu, se observa la conexión a thecrownsgolf.org y las respectivas tramas asociadas a las dos siguientes respuestas. Que aparezcan es señal de que el malware se ha ejecutado de forma correcta. La figura 64 muestra la monitorización del funcionamiento de la muestra.

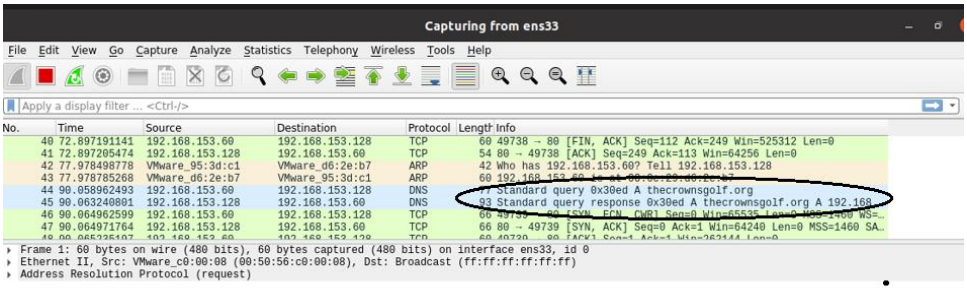


Figura 64: Ejemplo de comunicación con página Web

Tercero, las figuras 65 y 66 muestran información acerca de las tramas siguientes a la descrita anteriormente. Hay que saber manejarse por el Wireshark de forma más o menos básica para acceder a una información más detallada de las tramas. Para ello se usa botón derecho en la trama correspondiente y se selecciona TCP Stream. En ella se pueden ver datos relacionados con el tipo de respuesta (GET), el User-Aggent, el Host y un cuerpo de tipo html con información relacionada con el uso de InsetSim.

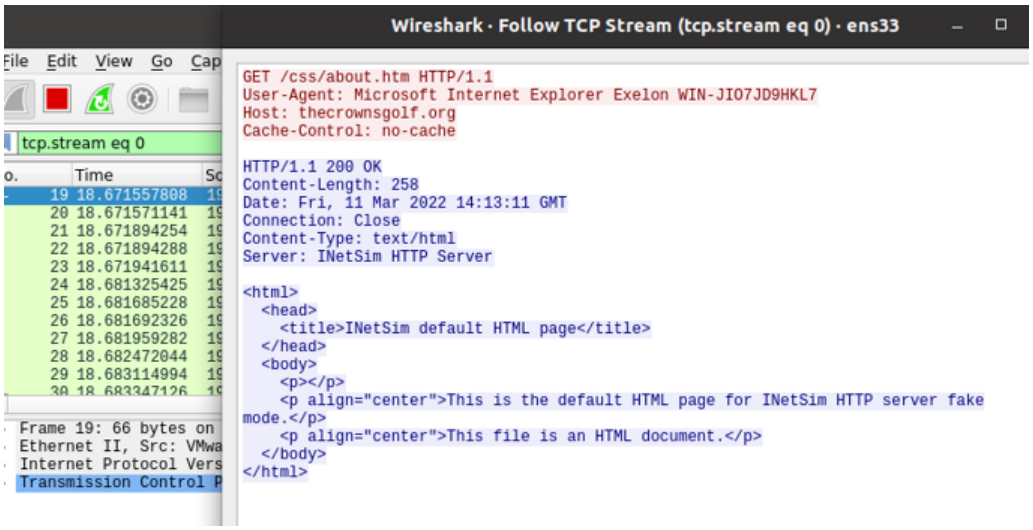


Figura 65: Información respuesta 1

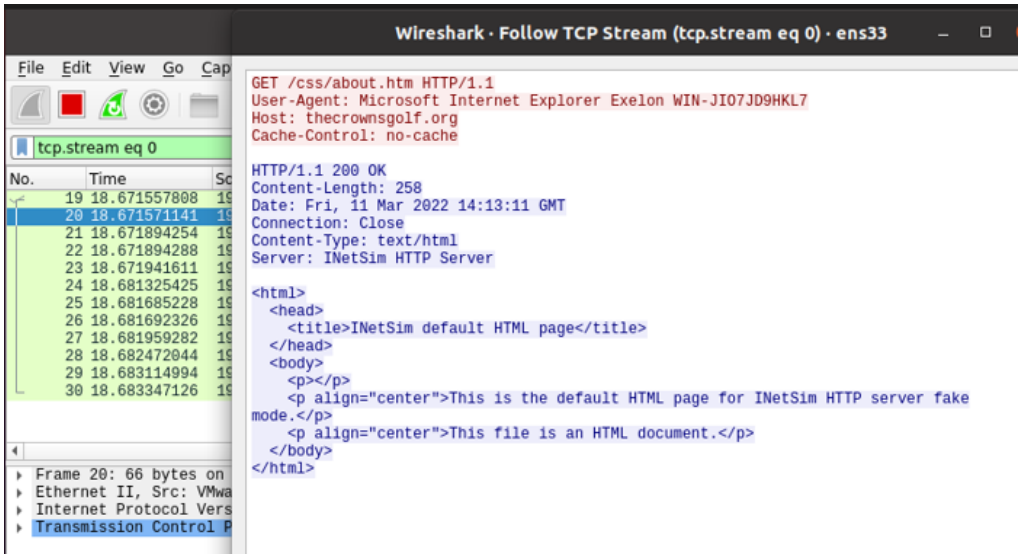


Figura 66: Información respuesta 2

La figura 67 muestra el nombre del servidor. Para obtenerla podemos usar un “comando Windows”.

Figura 67: Nombre del server

Se observa que el User-Agent es el mismo que en %computername%. En el ejemplo se está supuestamente infectando thecorwnsgolf.org al hacer una petición GET a una página ubicada en dicho host.

5.3 IDA

En este apartado se usará el programa llamado IDA para examinar el malware WEBC2-DIV. Los resultados de este apartado son hallados mediante ejemplos basados en la ejecución de instrucciones ubicados en una página web, dichos ejemplo serán llamados

el “tutorial”. Se ha mezclado información obtenida por mí mismo con información de la web que he sido usada como “tutorial”. La información que ha sido extraída directamente del tutorial ha sido hecha debido a que ciertas soluciones alcanzadas “manualmente” no coincidían con las reflejadas en la página web de dicho tutorial. Este apartado tiene poca relación con el apartado anterior, ya que esta vez analizamos el malware “directamente”. En los siguientes apartados se visualizará información relacionada con direcciones e instrucciones. Nos vamos a ir moviendo entre dichas direcciones, interpretando la información de dichas instrucciones. El “tutorial” está ubicado en [15].

Lo primero de todo es convertir la información mostrada en forma de texto. La figura 68 muestra como transformar en texto los diagramas mediante el comando correspondiente dentro de las opciones de la herramienta. Esto es debido a que necesitamos ver las instrucciones en un formato adecuado para entenderlas.

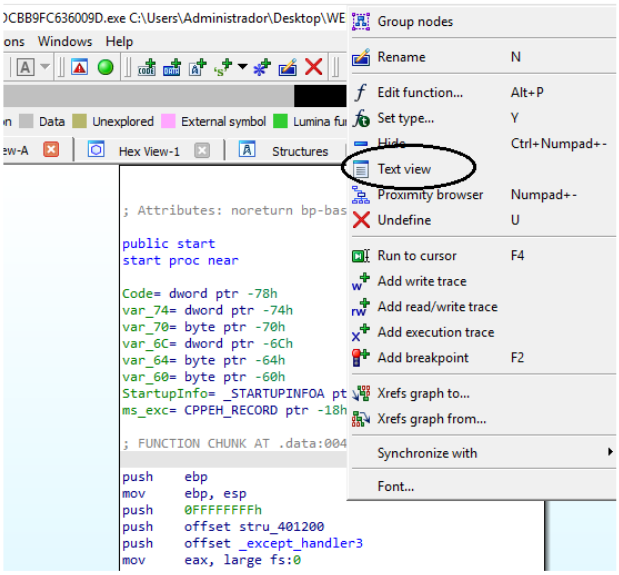


Figura 68: Text view

El malware llama a GetModuleFileNameA para recuperar la ruta completa. Las figuras 69 y 70 muestran una instrucción y su resultado, se trata de un gran conjunto de imágenes que relacionan instrucción con salida resultante. Se extrae el nombre de la ruta mediante su interpretación en IDA mediante su respectiva instrucción. Esto se hace para obtener el nombre del fichero del módulo.

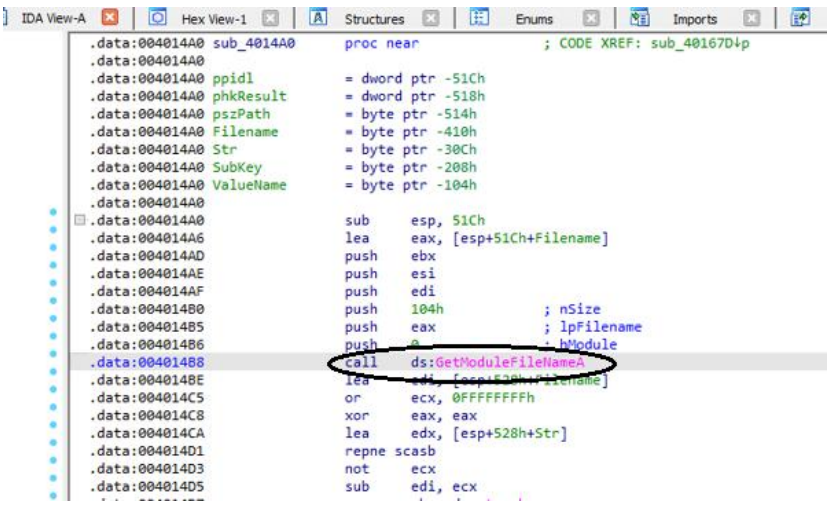


Figura 69: GetModuleFileNameA

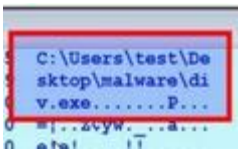


Figura 70: Ruta completa [15]

Se accede al apartado de visión en hexadecimal mediante una opción de la herramienta llamada Hex-View-1. La información está encriptada. La función de este apartado es ver de otra manera los datos a estudiar. Su respuesta en el visor hexadecimal de IDA es el siguiente (figura 71):

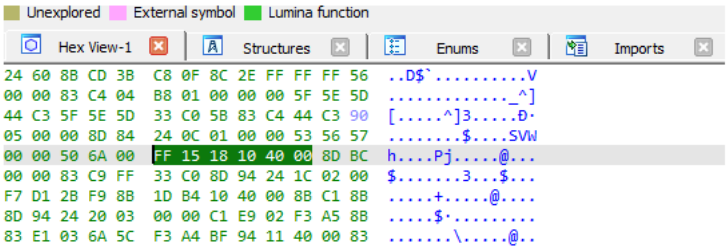


Figura 71: Hex View

El malware abre el identificador para ejecutar la clave de registro (figuras 72 y 73). La instrucción sirve para ejecutar (abrir) la clave del registro.

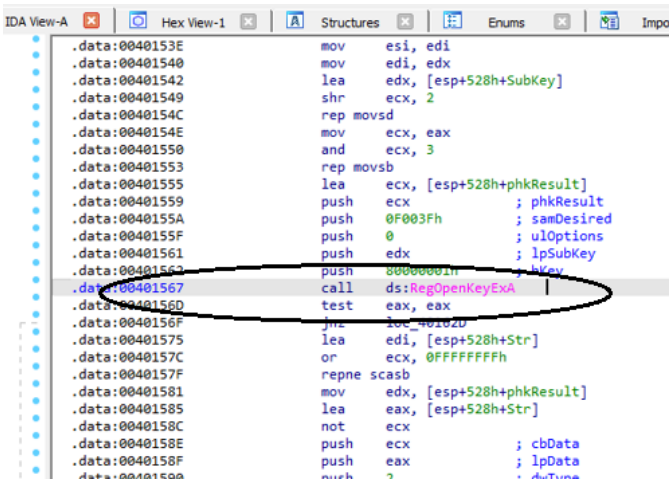


Figura 72: RegOpenKeyExA

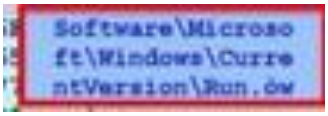


Figura 73: Identificador [15]

El malware establece el valor en la clave de registro de ejecución (figuras 74 y 75). Se introduce el valor malicioso. Es decir, se ejecuta el valor ubicado en la ruta descrita en pasos anteriores.

```
.data:0040158C          not     ecx
.data:0040158E          push    ecx             ; cbData
.data:0040158F          push    eax             ; lpData
.data:00401590          push    2               ; dwType
.data:00401592          lea     ecx, [esp+534h+ValueName]
.data:00401599          push    0               ; Reserved
.data:0040159B          push    ecx             ; lpValueName
.data:0040159C          push    edx             ; hKey
.data:0040159D          call    ds:RegSetValueExA
.data:004015A3          test    eax, eax
.data:004015A5          jz      loc_40162D
.data:004015AB          mov     ecx, 41h ; 'A'
.data:004015B0          xor     eax, eax
.data:004015B2          lea     edi, [esp+528h+pszPath]
```

Figura 74: RegSetValueExA

Figura 75: Clave de registro de ejecución [15]

El malware descifra la cadena de una URL (figura 76), esto es importante para observar cómo funciona el malware.

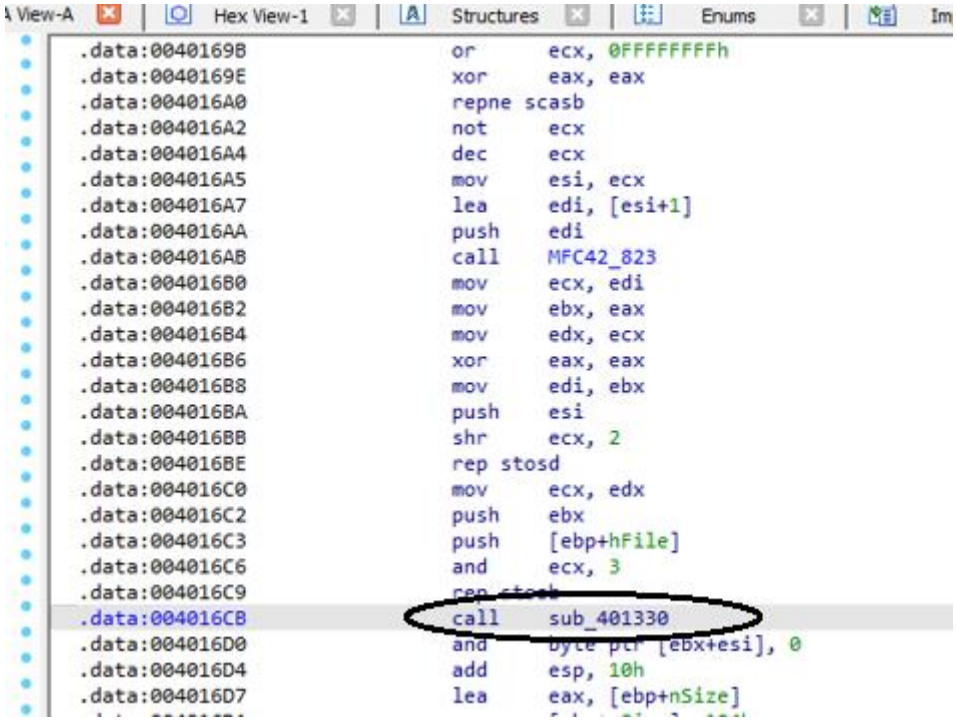


Figura 76: sub_401330

En vez del dato señalado, según el ejemplo del documento, este debería decrypt_func. Nos movemos a la dirección mostrada en la instrucción que contiene al “sub…” a ver qué información nos ofrece (figura 77).

```
.data:0040132D                  retn
.data:0040132D sub_401290       endp
.data:0040132D
.data:0040132D ; ---------------------------------------------------------------------------
.data:0040132E                  align 10h
.data:00401330
.data:00401330 ; ============== S U B R O U T I N E =======================================
.data:00401330
.data:00401330
.data:00401330 sub_401330       proc near               ; CODE XREF: sub_40168C+3F↓p
.data:00401330                                          ; sub_4017F3+118↓p
.data:00401330
.data:00401330 var_44           = byte ptr -44h
.data:00401330 var_43           = byte ptr -43h
.data:00401330 var_1D           = byte ptr -1Dh
.data:00401330 var_1C           = byte ptr -1Ch
.data:00401330 var_1B           = byte ptr -1Bh
.data:00401330 arg_0            = dword ptr  4
.data:00401330 arg_4            = dword ptr  8
.data:00401330 arg_8            = dword ptr  0Ch
```

Figura 77: En dirección paso anterior

Observamos que en dicha dirección no hay información relevante. Ahora es mostrado el resultado si el ejemplo habría sido el decrypt_func. Mostramos la figura que nos ofrece el ejemplo (figuras 78 y 79), en ellas se puede observar la cadena encriptada y luego la desencriptada, “extrayendo” la respectiva dirección HTTP:

```
0040190B call    decrypt_func
```

Figura 78: decrypt_func [15]

```
6k6GpmsqiHrgTD87
HrO\f30dgzZ7st2x
*uXqkz. Software
```

```
5 63 72 6F 77 6E 73  http://thecrowns
3 73 73 2F 61 62 6F  golf.org/css/abo
B AB AB AB AB AB FE  ut.htm
6 5C 84 5E 38 00 00  .....[µ\ä^8..
```

Figura 79: Cadena encriptada y desencriptada

El malware obtiene el hostname del sistema mediante la instrucción correspondiente (figuras 80 y 81)

```
.data:004016DA                  mov     [ebp+nSize], 104h
.data:004016E1                  push    eax             ; nSize
.data:004016E2                  lea     eax, [ebp+Buffer]
.data:004016E8                  push    eax             ; lpBuffer
.data:004016E9                  call    ds:GetComputerNameA
.data:004016EF                  push    9
.data:004016F1                  mov     esi, offset aMicrosoftInter ; "Microsoft In
.data:004016F6                  pop     ecx
.data:004016F7                  lea     edi, [ebp+szAgent]
.data:004016FD                  rep movsd
.data:004016FF                  push    38h ; '8'
```

Figura 80: GetComputerNameA

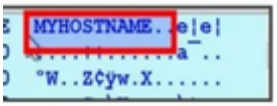


Figura 81: Obtención del hostname [15]

El malware concatena el nombre de host con una cadena codificada y lo usa como un agente de usuario relacionado con el navegador (figuras 82 y 83).

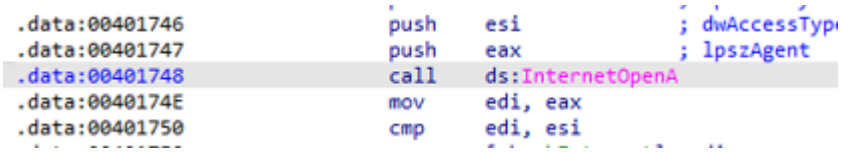


Figura 82: InternetOpenA

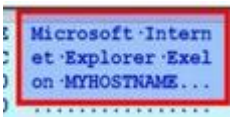


Figura 83: Concatenación [15]

El malware obtiene el contenido html mediante la llamada a la API InernetReadFile (figuras 84 y 85). Esta parte es muy importante, ya que el malware se extrae de dicho contenido html.

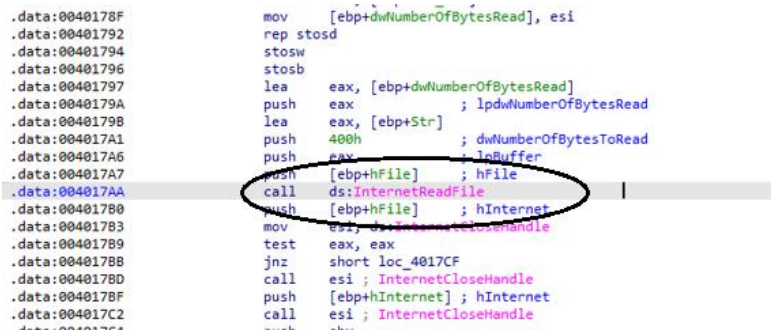


Figura 84: InternetReadFile

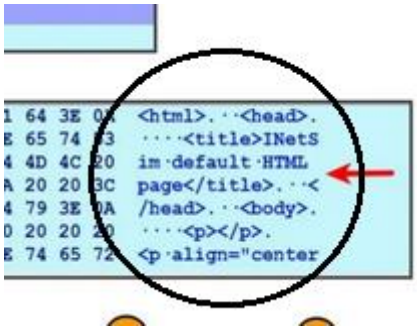


Figura 85: Contenido html [15]

El malware busca una cadena dentro de la etiqueta div en el contenido html (figuras 86 y 87), esto lo hace mediante unas instrucciones ubicadas en el malware que estamos estudiando.

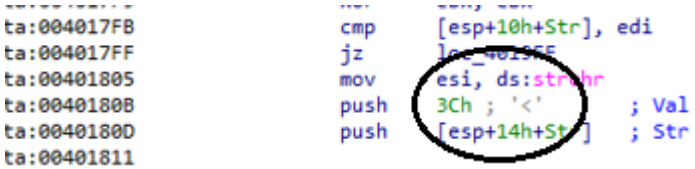


Figura 86: Búsqueda div 1

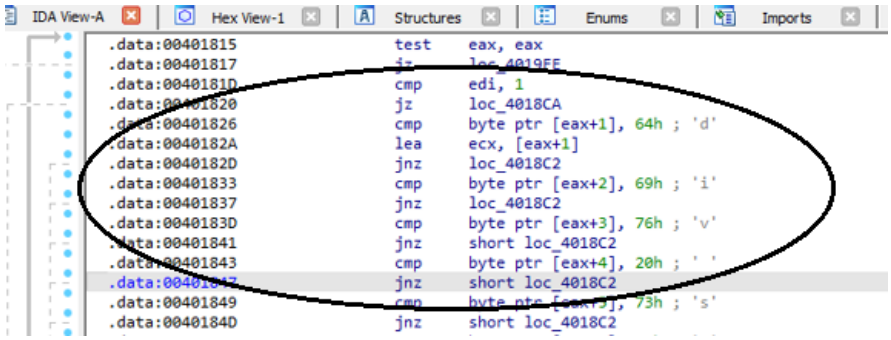


Figura 87: Búsqueda div 2

Se observa el contenido de div mediante un visor de Wireshark (ver la figura 88) usando InetSim. Dicho contenido muestra un archivo codificado en el div del HTML, esto servirá para poner a “dormir” el sistema.

Figura 88: Clave encriptada 1 [15]

Se desencripta el contenido del div tag. Las imágenes de la figura 89 muestran el desencriptado, es decir, de Ufmz_w… a Jabc01…

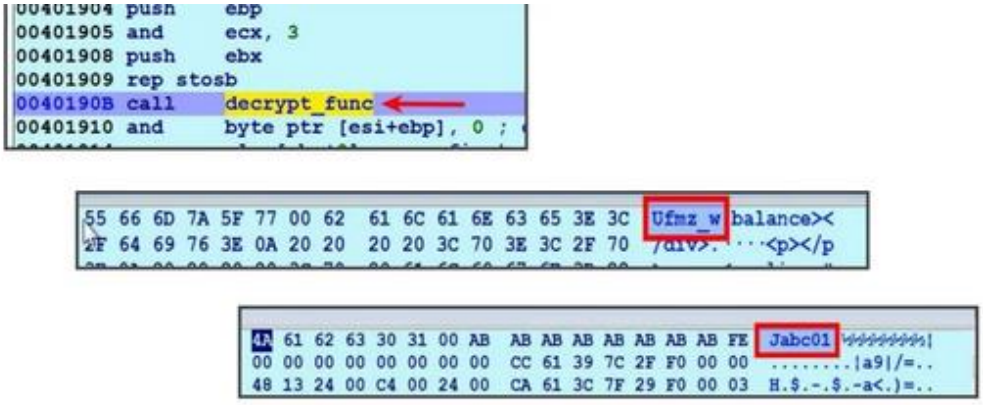


Figura 89: Desencriptado [15]

Si el primer carácter del contenido descifrado es 'J', el malware analiza argumentos adicionales en la cadena descifrada para especificar el intervalo de suspensión. El carácter "J" en la cadena descifrada es el comando que le indica al malware que se “duerma”. Las figuras 90 y 91 muestran las características que se deben de tener para poner a “dormir”.

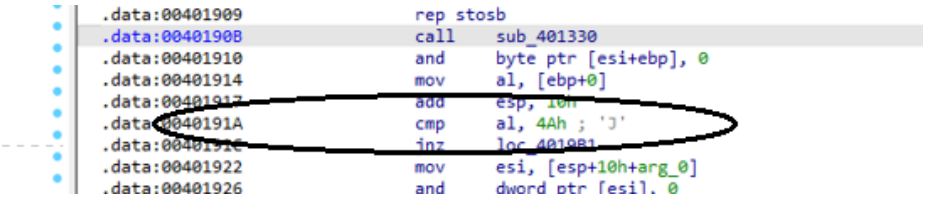


Figura 90: Comprobación J

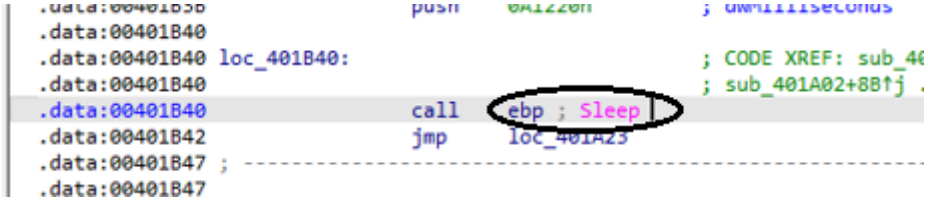


Figura 91: Dormir en caso J

Se estudia otro contenido del malware distinto al del ejemplo anterior (con otro contenido de InetSim). Esto lo hacemos para ver la variedad de “ataques” que pueden ejecutar el malware WEBC2-DIV. La figura 92 se puede ver otro archivo encriptado.

Figura 92: Clave encriptado 2 [15]

El malware extrae el contenido dentro de la etiqueta div y lo descifra en la cadena que se muestra a continuación en la figura 93. Es decir, se observa el desencriptado, en este caso ejecuta un programa en la ruta correspondiente. Importa mucho como es el formato de la ruta del archivo “http”, se observa, por ejemplo, los caracteres “Do” al principio de la cadena desencriptada.

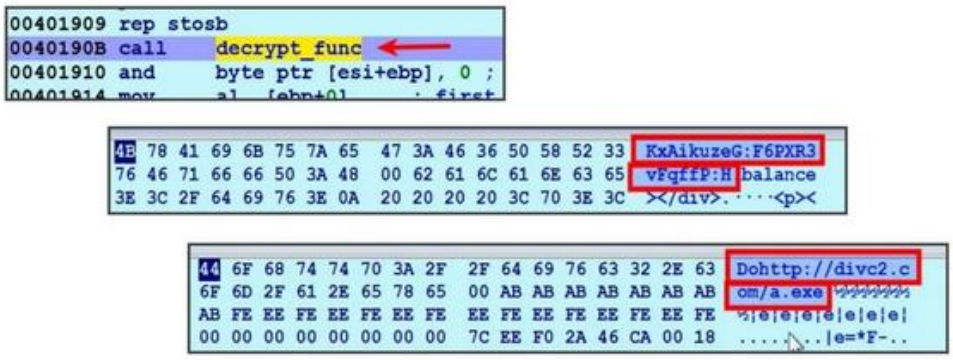


Figura 93: Obtenida la dirección a infectar

El malware comprueba si los dos primeros caracteres de la cadena del descifrado son "Do". Si se cumple esta conexión, se usa esto como el comando para descargar un ejecutable.

En la figura 94 se ve las condiciones para que se pueda ejecutar el programa mediante la decodificación anterior. En la figura podemos ver las instrucciones que tienen asociado el “Do” (en instrucciones separadas).

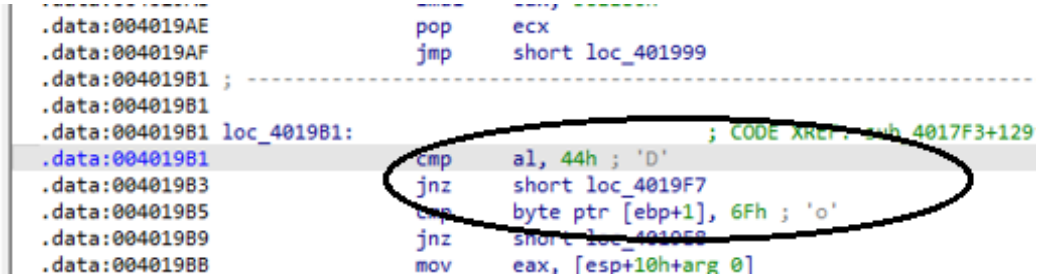


Figura 94: Comprobación Do en cadena

El malware descarga un ejecutable de la URL (que formaba parte de la cadena descifrada). El malware descarga el ejecutable y lo guarda en el directorio %temp%. En las figuras 95 y 96 se puede observar la instrucción de la descarga (http://divc2...) en la ruta correspondiente (C:\Users\test…) y su resultado.

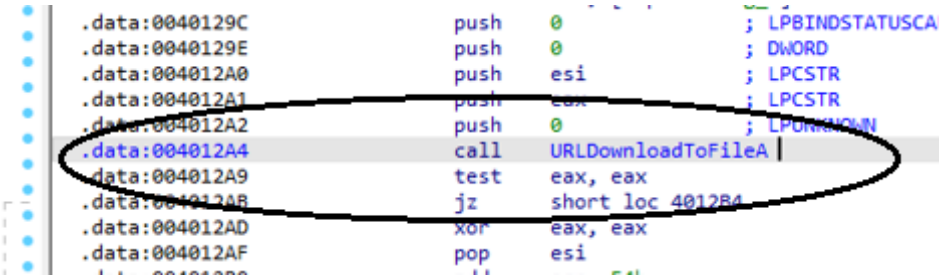


Figura 95: URLDownloadToFileA

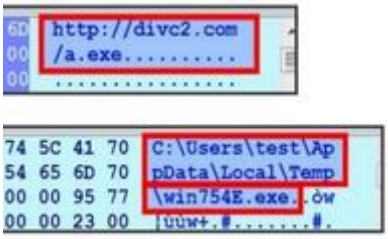


Figura 96: Cálculo de la página a infectar [15]

Una vez que se descarga el ejecutable, se ejecuta mediante la llamada a la API CreatedProcess(). Esto confirma que los caracteres "Do" en la cadena descifrada son el comando que le indica al malware que descargue y ejecute un archivo.

En las figuras 97 y 98 se muestra la ejecución y su resultado. A modo de conclusión dicho apartado es el último de todos:

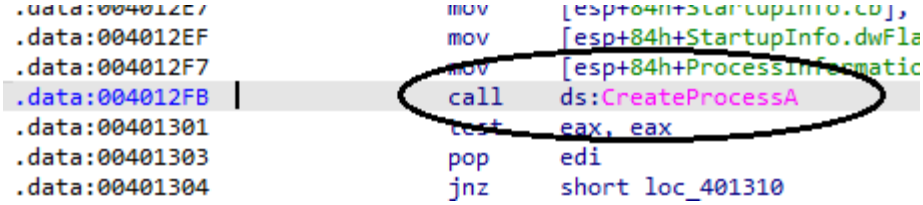


Figura 97: CreateProecessA

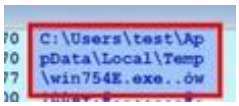


Figura 98: Eliminación de Do [15]

Estudiando el comportamiento de WEBC2-DIV, se observa el siguiente esquema (su arquitectura) y su explicación:

En cuanto al análisis de malware, existen dos técnicas principales para dicho análisis que son las más utilizadas. El método de estudio se basó en el análisis estático y el análisis dinámico. El análisis estático es un método de análisis de malware que se realiza sin ejecutar el malware, por lo que el análisis con este método es mucho más seguro que con el método de análisis dinámico. Dicha información proviene de [19] (información y figura).

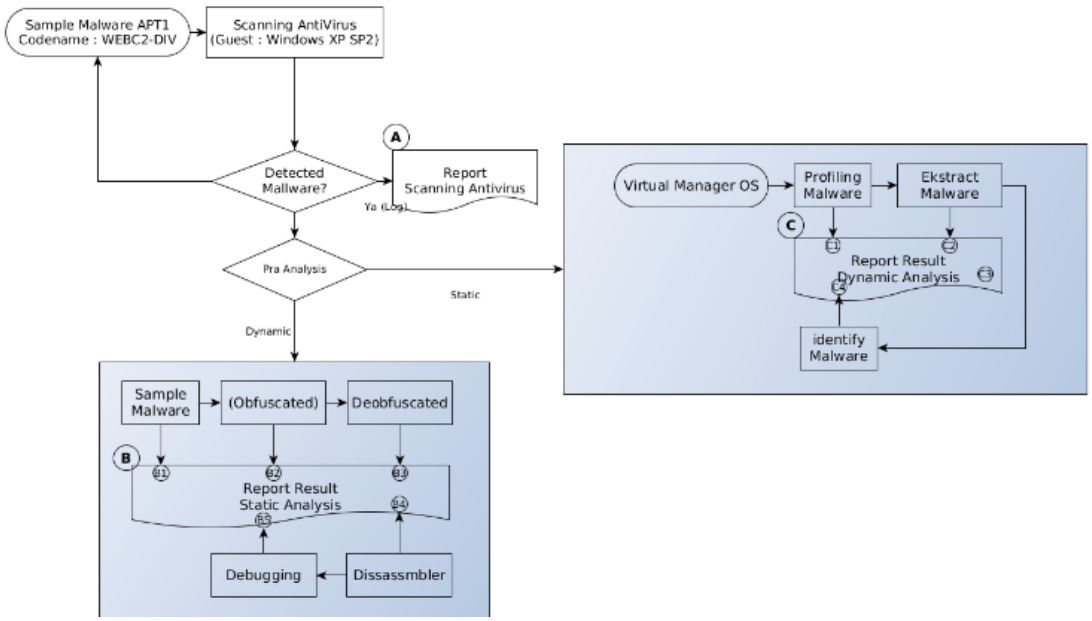


Figura 99: Ejemplo de análisis del malware WEBC2-DIV en funcionamiento [19]

Posteriormente se detallará como se ha usado otra herramienta cuya función es muy parecida a la explicada en este apartado. Esto servirá para contrastar la función de ambas herramientas.

5.4 Cutter

En primer lugar, se mostrará información general sobre el archivo malicioso div.exe (en nuestro caso WEBC2-DIV_sample_1E5EC6C06E4F6BB958DCBB9FC636009D) que utiliza el software Cutter. Se ejecuta el cortador y se inserta WEBC2-DIV_sample_1E5EC6C06E4F6BB958DCBB9FC636009D, obtenemos información sobre hash y biblioteca, como podemos observar en [19].

Este análisis se hace para ver otra manera de trabajar con la información de un mismo malware. Esta herramienta muestra una versión reducida de las funciones descritas en IDA. La información vista en Cutter es distinta que la empleada en IDA, teniendo ambas (IDA y Cutter) información en hexadecimal y en código ensamblador, solo que en Cutter tenemos menos información a estudiar. Tanto IDA como Cutter están basados en interpretar información audiovisual y el estudio de la ejecución de instrucciones en lenguaje ensamblador. Las figuras 100 y 101 muestran la respectiva información del malware en Cutter.

Figura 100: Info muestra 1

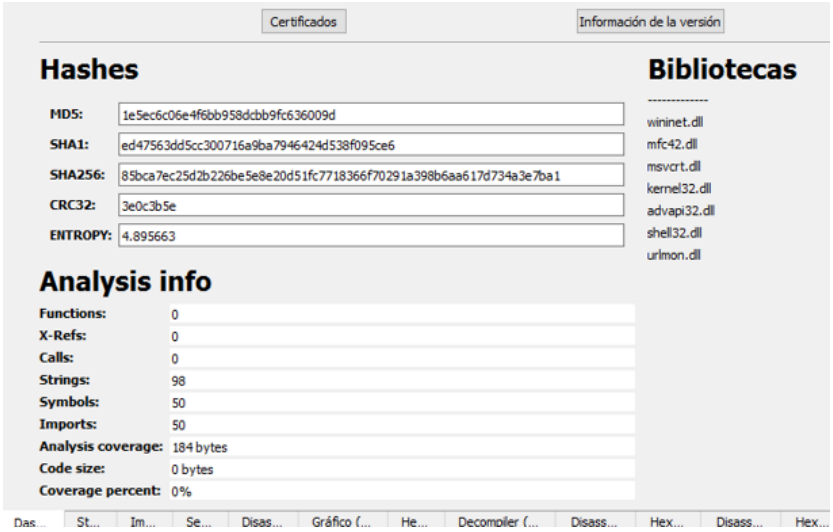


Figura 101: info muestra 2

Se va a desmontar el cuerpo del malware para encontrar el comportamiento del malware principal y hacer el valor de “plantación” en la dirección “regedit” de Windows HKEY_CURRENT_USER/Software/Microsoft/Windows/Versión actual/Ejecutar. Se puede ver en bloques de color. En la figura 102 podemos observar la ruta en la que se produce el ataque, esta se muestra de forma explícita, sin encriptar.

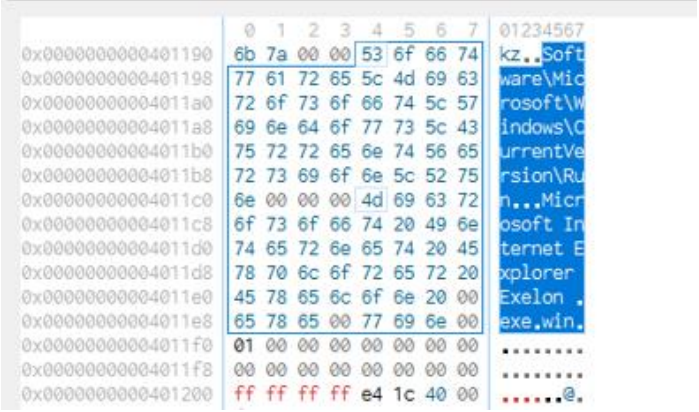


Figura 102: Dirección comportamiento del malware

Desensamblar el estado no solo se detiene en esto, el malware obtuvo el cifrado que contenía un mensaje importante y puede que la computadora infectada sea dañina. Las figuras 103 y 104 muestra el malware cifrado, tanto en hexadecimal como en código ensamblador. En el código ensamblador se puede ver tanto la cadena encriptada como las rutas “objetivo” del ataque que fueron mostradas en la figura anterior.

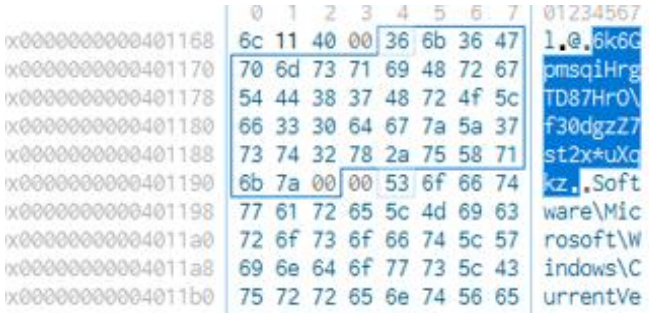


Figura 103: Información desencriptada

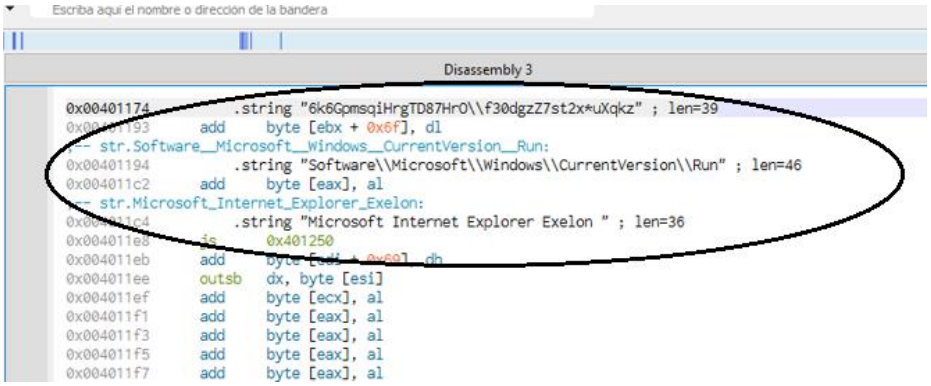


Figura 104: Encriptado mostrado en Desensamblador

La figura 104 muestra información adicional relacionada con direcciones de memoria.

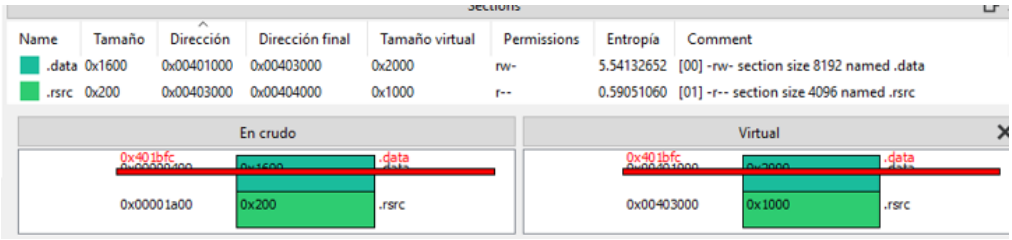


Figura 105: Información adicional

5.5 Poison Ivy

Una explicación rápida sobre el funcionamiento de este troyano: Poison Ivy se encuentra dentro de la categoría de troyanos RAT (Remote Access Trojan) que utiliza un tipo de conexión llamado «conexión inversa», es decir, este troyano infecta a un usuario y el pirata informático no necesita conocer la IP de la víctima, sino que el troyano ya está configurado con la IP del pirata y será este el que se conecte a dicha IP en busca del servidor con el que controlarlo. Poison Ivy es toda una suite de control remoto. Permite tomar capturas de pantalla, activar la webcam, robar las pulsaciones del teclado, acceso a los archivos de la víctima, etc. Prácticamente cualquier ataque se puede llevar a cabo con Poison Ivy.

6 Resultados

En cuanto al primer objetivo:

> Analizar unas determinadas direcciones IP relacionadas con APT1. Si analizamos el "como" conseguimos esas direcciones, serán extraídas de varios documentos, dos denominados JIB (Joint Indicator Bulletins, contenidos en [21] y [22]) y otro mediante un documento apéndice del informe Mandiant, dicho informe es el más extenso en cuanto a la descripción del ataque APT1, pero, apenas tiene contenido "práctico". Atendiendo al "para que"**,** se obtiene mucha información de las direcciones IP (marcas, puertos, números de servicio…), luego se obtienen otras direcciones IP para su análisis según su ASN. Todo esto puede ser útil para entender las características de dichas direcciones IP, lo que nos da ventaja a la hora de actuar en caso de ataque.

Antes de empezar a comentar la parte práctica y, como introducción a los objetivos, he explicado lo que es una APT y como su estudio está relacionado con análisis forense, en concreto he tratado los siguientes temas relacionados como introducción teórica: el concepto sobre APT y su impacto en la ciberseguridad, análisis forense y APT1s. Después se hace más énfasis en aspectos profundos y técnicos del trabajo, como por ejemplo los objetivos del trabajo, la planificación (inicial y temporal) o la organización. Los siguientes apartados son: una aproximación teórica a APT1 (volvemos a explicar teoría), luego volvemos al tema técnico: metodología, que se divide en fases y en el laboratorio, con una tabla que explica las ventajas y desventajas de las herramientas.

Introducción:

El concepto sobre APT y su impacto en la ciberseguridad: se hace una idea de la peligrosidad de un APT, se divide el ataque en etapas.

Análisis forense: se define lo que es análisis forense y se da una pincelada inicial de lo que se va a tratar en el trabajo.

Objetivos del trabajo: se dividen en tres y se especifican.

Planificación Inicial: Se dividen en etapas las que forman el proyecto.

Planificación temporal: Lo que ha durado el proyecto y sus fechas.

Presupuesto y estimación de costes: Lo que ha costado en general el proyecto.

La organización: Descripción paso por paso del proyecto.

Una aproximación teórica a APT1: Donde, como, que es, a quién afecta…

Metodología:

Fases del trabajo: Por qué se ha seleccionado APT1, fases ordenadas en las que se divide el trabajo.

Laboratorio forense creado para realizar el análisis: SilK, comandos de Linux, Internet Census, IDA, Process Hacker, Cutter, Wireshark, InetSim, Posion Ivy.

Tabla comparativa: ventajas y desventajas de las herramientas.

Estos resultados han sido fáciles de conseguir, puesto que solo consistía en buscar la información en internet. Dicha información derivada del resultado ha sido fácil de encontrar, puesto que ha sido extraído del informe más popular que trata sobre el ataque, el mostrado en [13], excepto las partes más técnicas relacionadas con el trabajo, esas han requerido más esfuerzo en términos de qué y cómo organizar la información del trabajo. En cuanto al resultado de lo contenido en dicho informe es que trata en detalle el ataque de forma teórica y descriptiva. El resultado de realizar la parte teórica ha sido fácil puesto que no hay demasiadas imágenes con ventanas de algún software especializado, sino más bien gráficos sobre estadísticas, tablas e imágenes de datos concretos.

El primer resultado derivado de la parte práctica ha sido obtener las direcciones IP de los JIB (Joint Indicator Bulletins).

Después, se tiene unas FQDNs (almacenamiento Mandiant) que fueron convertidas de nombres de dominio a direcciones IP mediante una página web específica. Para saber en detalle los resultados derivados obtenidos de dichas FQDNs se procede a su definición:

El término Fully Qualified Domain Name (FQDN) se refiere a la dirección completa y única necesaria para tener presencia en Internet. Está formada por el nombre de host y el de dominio y se usa para localizar hosts específicos en Internet y acceder a ellos mediante la resolución de nombres.

Estos resultados, tanto el almacenamiento JIB y como el almacenamiento Mandiant han sido realizados de forma exitosa, debido a que se han encontrado todos los datos, sin embargo, ha sido difícil conseguir el resultado de encontrar el material en internet: FQDNs pertenece a un conjunto de información más amplio llamado appendix y, dentro de este, Digital Appendices, todo relacionado con el informe Mandiant, encontrar el archivo FQDNs ha sido sencillo (una vez descargado las carpetas pertinentes, más complejo de encontrar), puesto que es solo desplazarse entre carpetas.

En los JIB, para obtener resultados sobre su utilidad, las direcciones IP se encuentran en la parte inferior del documento, llamado Technical Data (Las direcciones IP), después del apartado Document FAQ, este contiene:

Observe que este documento está etiquetado como TLP: VERDE. ¿Puedo distribuir esto a otras personas? Los destinatarios pueden compartir TLP: información VERDE con organizaciones asociadas dentro de su sector o comunidad, pero no a través de canales de acceso público. Comuníquese con US-CERT para consultas de distribución específicas.
¿Puedo editar este documento para incluir información adicional? Este documento no debe ser editado, cambiado o modificado de ninguna manera por los destinatarios. Todos los comentarios o preguntas relacionados con este documento deben dirigirse al Centro de Operaciones de Seguridad de US-CERT al 1-888-282-0870 o soc@us-cert.gov.

Un apartado que no ha sido usado de los documentos JIB han sido “Domain Name Awareness List” (Números de dominio), porque no era necesario el resultado de su uso en dicho trabajo. Lo mismo pasa con el apartado Malware “Indicator Awareness List” y su subapartado “MD5 Checksum” (242946ed32dc3749e5b4f7827b905e5e, por ejemplo). Al principio, toda esta información puede confundir al trabajador, puesto que las direcciones IP se encuentran abajo del todo, mostrando anteriormente mucho texto adicional.
También hay otra información de contacto:

Contact NCCIC/US-CERT:

US-CERT está interesado en cualquier información extra que su organización pueda compartir con respecto a esta u otra actividad de iguales características. Para cualquier pregunta o comentario relacionado con este informe, comuníquese con US-CERT:

(UNCLASS) Phone: +1-703-235-8832
(UNCLASS) Email: soc@us-cert.gov
US-CERT's PGP key may be downloaded at us-cert.gov/contact
(SIPRNET) Email: us-cert@dhs.sgov.gov
(JWICS) Email: us-cert@dhs.ic.gov

Contact FBI:

También puede comunicarse con el FBI si tiene alguna pregunta relacionada con este JIB

Email: cywatch@ic.fbi.gov
Voice: +1-855-292-3937

Otro resultado, ya entrando en la parte más práctica y entrando en más detalle, ha sido cambiar las direcciones están en un formato poco adecuado (por ejemplo 107[.]6[.]38[.]55). Mediante una herramienta (descrita anteriormente en la memoria, el uso de Reemplazar por…) cambiamos los [.] por . (punto). Esto se puede considerar la fase previa al primer objetivo.

El siguiente resultado, una vez que ya sabemos dónde están las direcciones IP, el paso de mover las direcciones IP a varios de los archivos (*Ij.list, Im.list, Ij+Im.list*…) ha sido satisfactoria debido a que solo ha sido copiar y pegar. El resultado del paso de cambiar de formato de las direcciones IP de los .list al formato indicado en el guion (^127\.0\.0\.1[[:blank:]]), para crear el *Ij_reg.list, Ij+Im_reg.list*…, ha sido costoso, puesto que ha habido que hacer los cambios manualmente. Después, el resultado del paso de usar herramientas SilK, para crear el Ij.set derivado del archivo Ij.list, ha sido realizado de forma exitosa y ha resultado sencilla debido a la facilidad de sintaxis de la instrucción en la terminal de comandos Ubuntu.

Solo se ha convertido en .set el *Ij.set* puesto que se utiliza en apartados anteriores, a diferencia del *Im.list.*

La fase explicada a continuación se corresponde con la primera parte del objetivo: analizar unas determinadas direcciones IP relacionadas con APT1. Cuyo primer resultado ha sido el intento de obtener las características de las direcciones IP de JIB y Mandiant, (almacenadas anteriormente en sus archivos correspondientes) mediante un filtrado con direcciones IP extraídas de un apartado de Internet Census llamado fingerprint, el cual da información sobre muchas direcciones IP. Enlazando con el primer objetivo, este se encuentra especificado en la sección "marcas". Dicha información ha sido difícil de encontrar debido a que el documento donde estaban (Internet Census) era muy extenso y "estaba muy escondida". Para entender mejor los resultados derivados de Internet Census, se procede a su explicación:

En internet Census tenemos la siguiente información: Información, "prueba de concepto", "diseño e implementación"; esta última se divide en "Be Nice", "plataformas

como objetivo". "C&C menor infraestructura", "nodos medios", "coordinación de scaneo", "cadena de herramientas". El siguiente punto es llamado "desafíos de implementación". El siguiente punto es métodos de escaneo, que se divide en: Ping ICMP, DNS reverso, Nmap,"service probe (sondas de servicio)", traceroute (rastreo de ruta0). El siguiente punto es el análisis, que se divide en: curvas de Hilbert, mapas del mundo, (otra vez) DNS reverso, (otra vez) service probe, "números", "ruido". Para acabar están los apartados, conclusión, trivialiades, quién y por qué.

Una vez encontrada la información, el resultado derivado de dicha información mostrada ha sido fácilmente interpretado debido a que los archivos estaban numerados en orden numérico (1, 2, 3…) y, aunque estaba comprimida, le herramienta de compresión (zpaq) ha dado un resultado satisfactorio debido a que sido fácil de instalar y ejecutar en un terminal Ubuntu. Los números reflejan el primer número de las direcciones IP contenidas, por ejemplo, las IPs contenidas en "1", empezarán en 1.x.x.x. El resultado de interpretar la información contenida en el archivo fingerprint ha sido complicado debido a la cantidad de información críptica: en el campo "Match" existe un número el cuál no se sabe su utilidad. En el campo "fingerprint" ubicado justo delante de "Match" existen numerosos caracteres de carácter críptico, con palabras clave como SCAN y SEQ y muchos caracteres relacionados con ellos, por ejemplo: V=6.01%E=4%D=9 / 4%OT...En SCAN y SP=E4%GCD=1%ISR=10D% en SEQ...Entre dichos datos se puede extraer la siguiente información relacionada con el sistema operativo: mipsel-openwrt-linux-gnu. El resultado de usar el comando Ubuntu grep para extraer de *Ij+Im_reg.list* aquellos valores almacenados en 1 o 2 o 3…no ha sido realizado de forma correcta debido a un fallo en el "grep" (o derivado). Para ver otra versión de visualización de resultados se ha creado una tabla en Excel mediante un código escrito en Python.

Atendiendo a otro resultado obtenido ha sido obtener los puertos abiertos de una serie de direcciones IP facilitadas por Internet Census, este apartado se llama synscan y se enlaza con el objetivo cuando nos referimos a "puertos" en el paréntesis de "para que". El resultado de encontrar la información derivada ha sido igual que en el apartado fingerprint, ya que tiene el mismo formato (1, 2…). Sigue existiendo el orden de forma similar al anterior apartado de, por ejemplo: archivo número 1, comienzo de número de direcciones por 1 (1.x.x.x). El resultado de interpretar la información contenida ha sido más sencillo que en el apartado fingerprint debido a que las direcciones IP mostraban (en synscan) información de puertos abiertos, cerrados, filtrados…En vez de información con caracteres crípticos (excepto un número de largos caracteres después de la dirección IP, que no he sabido interpretar). Los campos son los siguientes: un campo en el que se muestra un "no-response", el tipo de protocolo, (tcp principalmente), el tipo de trama (syn-ack) y, por último, los puertos asignados. Toda esta información es fácil de interpretar debido a que está bien relacionada entre ella, y no se necesitan gran cantidad de conocimientos sobre redes, pero si tener algunas nociones básicas. Mediante un sencillo filtrado mediante grep y recortando la información mediante cut para obtener solo puertos abiertos se ha obtenido la información requerida causando un resultado satisfactorio. Cut no ha sido usado en el apartado anterior, no era necesario, pero en este apartado si se ha usado debido a que la forma de discriminar los datos ha sido más concreta y concisa. Esta vez el uso del grep ha sido exitoso, a diferencia del apartado anterior. Enlazando con este concepto, la referencia a la información que almacena todos los datos ya no ha sido un número, sino un asterisco (*).

El siguiente resultado obtenido ha sido el de extraer unos datos de direcciones IP facilitadas, de nuevo, por Internet Census, dicha extracción consiste en quedarnos solo con las direcciones IP con un número de servicio en concreto, frente a muchos servicios

asignados. Esta vez el resultado de interpretar la información no se nos muestra como en apartados anteriores, esta vez los campos son los siguientes: tcp_GetRequest_1 (o 2, o 3…).zpaq. Para obtener un resultado correcto con zpaq se usa la siguiente instrucción: zpaq x “ejemplo”.zpaq, de esta manera se extraen los archivos en la carpeta de trabajo en la que se esté ubicada.

Como análisis del resultado, dicho apartado tiene mucho menos contenido que los anteriores, es fácil interpretar tanto la dirección IP mostrada como su número de servicio, no siendo así un número de muchas cifras que se encuentra en medio de ambos campos. Este apartado se llama service probe y se corresponde con el tercer punto del objetivo (“números de servicio”). En nuestro caso se extraerá las direcciones IP de número de servicio 4 (destacar que el archivo con el cuál se trabajará tiene muchos 5 como número de servicio), esto ha resultado correcto debido a que las instrucciones a realizar tenían una sintaxis clara, además de que se repiten comandos de otros apartados, como grep, cut y rwsetbuild, esta parte ha sido correcta. Ambos comandos se han usado en apartados anteriores, por lo que su función es clara. Sin embargo, también se especifica que ha hay que realizar una intersección entre los datos de servicio 4 y el *Ij.set,* mediante una herramienta que tiene una sintaxis un poco más compleja que las anteriores llamadas rwsettool, el resultado de esta operación de intersección no ha sido realizada de forma correcta.

Si seguimos con la consecución de otro resultado, esta consiste en el análisis de una serie de direcciones IP mediante su ASN (Autonomous System Number). Esto se ajusta al cuarto punto del objetivo (“luego se obtienen otras direcciones IP para su análisis según su ASN.”) para ello primero se ha convertido el archivo IP_ASN, que contiene la dupla dirección IP-ASN (por ejemplo: 192.137.127.0, correspondiente a España, con un ASN asignado de 33) de un país concreto a IP_ASN.pmap mediante un comando SilK llamado rwpmapbuild, esta parte ha sido fácil, sin embargo, es desconocido el contenido del archivo .pmap. En cuanto al resultado asignado a este apartado, difiere bastante de los anteriores en cuanto a la información a tratar debido a que esta vez no usaremos Internet Census para descargar los archivos. Ha sido encontrada una página web que permite calcular la dupla dirección IP-ASN de un determinado país. Encontrar dicha página ha sido difícil, como resultado de su análisis se puede observar que tiene mucha información contenida, por lo que los resultados varían mucho dependiendo de que país elegimos, que son muchos. En mi caso he elegido los países según el criterio de importancia, descartando países pequeños, por ejemplo, Andorra (como explico en un ejemplo anteriormente). El uso de esta página web ha sido muy sencillo ya que todos los datos se muestran de “manera directa” y con una interfaz gráfica “amigable”. Una vez almacenados nuestros ejemplos de la dupla para alcanzar algún resultado concreto, se usará un comando SilK llamado rwtuc que relaciona el IP_ASN.pmap con el *Ij+Im.list* creado en apartados anteriores. Rwtuc es un comando nuevo no usado en apartados anteriores. Este resultado consta de una información muy críptica, en pantalla se muestra src-asn | record y debajo de lo mostrado unknown | 1253.

El siguiente resultado ha sido el intento, sin éxito, de realizar los apartados Routing Data, Country Code y Open resolvers. No se ha logrado obtener los archivos asociados a dichos apartados.

Se crea un archivo con 100000 direcciones IP de un archivo más grande mediante el comando SilK rwsettool. Esto se hace para examinar cierta información de interés, en vez de toda la información.

En cuanto al segundo objetivo:

Estudio del malware WEBC2-DIV. En cuanto al **"**como**"** extraemos ese malware, este fue obtenido de una página web que almacenaba un montón de malware en formato comprimido, para descomprimirlo tuve que pedir la contraseña al "dueño" de la página. En cuanto al **"**para que**"**, dicho estudio permite esclarecer cómo funciona el malware desde un punto de vista detallado con programas como: Process Hacker, InetSim, Wireshark, lenguaje ensamblador mediante IDA y Cutter...

Este objetivo es completamente diferente al anterior, tanto en los programas usados como en la metodología usada, obteniendo resultados derivados del uso de las herramientas (software) que se va usando en cada apartado.

Atendiendo a esto, el siguiente resultado ha sido el análisis del malware WEBC2-DIV. Ha habido que obtener la muestra, el resultado de este apartado ha sido logrado mediante una búsqueda en internet y un contacto con el propietario de ese "blog" para obtener la muestra de WEBC2-DIV. Observar que el nombre de la muestra comprimida tiene muchos caracteres, eso no nos afecta.

Si nos afecta, sin embargo, saber ejecutar el malware para poder analizarlo con un programa llamado Process Hacker, ubicado en la máquina Windows. El resultado de interpretar dicho programa ha sido satisfactorio, pues es casi similar al editor de registro. Esto se ajusta al primer punto del segundo objetivo (cómo funciona el malware desde un punto de vista detallado con programas como: Process Hacker)...

Dentro de Process Hacker podemos encontrar procesos como svchost.exe (el más común), vm3dservice.exe, MsMpEng.exe, wlmps.exe…

El resultado de saber cómo analizar la muestra ha sido logrado, pero también difícil, puesto que no era intuitivo. Este resultado se lograba simplemente se lograba mediante un cambio en el nombre a .exe en la parte final. El resultado de saber si la muestra está preparada para su análisis es dar "doble click" en ella y fijarse que se encuentra en el Process Hacker, dicha muestra del malware se encontraba en el escritorio cuando se ejecutó. Se puede observar que se encuentra casi al final del programa Process Hacker.

El resultado de comunicar la máquina virtual Ubuntu con la máquina virtual Windows no ha sido sencillo, pues esta vez intervienen comandos de Ubuntu más complejos, por ejemplo, el uso de nano para editar un fichero, cp para copiar la información de un archivo a otro, modificar campos en un archivo de texto o saber instalar un programa mediante sudo apt-get install... El resultado de saber la dirección IP asignada a la máquina virtual también ha requerido de ciertos conocimientos relacionados con máquinas virtuales, para ello se ha accedido a las opciones de la parte superior derecha de la pantalla (se trabaja en VMware como entorno de máquinas virtuales), seleccionar "Wired connected", "Settings" y pulsar en la rueda delante de "Connected". Para hallar resultados posteriores ha habido que instalar y configurar un "programa" llamado InetSim, para lograr eso ha habido que usar los comandos de Ubuntu que defino anteriormente (nano, cp…), haciendo hincapié en los cambios que se hacen dentro de un fichero llamado inetsim.conf, en concreto, ponemos la dirección IP de Ubuntu en el campo "service_bind_address" y otro con el" dns_default_ip", desconozco por qué hay que cambiar esos dos datos en concreto para que InetSim funcione. Hay que saber cómo matar un proceso, para ello ha habido que saber qué proceso matar, alcanzar dicho resultado no ha sido fácil debido a que la información de "matar" estaba en un fichero difícil de conocer, de nombre inetsim.pid. La configuración de InetSim ha sido satisfactoria, mostrándonos por pantalla datos como la dirección IP, datos que parecen

aplicaciones, como están dichas aplicaciones (started) y su PID, el resultado de saber que significa cada cosa también es complejo.

Una vez logrado el resultado de instalación y configuración de InetSim pasamos a evaluar los resultados derivados de la herramienta Wireshark. Esto se enlaza con el objetivo del "para que" en el primer apartado (de nombre Wireshark).

En Wireshark se utiliza un sistema de análisis de tramas de red, dichas tramas están formadas por: Nº, tiempo, origen, destino, protocolo, longitud e información.

Mediante ese programa se pueden analizar varios comandos de utilidad. Se sabe que el resultado de evaluar las tramas es correcto si, al hacer doble click en el malware, nos aparecen unas tramas (con el Wireshark abierto) que, al compararlas con las mostradas en el guion que he seguido para hacer el trabajo, coinciden. El detalle de conocer como es el paso del "doble-click" del malware al análisis de Wireshark de las tramas relacionadas con dicho programa lo desconozco. Afortunadamente, el resultado es satisfactorio. Para seguir calculando resultados, es imprescindible conocer la herramienta de forma básica, para examinar las tramas en más detalle, en este caso concreto, encontrar el comando TCP Stream mediante el correspondiente menú, esto es, click derecho en la trama, opción "Follow" y después "TCP Stream". El contenido de las primeras tramas es el siguiente: en la primera es "standard query 0x30ed A thecrownsgolf.org" y la segunda "standard query response 0x30ed a A thecrownsgolf.org A 192.169…" En las tramas de thecrownsgolf vemos que se produce un resultado de tipo DNS, con dirección origen la dirección IP de la máquina Windows y dirección destino la dirección IP de la máquina Ubuntu y viceversa. El resultado mostrado en el trabajo es el siguiente: analizamos las dos tramas siguientes a la de "crownsgolf ", analizamos los resultados obtenidos en ellas (que al parecer es el mismo en ambas tramas): métodos de petición de tipo GET, la aplicación cliente (user-agent), que muestra el navegador y el nombre del servidor, el host al que se envía la petición (thecrownsgolf.org), un texto que nos muestra que no hay caché, una respuesta de tipo 200 OK, la longitud del mensaje (258), una fecha, (11 de Marzo de 2022 14:13:11 GMT), un apartado llamado conexión (close) que significa que la sesión será cerrada después de la respuesta, un apartado llamado Content-Type (text/html) que nos dice como es el contenido del texto enviado, un apartado dedicado al servidor al cual nos comunicamos (InetSim HTTP Server) y texto relacionado con el contenido de una página HTML, donde viene reflejado que, como servidor de destino, estamos usando InetSim.

Para seguir calculando resultados derivados de WEBC2-DIV se usará el programa IDA. Enlazando con el objetivo, se define la parte del "para que" en la sección "lenguaje ensamblador mediante IDA y Cutter". Para manejarse en detalle con esta herramienta hay que tener conocimientos de lenguaje ensamblador. Sin embargo, para el trabajo que nos ocupa, con seguir los pasos de un guion y teniendo conocimientos mínimos de programación se puede desarrollar este apartado sin grandes complicaciones. Lo primero de todo, antes de calcular resultados prácticos, tenemos que saber poner la interfaz gráfica del programa en modo texto, este resultado no ha sido complicado de alcanzar, solo hay que saber manejarse por los menús de forma básica, muy parecido a programas similares. Se consigue mediante click derecho en la pantalla y seleccionando "Text-view". Después, otro resultado se deriva en saber cómo analizar la Información mediante formato hexadecimal. En el trabajo no se ha hecho hincapié en la dirección de las instrucciones. Vamos analizando los resultados fijándonos en la dupla instrucción derivada-resultado. Como aspecto a señalar, las instrucciones indicadas aparecen en color rosa. Se ha conseguido interpretar:

El comando GetModuleFileNameA,- cálculo de ruta completa. Su visión en formato hexadecimal es muy críptica.

RegOpenKeyExA-ejecutar clave de regsitro.

Sub_401330-instrucción errónea, debería ser decrypt_func- descifrar cadena de url.

GetComputerNameA-obtener hostname.

InternetOpenA-Concatena un nombre de host con una cadena codificada y lo usa como un agente de usuario relacionado con el navegador.

InternetReadFile-Extracción de malware de html.

Las correspondientes con <div dispuestas en instrucciones en vertical en distintas direcciones con valores asociados de “byte pr, 64h…”

Otra vez decrypt_func-desencripta el contenido en <div

Análisis de la instrucción de J-pone el sistema a dormir si se dan las condiciones.

Las correspondientes con Do dispuestas en vertical en distintas instrucciones con valores asociados al, 44h…

URLDownloadToFileA-Instrucción de descarga y resultado.

CreateProcessA-Ejecución del malware y resultado en la ruta correspondiente.

Otro resultado independiente de cualquier programa es como es la arquitectura de WEBC2-DIV, siendo el resultado de su comprensión general sencilla y el detalle más complejo. Mediante “cajas” se definen los diferentes procesos que forman el malware. Como por ejemplo escaneo de virus, extracción de malware, desensamblamiento…

Para calcular otra manera de obtener resultados se ha usado la herramienta Cutter. Enlazando con los objetivos, este forma parte del “para que”, sección “lenguaje ensamblador mediante IDA y Cutter”, igual que en el apartado anterior Los resultados obtenidos aquí, a pesar de que parte de ellos se muestran también en lenguaje ensamblador, difieren bastante de los obtenidos en IDA. Se obtendrá un Resumen con información de:

Archivos: C:\Users\Adminstrador…

Formato: pe

Bits: 32,

…

//

Hashes (MD5, SHA-1…)

//

Analysis Info:

Strings: 98

Symbols: 50

Imports: 50

…

En este programa se hace mucho hincapié en resultados basados en formato hexadecimal, más que en el apartado de uso de IDA. En dicho formato hexadecimal se analizará la ruta en la cual se produce el ataque (Software/Microsoft…) y el malware cifrado (6k6g…). Destacar el navegador usado, que es el mismo que el campo User-Agent (Internet Explorer) del análisis de tramas de Wireshark. En cuanto a los resultados de la parte de lenguaje ensamblador de Cutter, se observa la misma información que la descrita en el formato hexadecimal, las siguientes cadenas: Codificada, sin codificar y navegador, éstas están delante de un texto llamado .string y muy juntas unas de otras. En las dos primeras cadenas hay un “add byte[…], dl o al, dependiendo de la instrucción.

El resultado derivado de la última figura es complejo de interpretar, parece una gestión de memoria de los procesos.

En cuanto al tercer objetivo:

> Descripción superficial del malware Poison Ivy. Para ello se definirá lo que es una RAT y el manejo de dicho malware.

Los resultados de este objetivo son puramente teóricos y testimoniales, explicando a que categoría pertenece (RAT Remote Access Trojan), que tipo de conexión usa (conexión inversa) y la explicación de que es una suite de control de remoto y que realiza acciones tan dañinas como tomar capturas de pantalla, activar la webcam, robar las pulsaciones del teclado...

7 Estado del arte

Los motivos de la elección de este proyecto son porque se estudia, recolecta, analiza…Por una parte, todo lo relacionado con la información, pilar clave en cuanto a la ciberseguridad se refiere. Dicha información está ordenada y distribuida y en gran cantidad. Por otra parte, he elegido el estudio de un malware por que el análisis de malware (sus características, su comportamiento…) es también uno de los aspectos íntimamente relacionado con la ciberseguridad, es decir, analizando algo que puede comprometer la seguridad de un equipo, red…

Las alternativas en cuanto a herramientas y malware son: en cuanto al uso de Internet Census, no he encontrado nada que se pueda comparar, lo cual parece lógico, ya que es difícil de encontrar una página que albergue tal cantidad de información. Sin embargo, habría resultado más eficiente encontrar una fuente de información con más datos, pues habría resultado más fácil encontrar la información requerida. En cuanto al estudio del WEBC2-DIV, como alternativa se podría haber estudiado algún otro malware asociado a APT1: AURIGA, BANGAT, BUISCUIT…

Algunos trabajos relacionados que tratan sobre APT1 más en general son los siguientes:

En [32], donde se explica el APT1 de forma teórica y algunos ejemplos de su ataque, como por ejemplo el ataque a unos determinados objetivos denominados Tibet Group 1 y Rgihts Group 1. En [33], donde se explica el APT1 de forma teórica, en las que se hace hincapié en el idioma mayoritario que tienen las víctimas (el inglés) y que la mayor parte de los ataques tuvieron como objetivo Estados Unidos. En [34], donde se explica los siguientes aspectos acerca de APT1: quiénes son (en líneas generales Unit 61398), cuáles son sus objetivos y cómo es de dañino. En [35], donde, aparte de información teórica, se muestran varios gráficos de tipo circular relacionados con APT1 y un ejemplo ilustrativo de ataque: captura de una sesión de video.

El siguiente apartado consta en “campañas” como objetivo de Gobierno/industria y CSOs (Centro de Operaciones de Seguridad). Mediante el análisis de clúster que agrupa ataques por malware de tipo común, patrones de desarrollo, infraestructura de tipo compartida, tácticas de ingeniería social y otros indicadores, se ha identificado diez grupos de ataques diferentes de los cuales cuatro tienen conexiones de tipo claro con campañas como objetivo el gobierno y la industria privada. Los hallazgos se hacen eco de informes anteriores que se remontan al menos al año 2008 (por ejemplo, Tracking GhostNet). También han mostrado actores de amenazas con objetivos a gobiernos y la industria privada. Los ataques relacionados son: APT1, DTL Campaigns, NetTraveler y PlugXCampaigns.

7.1 NetTraveler

En junio de 2013, Kaspersky publicó un informe que detalla las operaciones de un conjunto de amenazas que comprometió a más de 350 víctimas en 40 países distintos. Kaspersky llamó al malware principal usado en estas campañas "NetTraveler" después de una cadena interna que fue encontrada en la herramienta: "¡NetTraveler se está ejecutando!" Los objetivos identificados en este informe son tibetanos y grupos uigures, la industria energética, contratistas militares, investigación científica, centros, universidades, instituciones gubernamentales y embajadas.

El informe de Kaspersky identifica la IP 209.11.241.144 como un servidor de "mothership" utilizado como C2 y VPN. Vemos la dirección IP 209.11.241.144 como IP del remitente para 19 correos electrónicos en el estudio. Buscar otros correos electrónicos que compartan el mismo malware que encontramos en un total de 34 correos electrónicos, podemos dividir en siete campañas en función de cómo se utiliza la infraestructura C2 común. Además, había un correo electrónico de este remitente que no se pudo agrupar, ya que el archivo adjunto era un archivo ZIP protegido con contraseña y la contraseña no era sencilla. Ataques que usan infraestructura relacionada con NetTraveler se enfocaron en los cinco Grupos del Tíbet en nuestro estudio, así como en el Grupo 3 de China.

También se sabe que los operadores de NetTraveler usan ataques de "watering hole" contra sitios web tibetanos. En diciembre de 2012, F-Secure informó sobre un malware que se basaba en un diferente método de compromiso y ataque, pero utilizó la misma infraestructura. El sitio web relacionado con HHDL en www.gyalwarinpoche.com fue comprometido y fue servido el exploit Java CVE-2012-0507 (el mismo que se usa en el malware Flashback) para comprometer las computadoras que ejecutan el sistema operativo OS X. Este malware, que F-Secure llama Dockster, se reconecta a la misma IP que mandó muchos de los correos electrónicos maliciosos que observamos, itsec.eicp.net:8088 (209.11.241.144). Kaspersky ha documentado también riego similar de ataques de "watering hole" contra sitios web relacionados con uigures.

Los ataques de tipo "watering hole" se definen de la siguiente manera:

1. El atacante hace un perfil de la víctima y obtiene el tipo de webs que visita.
2. El ataque busca vulnerabilidades en las webs visitadas por la víctima.
3. El atacante compromete la web e inyecta código malicioso, redirigiendo a la víctima a otro sitio donde infectarla.
4. El sitio comprometido está preparado para infectar a la víctima.

La primera aparición de un ataque que utilizó infraestructura parecida a NetTraveler fue al Grupo 3 de China el 30 de abril de 2010. El correo electrónico adjuntaba un PDF que usaba CVE-2010-0188 y con conexión a servidores C2 en akashok.w63.1860host.com:80 (69.43.161.162) y ww2.akashok.w63.1860host.com:80 (204.13.161.108). La IP del remitente coincide con el "mothership" identificado por Kaspersky (209.11.241.144).

La primera de las tres campañas que usan la familia de malware Conime involucró siete correos electrónicos, cinco de ellos fueron distintos, en su mayoría relacionados con las manifestaciones del Levantamiento Tibetano del 10 de marzo.
Las muestras de Conime usadas en estos ataques tienen una puntuación técnica de 1,25. Estos correos electrónicos se enviaron entre el 13 de febrero y el 7 de marzo de 2012 y todos estaban dirigidos a Tibet Group 1. La campaña usó una combinación de documentos XLS y RTF maliciosos explotando CVE-2010-3333. La mayoría de estos ataques tienen un TTI de 3,75. Un correo electrónico solo tiene una puntuación de 2,0 en sofisticación de ingeniería social y un TTI general de 2,5.
Vemos el servidor de la "mothership" (209.11.241.1440) y 120.50.35.60 utilizados como correo remitente. Todos los ataques en esta campaña usaron 61.178.77.98 (sin un asociado nombre DNS) como C2.

La segunda de las tres campañas que usan la familia de malware Conime involucró a siete correos electrónicos, tres de los cuales eran diferentes, enviados a los Grupos de

Tíbet 1, 2 y 4. Estos correos electrónicos obtuvieron un valor de sofisticación de ingeniería social de 3.0, para un TTI combinado de 3.75. Dos de los correos electrónicos diferentes tenían dos documentos XLS adjuntos; uno estaba encriptado, el otro no lo era. El tercer correo electrónico usó un documento RTF malicioso que explota CVE-2010-3333. El XLS cifrado se envió el 25 de julio de 2012 y los demás correos electrónicos se enviaron entre el 10 y el 12 de septiembre de 2012. Nuevamente vemos 209.11.241.1440 como un correo electrónico IP del remitente. Todos estos exploits arrojaron la misma variante de Conime, que conectaba a gen2012.eicp.net:1080 (61.178.77.98) como C2.

La tercera campaña con Conime fue más variada que las otras dos y tuvo como objetivo los Grupos 1, 2 y 4 del Tíbet. Se recibieron quince correos electrónicos, once de ellos fueron distintos (aunque uno mostró solo cambios menores), que van desde 2.0 a 4.0 en la puntuación de sofisticación de la ingeniería social.
Dichos correos electrónicos se enviaron durante un período más largo de plazo que las otras campañas, que se expande entre el 14 de junio de 2012 y 12 de septiembre de 2012. Las vulnerabilidades utilizadas incluyeron los principales RTF (CVE-2010-3333, CVE-2012-0158) y XLS (CVE-2009-3129). Un correo electrónico, recibido por el Tíbet Grupo 2, recibió un puntaje de sofisticación de ingeniería social de 4.0. Este correo electrónico fue resaltado para nosotros por el destinatario como altamente dirigido, y se refirió a una conferencia próxima. Como las dos campañas anteriores de NetTraveler, el malware está conectado directamente a
61.178.77.98.

La cuarta campaña usó una variante de Gh0st RAT, con un texto de bandera de "Snow". Mismos correos electrónicos, en relación con una visita de HHDL a Portland, fueron enviados al “Tíbet Groups” 2 y 4 el 28 de enero de 2013. Los correos electrónicos tienen una puntuación de ingeniería social de 2, con una puntuación general de TTI de 2,5. los atacantes nuevamente usé 209.11.241.144 como correo remitente y 61.178.77.98 como C2.

La quinta campaña usó un malware de distinta familia, RegSubDat, que estaba dentro en un RTF usando CVE-2012-0158, adjunto a un correo electrónico enviado al “Tíbet Group 1”. Nuevamente se observa el correo enviado desde 209.11.241.144, pero en este caso el malware está conectado a un C2 diferente: server: itsec.eicp.net:443 (1.203.31.195). Este ataque obtuvo una puntuación de 3,0 en el valor de sofisticación de ingeniería para un TTI general de 3.75.

La última muestra del grupo NetTraveler fue demostrada en un mensaje de correo electrónico enviado al Tíbet Group 1 el 15 de marzo de 2012. Este malware se mandó desde la misma “mothership”, con servidor en 209.11.241.144, pero, en lugar de adjuntar el archivo malicioso, como se había hecho para todos los ataques anteriores, este correo electrónico contiene un enlace a un archivo XLS establecido. El archivo fue alojado en www.eaglessey.com (120.50.35.46), pero ya no estaba presente cuando se intentó acceder a él.

La campaña NetTraveler sirve como otro ejemplo de una campaña dirigida a las OSC junto con la industria y el gobierno. Estas campañas son conducidas por un prolífico “actor de amenazas” que se ha dirigido a una variedad de distintos sectores. Nuestros hallazgos confirman la identificación de grupos tibetanos como objetivo por Kaspersky, como los cinco de Tíbet. Esta campaña demuestra un “ataque adaptativo” que utiliza una variedad de vulnerabilidades para diferentes aplicaciones, incluida la orientación de las plataformas Mac y Windows.

7.2 PlugX

PlugX es una familia de malware que los investigadores han observado que se usa en ataques dirigidos contra organizaciones tibetanas, ONG, instituciones gubernamentales y compañías de tipo privado.

Trend Micro ha publicado un informe sobre PlugX, que describe una campaña que anteriormente usaba Poison Ivy, otra familia de malware. Jaime Blasco, en Alien Vault, dice haber rastreado al autor de PlugX, que supuestamente tiene su base en una compañía de seguridad China.

Las muestras de PlugX vistas en el estudio se pueden agrupar en cuatro campañas, según la IP del remitente de correo electrónico e infraestructura C2. Examinar temas relacionados con correo electrónico, vulnerabilidades utilizadas y rutas de compilación (como se describe en la publicación del blog Alien Vault) sugiere que las cuatro campañas son de la misma fuente. También hemos visto una muestra del malware Poison Ivy utilizada en esta campaña.
Los vectores de ataque y las vulnerabilidades utilizadas en PlugX son más variadas que otros ataques. Las vulnerabilidades utilizadas incluyen instancias de CVE-2012-0158 en tres formatos de archivo separados, una vulnerabilidad de Internet Explorer (CVE-2012-1889) que instalará PlugX como una descarga oculta y una vulnerabilidad Flash (CVE-2012-5054) que se encuentra en un sitio web externo. El ataque anterior de Poison Ivy usó dos archivos PDF más antiguos. La vulnerabilidad de Flash fue particularmente notable; era un día cero en el momento del ataque, dejando a cualquier usuario que hizo clic en el enlace malicioso en el que estaba alojado vulnerable al compromiso.

El primer conjunto de ataques consta de quince correos electrónicos, cinco de los cuales eran únicos, que se envían desde el 11 de mayo de 2012 al 1 de junio de 2012. El Grupo 1 del Tíbet y el Grupo 2 del Tíbet fueron atacados con al menos cuatro de los cinco correos electrónicos señalados. Estos correos electrónicos muestran muchas señales de venir de una misma fuente, incluida una dirección de retorno común de kandid77@rambler.ru, una IP del remitente de 98.126.14.13 y una infraestructura C2 común.

Se utilizaron dos nombres de dominio C2 diferentes: systen.windowsdeupdate.com (puerto TCP 8080) y web.windowsdeupdate.com (puerto UDP 7070). Estos nombres DNS señalaron a las mismas direcciones IP, incluidas 174.139.12.84 y 98.126.14.13.

Todos los correos electrónicos tienen un puntaje de ingeniería social de 3.0 y un TTI general de 4.5. Por ejemplo, se envió al Grupo 2 del Tíbet, falsificando a un oficial tibetano legítimo, un documento de Word que describe el calendario de una gira europea real realizada por el Dalai Lama.

El 22 de mayo de 2012, se mandó un correo electrónico desde la dirección IP 69.46.75.74 a “Tibet Group 2”, que afirmaba ser de un individuo llamado Tsering Dolma, con una firma de correo electrónico perteneciente a la Administración Central Tibetana, y con la dirección del remitente'tdolma6248@yahoo.com'. Este correo electrónico contenía un RTF adjunto con CVE-2012-0158 que se usó para instalar PlugX.

Se enviaron tres correos electrónicos al Grupo 2 de Tíbet y al Grupo 1 de China entre las fechas 15 de junio y 30 de agosto de 2012. Cada correo electrónico tenía contenido único, vectores de ataque, dirección de correo electrónico del remitente, IP, y vulnerabilidad. Dichas vulnerabilidades incluían una variante de Word de CVE-2012-0158, la vulnerabilidad de Flash CVE-2012-5054 y la vulnerabilidad de Internet Explorer CVE-2012-1889.

La vulnerabilidad de Flash CVE-2012-5054 era de día cero en el momento en que se utilizó en un ataque contra el Grupo 1 de China. El ataque se mandó en un correo electrónico muy personalizado para el destinatario y se utilizó un enlace malicioso en el mensaje como vector. Eso se refería a un grupo de individuos que recientemente habían estado involucrados en asuntos privados interno, como, por ejemplo, reuniones y parecía ser un mensaje reenviado del director de la organización. La naturaleza altamente dirigida de este ataque, combinada con la sofisticación técnica de la familia de malware PlugX, resultó en un puntaje TTI de 7.5, el más alto visto en el estudio.

La última campaña consistió en cuatro correos electrónicos únicos que se enviaron a los Grupos 1 y 2 del Tíbet entre el 22 de diciembre de 2012 y el 15 de enero de 2013. Todos estos correos electrónicos incluían archivos adjuntos de tipo CVE-2012-0158. El dominio C2 utilizado fue jinyuan2011.zapto.org:443, que se resolvió en la dirección IP 123.129.19.145 en el momento del ataque. tres de estos cuatro correos electrónicos obtuvieron 2.0 en el puntaje de sofisticación de ingeniería social (y 3.0 TTI en general), y uno obtuvo 3.0 en ingeniería social para un TTI general de 4.5.

Este ejemplo fue interesante porque no utilizó los archivos BOOT.LDR encriptados, en su lugar, se usó NvSmartMax.dll.url y el registro de los datos del teclado en NvSmart.hlp. Este ejemplo corresponde funcionalmente a las observaciones realizadas por los investigadores de Kaspersky de que PlugX estaba madurando. En particular, observamos que las cadenas de identificación y Los datos de registro se eliminaron. Es particularmente interesante que mientras el propio malware está mejorando, posiblemente en respuesta a los informes publicados de investigadores de amenazas, la calidad de la orientación en esta campaña ha disminuido.

En septiembre de 2012, Trend Micro describió el uso de PlugX en una “campaña” que anteriormente usó la RAT Poison Ivy y se dirigió a empresas gubernamentales y privadas en Japón. También vimos evidencia en nuestro estudio de Poison Ivy siendo usado en conjunto con PlugX en un ataque enviado al Grupo 2 de China el 10 de febrero de 2011, un año antes de nuestro primer ataque PlugX observado. Este correo electrónico incluía un PDF con dos vulnerabilidades, CVE-2009-4324 y CVE-2007-5659.

La RAT Poison Ivy se conecta a un C2 de nombre sociapub.flower-show.org:8080 (14.102.252.142), el mismo dominio Poison Ivy C2 observado por Trend Micro en 11 de Julio de 2012. Este ataque también se ha visto en otros lugares en la naturaleza, como se indica en un informe que describe el mismo malware visto con un tamaño de archivo diferente y un hash MD5 (9ADFC6DD86D5FF36F2CAB781663E1075).

La campaña PlugX proporciona otro ejemplo más de una “campaña” dirigida a la población de tipo civil, con organizaciones de la sociedad junto con grupos gubernamentales e industriales, utilizando la misma infraestructura y malware para poner en riesgo objetivos. Aparte de estas similitudes, por lo demás, la “campaña” tenía una serie de características únicas que la separaban de otros en nuestra investigación. En particular, fue la única “instancia” de una vulnerabilidad de “día cero” visto en nuestro estudio. Dado que los días cero son muy efectivos, ya que los desarrolladores de

software aún tienen que parchear la vulnerabilidad, son muy lucrativos y buscados. Es notable que los ataques maliciosos usarían este “día cero” para apuntar a una OSC (Centro de operaciones de ciberseguridad). Una vez que se expone un exploit de este tipo, corre el riesgo de ser identificado y tener la vulnerabilidad arreglada. La campaña PlugX también incluyó un conjunto más amplio de ataques de vectores de lo que se vio en la mayoría de las “campañas”. Los archivos adjuntos incluían la vulnerabilidad Flash de día cero, un exploit para Internet Explorer, así como el estándar de exploits de Microsoft Office vistos en otros lugares.

El apartado siguiente difiere de los anteriores debido a que es un ataque de naturaleza más específica y menos global, es decir, no será una familia de malware.

Toda esta información procede de [32].

7.3 La creencia de que China copió el “MQ-1 Predator Drone” a través de ciber-hackeo

QinetiQ Norteamérica (QQ) es una empresa de tecnología líder en defensa y seguridad proveedora de satélites, drones y servicios de software para las Fuerzas Especiales de EE. UU. Que están desplegadas en Afganistán y Oriente Medio.

En 2009, China tuvo casi su entero control sobre las computadoras de QinetiQ TSG, y consiguieron robar 1,3 millones de páginas de documentos y 3,3 millones de páginas de Microsoft Excel, conteniendo el código de TSG y los datos de ingeniería. Se creía que estos documentos se utilizaban por los chinos para fabricar el dron MQ-1.

¿China realmente lo está haciendo? ¿Lo admiten?

China dice: “Hemos comentado muchas veces que tales ataques son transnacionales y anónimos y determinar sus orígenes es extremadamente difícil". Así que están negando la acusación con firmeza.

El enfoque es de tipo indirecto. Primero el hacker pondría en un compromiso un servidor de EE. UU. y luego usaría eso para seguir atacando. La seguridad de la gente se veía muy comprometida debido a que visitaba ese servidor y luego se conectaba y se rastreaba la actividad. Después de toda esta evidencia, no hay manera de que ellos puedan negar eso, pero no se atreven a admitir el Cyber-Espionaje. El pensamiento puede ser que “Estados Unidos está haciendo eso todo el tiempo”.

La evidencia más condenatoria contra China, es la 'infraestructura del atacante' desde la cual se lanzan los ataques, el 98% de las veces estaban iniciando sesión desde ese bloque en Shanghái y el 97% de las veces se mostraban utilizando un conjunto de caracteres chinos en sus sistemas.

Esta información procede de [35].

8 Conclusiones

8.1 Conclusiones

El trabajo ha sido dividido en varios bloques muy diferenciados entre ellos, con un fuerte contraste entre ellos.

Todos los pasos se han realizado de forma organizada. Se han cubierto los objetivos de obtener, extraer, manipular la información y el cálculo de los resultados, tanto de las partes de Internet Census como en las del análisis de WEBC2-DIV mediante IDA y Cutter. En relación a la problemática del trabajo es que este ha tenido aspectos difíciles, como la obtención de determinados recursos mediante Internet (los relacionados con Internet Census, el malware WEBC2-DIV), y otros más fáciles, como el uso de algunos comandos Linux, tanto de la herramienta SilK como de otros comandos (grep, cut, rwsettool,…).

Se ha logrado identificar y almacenar las direcciones IP de las páginas correspondientes. Se ha logrado extraer los datos de Internet Census. No se ha logrado realizar un correcto filtrado, pero si lograr un resultado lo más óptimo posible. Se ha logrado estudiar el service proble, quizás excepto la intersección, cuya información es difícil de saber si es correcta o no. Se ha intentado realizar un análisis de datos mediante un archivo con información relacionada con ASN, el resultado no ha sido el esperado. Se ha logrado analizar el análisis del WEBC2-DIV mediante los programas adecuados (IDA y Cutter).

No se ha logrado realizar los apartados relacionados con Routing Data y Country Code ni los de Open Resolver.

Por último, las conclusiones obtenidas son que el trabajo puede servir de ayuda para alguien que tenga interés en conocer de forma detallada cómo funciona APT1.

8.2 Trabajos futuros

Analizar con Excel o LibreOffice Calc el conjunto de las direcciones IP almacenadas y usar sus comandos para mostrar distintas formas de encontrar y ordenar la información. Tanto Excel como LibreOffice Calc sirven para visualizar y trabajar con la información de manera óptima. Excel forma parte de “Office”, servicio cuyo propietario es Micrsoft. LibreOfficeCalc pertenece a LibreOffice. De esa manera podremos analizar con mayor fluidez la información relacionada con dichas IP, clasificándolas según su sistema operativo o qué puertos tienen abiertos. Usar las herramientas descritas arriba nos simplifica y clarifica mucho el trabajo a la hora de encontrar la información deseada.

Infectar con un malware contenido en APT1, por ejemplo, el examinado de nombre WINC2-DIV realizando un ataque de una máquina virtual a otra, es decir, realizar un enfoque práctico del uso del malware. Para ello se requeriría un profundo conocimiento de análisis de malware y saber que ataque realizar en cada momento (Intervalos de sueño, ejecución de malware…) y supervisar que dicho ataque se ha realizado de forma correcta. Todo ello en un entorno controlado y consentido por la parte atacada, o solo controlado si el ataque se realiza de forma local.

Examinar otros malware relacionados con APT1, como otro de la familia de WEBC2-DIV, AURIGA, BISCUIT… y comparar los resultados con los obtenidos entre ellos. Por ejemplo, AURIGA es un backdoor que comparte mucho contenido con respecto a funciones con el backdoor BANGAT. El malware puede iniciar un registrador de teclas (keylogging), conectarse a un controlador y crear una conexión a un servidor C2. BISCUIT se comunica mediante un protocolo personalizado, que después se cifra mediante SSL (Secure Socket Layer). Una vez que se instala BISCUIT intentará conectarse a sus servidores de comando/control alrededor de 10 o 30 minutos. En primer lugar, se dirigirá a su servidor principal, seguido de un servidor secundario. Toda la comunicación está encriptada con SSL (OpenSSL 0.9.8i). Los primeros datos enviados por el malware al host y puerto configurados son una “beacon sequence”. El campo de nombre de host en la “beacon squence” enumerada anteriormente es el nombre de host del sistema local. la ip es una lista de todas las direcciones IP del sistema local. mediante la misma conexión, el malware espera una de los siguientes comandos que se devolverán desde el servidor.

Bibliografía

[1] Matías Porolli, https://www.welivesecurity.com/la-es/2013/08/12/en-que-consiste-analisis-forense-de-informacion/, 12 de agosto de 2013 01:41 pm. Fecha de último acceso: 2 de Septiembre de 2022.

[2] Pedro Sánchez, http://www.innovarioja.tv/docs/1091/05.AUTOPSIA_FORENSEm-Pedro_Sanchez_.pdf, (sin fecha). Fecha de último acceso: 2 de Septiembre de 2022.

[3] CERT Network Situational Awareness (CERT NetSA), https://tools.netsa.cert.org/silk/, Junio 1991 group. Fecha de último acceso: 2 de Septiembre de 2022.

[4] Rubén Estrada Marmolejo, https://hetpro-store.com/TUTORIALES/comandos-linux-nivel-basico/, antes/sobre abril del 2001. Fecha de último acceso: 2 de Septiembre de 2022.

[5] autor desconocido, http://census2012.sourceforge.net/paper.html, 2012. Fecha de último acceso: 2 de Septiembre de 2022.

[6] hex-rais, cerca de 1999, https://hex-rays.com/ida-pro/ Fecha de último acceso: 2 de Septiembre de 2022.

[7] Rubén Velasco, https://www.softzone.es/programas/sistema/process-hacker/, 5 de Junio de 2020, 20:00. Fecha de último acceso: 2 de Septiembre de 2022.

[8] Cutter, https://cutter.re/, antes/sobre 2017. Fecha de último acceso: 2 de Septiembre de 2022.

[9] Redacción, https://cso.computerworld.es/tendencias/que-es-wireshark-asi-funciona-la-nueva-tendencia-esencial-en-seguridad, 19 de Septiembre de 2018. Fecha de último acceso: 2 de Septiembre de 2022.

[10] Matías Porolli, https://www.welivesecurity.com/la-es/2013/07/10/utilizando-inetsim-analisis-dinamico-malware/, 10 de Julio de 2013, 11:28 PM. Fecha de último acceso: 2 de Septiembre de 2022.

[11] Rubén Velasco, https://www.redeszone.net/2013/08/22/poison-ivy-sigue-siendo-utilizado-en-ataques-informaticos/, 22 de agosto de 2013, 17:00. Fecha de último acceso: 2 de Septiembre de 2022.

[12] Kaspersky, https://latam.kaspersky.com/resource-center/definitions/advanced-persistent-threats, antes de 2022. Fecha de último acceso: 2 de Septiembre de 2022.

[13] Mandiant, https://www.mandiant.com/sites/default/files/2021-09/mandiant-apt1-report.pdf, sobre 2013. Fecha de último acceso: 2 de Septiembre de 2022.

[14]Dan Mcwhorter, https://www.mandiant.com/resources/apt1-exposing-one-of-chinas-cyber-espionage-units, antes/sobre 2021. Fecha de último acceso: 2 de Septiembre de 2022.

[15] Monnappa K A, https://cysinfo.com/8th-meetup-understanding-apt1-malware-techniques-using-malware-analysis-reverse-engineering/, 2016. Fecha de último acceso: 2 de Septiembre de 2022.

[16] Deana Shick y Angela Horneman, https://resources.sei.cmu.edu/asset_files/TechnicalReport/2014_005_001_90523.pdf, Mayo de 2014. Fecha de último acceso: 2 de Septiembre de 2022.

[17] Arthur Salmon, https://www.youtube.com/watch?v=BGhZ0o0s7eM, 30 de septiembre de 2017. Fecha de último acceso: 2 de Septiembre de 2022.

[18]Mandiant,https://packetstormsecurity.com/files/download/120383/Mandiant_APT1_Report_Appendix.zip, fundado en 2004. Fecha de último acceso: 2 de Septiembre de 2022.

[19] Raidtya Faisal Waliulu y Teguh Hdayat Iskandar, https://journal.ittelkom-pwt.ac.id/index.php/inista/article/view/10/6, Alam 1 de Abril del 2018. Fecha de último acceso: 2 de Septiembre de 2022.

[20] whatsmyip, https://www.whatismyip.com/asn/, 1999. Fecha de último acceso: 2 de Septiembre de 2022.

[21] FBI, https://info.publicintelligence.net/NCCIC-MalwareIPs-1.pdf, 18 de Febrero de 2013. Fecha de último acceso: 2 de Septiembre de 2022.

[22] FBI, https://info.publicintelligence.net/NCCIC-MalwareIPs-2.pdf, 26 de Febrero de 2013. Fecha de último acceso: 2 de Septiembre de 2022.

[23] InfoByIp, https://es.infobyip.com/ipbulklookup.php, antes/sobre 2019. Fecha de último acceso: 2 de Septiembre de 2022.

[24] UNED. ETS de ingeniería informática, Introduccion_al_Analisis_Forense.pdf (ubicado en carpeta privada personal). Fecha de último acceso: 2 de Septiembre de 2022.

[25] Kaspersky, https://latam.kaspersky.com/resource-center/definitions/what-is-a-whaling-attack, antes de 2022. Fecha de último acceso: 2 de Septiembre de 2022.

[26] La redacción, https://laboratoriolinux.es/index.php/-noticias-mundo-linux-/software/17670-comprimir-y-extraer-archivos-con-zpaq.html, Sábado, 29 Abril 2017 10:41. Fecha de último acceso: 2 de Septiembre de 2022.

[27] CERT NetSA Security Suite, https://tools.netsa.cert.org/silk/rwtuc.html, 2006-2022. Fecha de último acceso: 2 de Septiembre de 2022.

[28] Mila, , http://contagiodump.blogspot.com/2013/03/mandiant-apt1-samples-categorized-by.html, 19 de Mayo de 2018. Fecha de último acceso: 2 de Septiembre de 2022.

[29] CERT NetSA Security Suite, https://tools.netsa.cert.org/silk/rwsetbuild.html, 2006-2022. Fecha de último acceso: 2 de Septiembre de 2022.

[30] file.extension, https://www.file-extension.info/es/format/set, 11/30/2019. Fecha de último acceso: 2 de Septiembre de 2022.

[31] CERT NetSA Security Suite, https://tools.netsa.cert.org/silk/rwpmapbuild.html, 2006-2022. Fecha de último acceso: 2 de Septiembre de 2022.

[32] Universidad de Toronto, https://tspace.library.utoronto.ca/bitstream/1807/80130/1/Deibert%20et%20al_2014_C

ommunities%20%40%20Risk.pdf, 11 de Noviembre de 2014. Fecha de último acceso: 11 de Septiembre de 2022.

[33] Nicholas Julian, http://www.sirjournal.org/research/2021/1/24/united-states-and-chinas-cybersecurity-policies-collaboration-or-confrontation, 24 de Enero de 2021, Fecha de último acceso: 11 de Septiembre de 2022.

[34] rajarshi, https://know.netenrich.com/blog/apt1-comment-crew-know-your-threat-actor/, 30 de Septiembre de 2021. Fecha de último acceso: 11 de Septiembre de 2022.

[35] Pranshu Bajpai, https://www.cse.msu.edu/~bajpaipr/resources/apt1_one_of_chinas_cyber_espionage_units_Pranshu_Bajpai.pdf, Marzo de 2010. Fecha de último acceso: 11 de Septiembre de 2022.