Un Enfoque de Teoría de Grafos para Mejorar la Clasificación y el Análisis de Crimeware

Jorge Kamlofsky – Jorge Mieres

Resumen* — *Crimeware es un software que realiza acciones ilegales contra usuarios que lo ejecutan a través de Internet. Las organizaciones de delitos cibernéticos distribuyen crimeware y lo difunden por todo el mundo. Debido a que los crimewares se usan a través de Internet, la mayoría de ellos están programados en PHP. Estos programas tienen una raíz en el archivo 'index.php' y sus archivos y carpetas forman una estructura de grafo tipo árbol a partir de éste. Por lo tanto, analizar crimeware usando un enfoque de Grafos es natural y adecuado. En este trabajo, se utilizó un modelo basado en Teoría de Grafos para encontrar las huellas dejadas por los programadores en el crimeware: se analizó una muestra de más de 100 crimewares. Los resultados experimentales sorprenden.

***Palabras Clave* — *Clasificación de Malware, Ransomware, botnets, crimeware*.**

I. Introducción

A. Trabajos Relacionados

El phishing es un mecanismo criminal que utiliza ingeniería social y trucos técnicos para robar datos de identidad personal de consumidores y credenciales de cuentas financieras. Usan correos electrónicos falsos que simulan ser de compañías y / o agencias legítimas, diseñados para redirigir a los consumidores a sitios web falsos, para robar información personal y credenciales para acceder al sitio original. En muchos casos, instalan crimeware en las PC para robar directamente las credenciales interceptando las comunicaciones mientras el usuario infectado realiza transacciones en línea [1].

Los ataques de phishing suelen ser el punto de partida del delito cibernético. Los resultados obtenidos permiten perpetrar el delito cibernético, no solo para la recolección de datos, sino también para la instalación de crimeware. El APWG (Anti-Phishing Working Group) define Crimeware: aplicación de software diseñada para delitos financieros en entornos en línea. Según los datos proporcionados por Panda Labs (miembro de APWG), durante 2015, hubo un fuerte aumento de PUP (Programas potencialmente no deseados) a través de paquetes de software que instalan programas sin el consentimiento del usuario. Finalizando 2015 una familia de crimeware causó caos en todo el mundo: Ransomware. Consiste en un malware que secuestra archivos o dispositivos que inhabilitan ciertos componentes [2] normalmente codificando datos.

En [3], Kaur y Kaur proponen un algoritmo de detección de clones (usando redes neuronales) para determinar los clones de códigos y unirlos con las firmas de malware en el repositorio para detectar Ransomwares. Para describir, comparar y comprender el ransomware, Gazet utiliza técnicas de ingeniería inversa [4]. Wan presentó un clasificador de malware que usa un algoritmo de clasificación de patrones [5]. En [6] Ruiz Azofra describe criptografía detrás de malwares: en ransomwares modernos, se pueden encontrar dos capas de cifrado RSA con claves de 2048 bits que codifican claves simétricas AES de 256 bits para encriptar datos secuestrados.

El APWG registró más phishing en 2016 que en cualquier año desde que comenzó a monitorear en 2004 [7]. El informe de tendencias de la actividad de phishing de 2016 establece que los ataques de phishing fueron 65% más que en 2015. Los diez primeros puestos de la clasificación de máquinas infectadas están encabezados por países asiáticos: China, Turquía y Taiwán. Y el resto de los países, excepto Rusia (6 ° lugar), son de América Latina: Guatemala, Ecuador, Perú, México, Venezuela y Brasil [7]. Debido a su impacto económico, Brasil tiene su propio capítulo en el informe.

Bots y crimewares están programados para Internet. Por lo tanto, el lenguaje de programación favorito es PHP[1]. PHP tiene una estructura de árbol clara con raíz en el archivo "index.php" donde se inicia la aplicación. Muchos programadores normalmente organizan métodos y / o funcionalidades en carpetas, cada una con su archivo "index.php". Y si se necesita un nuevo programa, el programador generalmente lo usa para copiar sus carpetas, formando su propio patrón de programación. Esta huella puede ser presentada por la teoría de grafos. Este documento presenta cómo rastrear programadores y/o clasificar crimeware para encontrar estas huellas.

B. Objetivos de este Trabajo:

Presentar una forma de identificar partes o componentes de malware de modo de mejorar la clasificación o descripción de crimeware.

Monitorear la evolución de crimeware y/o sus componentes.

C. Relevancia del Tema:

La evolución de las estrategias de infección y el monitoreo de las máquinas infectadas proponen la necesidad de recopilar información de interés sobre delitos, de modo de poder recolectar, identificar, clasificar y analizar el código fuente de cada software delictivo, descubrir su potencialidad y, de acuerdo con esto, proponer estrategias de mitigación para anticipar potenciales ataques.

El análisis de la estructura del crimeware junto con el código fuente proporciona la capacidad de recopilar datos

1 http://php.net/manual/es/intro-whatis.php

puntuales que pueden procesarse para generar una base de conocimiento y una capacidad analítica para producir inteligencia no solo durante los procesos de análisis de crimeware, sino también como consecuencia de la acción de recopilación, identificación, clasificación y análisis de procesos; tratando de descubrir quién está detrás de estas maniobras cibercriminales.

La evolución de las estrategias de infección y el monitoreo de las máquinas infectadas proponen la necesidad de recopilar información de interés sobre delitos, que permite recolectar, identificar, clasificar y analizar el código fuente de cada software delictivo, descubrir su potencialidad y, de acuerdo con esto, proponer estrategias de mitigación para anticipar un ataque potencial.

El análisis de la estructura del crimeware junto con el código fuente proporciona la capacidad de recopilar datos puntuales que pueden procesarse para generar una base de conocimiento y una capacidad analítica para producir inteligencia no solo durante los procesos de análisis de crimeware, sino también como consecuencia de la acción de recopilación, identificación, clasificación y análisis de procesos; tratando de descubrir quién está detrás de estas maniobras cibercriminales.

D. Nuestra Contribución:

La forma en la que los programadores desarrollan una aplicación y reusan códigos forman una huella. Este trabajo presenta una forma de mostrar esa huella.

E. Estructura de este Trabajo:

En la sección 2 se presenta el estado del arte. La sección 3 contiene información acerca del modelo propuesto. La sección 4 muestra datos experimentales y luego la importancia de los hallazgos. La sección 6 presenta las conclusiones, luego Trabajos Futuros y Referencias.

II. Estado del Arte

A. Cibercrimen:

Según Gordon y Ford [8], las definiciones de delito cibernético cubren desde la actividad delictiva contra los datos hasta el contenido y la infracción de los derechos de autor. Sin embargo, Zeviar-Geese [9] sugiere que la definición es más amplia, incluidas actividades como el fraude, el acceso no autorizado, la pornografía infantil y el acecho cibernético. El Manual de las Naciones Unidas para la prevención y el control de los delitos informáticos incluye el fraude, la falsificación y el acceso no autorizado [8] en su definición de delito cibernético.

Las investigaciones muestran que el número de personas y empresas afectadas por el delito cibernético está creciendo sin signos de disminución [7, 8].

La Organización Cibercriminal. Los hackers solitarios que operan independientemente o grupos de hackers con objetivos comunes han sido reemplazados por organizaciones jerárquicas de cibercrimen.

Sorprendentemente, están tan bien organizados como organizaciones criminales tradicionales como La Cosa Nostra [10]. El jefe: él es el jefe de la organización, y normalmente no se entromete directamente en los crímenes. El sub-jefe: proporciona troyanos y administra el comando y control (C & C) de esos troyanos. Administrador de Campañas: opera debajo del sub-jefe. Cada administrador de campaña tiene su propio territorio, mercado, con características específicas (llamado "campaña"). Red de afiliación: se usa para realizar ataques y robar datos. Ellos actúan como distribuidores. Atacantes: algunos simplemente operan malware. Otros atacantes piratean sitios legítimos insertando código malicioso para ser operado por otro. Revendedores: intercambian los datos robados. Implementan modelos de precios para los diferentes tipos de sus productos, incluso ofrecen garantía en la mayoría de los casos.

Modelo de Negocios Crimeware. Los delincuentes cibernéticos usan sofisticados modelos de negocio Criminal-2-Criminal (C2C). Estos profesionales del delito cibernético utilizan Crimeware robusto y escalable que les brinda la máxima flexibilidad en términos de C & C para robar y comercializar datos. Están implementando la estrategia empresarial '*pensar globalmente, actuar localmente*'. Tienen juegos de herramientas de crimeware que consisten en paquetes de software que guían a los atacantes a cómo realizar ataques paso a paso [10]. Los creadores de kits de herramientas de Crimeware también implementan el modelo de negocio de Software como servicio (SaaS), denominado *Crimeware-as-a-Service* (CaaS).

Efectos. Los ciberdelincuentes pueden ganar acceso a los balances de las empresas de todo el mundo y luego manipular los valores de las acciones. También pueden realizar transferencias bancarias, obtener acceso al presupuesto de las empresas y estados financieros; robar planes de productos de la compañía y propiedad intelectual para espionaje industrial (y militar).

Dado que los ataques web utilizan agujeros de seguridad en los navegadores, el problema se ha convertido en uno de los más importantes, que compromete a las empresas, organizaciones y personas de todo el mundo [10].

B. Crimeware:

Crimeware es un software que realiza acciones ilegales imprevistas por un usuario que ejecuta el software. Estas acciones están destinadas a generar beneficios financieros para el distribuidor del software. Crimeware es un hecho omnipresente de la vida en las interacciones modernas en línea. Se distribuye a través de una amplia variedad de mecanismos y ataques [11].

Las estadísticas de crimeware de APWG categorizan los ataques de crimeware a través de la taxonomía: esta crecerá a medida que se generen variaciones en el código de ataque.

Definición: Crimeware es un código diseñado con la intención de recopilar información sobre el usuario final para robar las credenciales del usuario.

A diferencia de la mayoría de los keyloggers genéricos, los registradores de pulsaciones basados en phishing tienen componentes de seguimiento, que intentan monitorear acciones específicas (y de organizaciones específicas, como instituciones financieras, comerciantes minoristas y de comercio electrónico) para obtener información específica [1].

Objetivo: Robo de Información Sensible. El crimeware se puede usar para obtener información confidencial de todo tipo, como: nombre de usuario y contraseñas, números de tarjetas de crédito, números de cuentas bancarias e información personal. También se usa para obtener credenciales para obtener acceso a VPN y luego, para robar secretos industriales y / o comerciales, crear un ataque de denegación de servicio o encriptar información comprometida y luego ofrece descifrar los datos por una tarifa.

Enfoque. Crimeware es una subclase de una categoría de malware que generalmente se refiere a software no deseado que realiza acciones maliciosas en la computadora del usuario. Además de malware, el crimeware posiblemente incluya software legal pero malicioso como adware y spyware, software ilegal sin fines comerciales, como virus destructivos.

Crimeware no incluye, pero puede usar, análisis de red y otras herramientas legales con potencial para ser parte de un ataque.

Propagación. El crimeware generalmente se disemina mediante ingeniería social o mediante la explotación de vulnerabilidades de seguridad. Un típico ataque de ingeniería social consiste en convencer a un usuario para que descargue un archivo que contiene código malicioso. Ejemplo: un correo electrónico que lo invita a ver imágenes atractivas (pornografía, bromas, etc.) y ha adjuntado código malicioso, o sitios que ofrecen aceleradores gratuitos o herramientas de rendimiento que también contienen el código adicional (Rogue [2]). Las vulnerabilidades de seguridad a menudo se basan en errores de programación. Por ejemplo, en navegadores web o complementos. Crimeware podría aprovechar estos e instalar código malicioso en la máquina invitada. Estos se llaman Exploits [2].

Variantes de Crimeware. Según Jacobsson y Zulfikar [10], no existe una clara división entre las variantes de crimeware porque muchos ataques son híbridos: usan múltiples tecnologías. Keyloggers y Screenscrappers: Keyloggers son programas que monitorean la entrada de datos en una máquina. Por lo general, se instalan en un navegador web o como un controlador de dispositivo. Los dispositivos de captura de pantalla supervisan la entrada del usuario y una parte de la pantalla. Pueden frustrar las medidas de seguridad de entrada alternativas como los teclados en pantalla. Ambos recopilan un conjunto de credenciales y luego lo envían a un servidor remoto. Redireccionamientos de correo electrónico y mensajería: son programas que interceptan y transmiten mensajes salientes (correos electrónicos o mensajes instantáneos) y envían una copia adicional a una dirección involuntaria a la que tiene acceso el atacante. Secuestradores de sesión: se refiere a un ataque en el que se secuestra una sesión de usuario legítimo una vez que el usuario estableció sus credenciales para realizar acciones maliciosas. Troyanos web: estos son programas maliciosos que colocan pantallas de inicio de sesión falsas, en un esfuerzo por recopilar credenciales de sitios específicos. La información recopilada se almacena localmente y luego se transmite al atacante.

Crimeware Basado en Motivación e Intentos. El término "crimeware" también incluye la intención que motiva este desarrollo. Desde este enfoque, ciertas clases de crimeware se pueden identificar como siguientes.

Kit de malware: programas que tienen una plataforma de administración web (generalmente escrita en PHP) desde donde se registran y monitorean las actividades de la computadora infectada. Y un generador que le permite desarrollar malware personalizado. Los ejemplos son ZeuS, SpyEye, Carber, etc.

Exploit Kit: Programas que tienen una serie de exploits precompilados para ser utilizados por los atacantes, que pueden usar todos los exploits o no, generalmente escritos en PHP y que a diferencia del primero no incluye un constructor así como el " malwarekit "no incluye exploits. El kit de exploits generalmente tiene una administración web básica. Pero el malware asociado generalmente se distribuye, generando una RAT (por ejemplo, PoisonIvy). Ejemplos de kits Exploit son: BlackHole, Phoenix, Eleonore, YES Exploit System, DdoS Framework. Desarrollos: generalmente en PHP, destinados a administrar y monitorear ataques DDoS. Algunos ejemplos de Frameworks DdoS son: DirtJumper, RusKill.

C. Técnicas de Detección de Malware:

Las empresas que luchan contra el malware (proveedores de antivirus) reciben miles de muestras nuevas todos los días. Las firmas que detectan amenazas maliciosas confirmadas se crean principalmente de forma manual, por lo que es importante discriminar entre las muestras que representan una nueva amenaza desconocida y las que son meras variantes de malware conocido [12].

Cuando se recibe malware, se desea conocer sus características para la identificación y clasificación (análisis) y luego, desarrollar contramedidas. Se utilizan técnicas de análisis manuales y automáticas. Las técnicas de análisis automático, se pueden clasificar en estático y dinámico. El análisis estático obtiene características sin ejecución de código. Está muy extendido ya que la barrera tecnológica es baja. En [13] se presentan los límites de estas técnicas. El análisis dinámico requiere la ejecución de código en un entorno controlado. La mayor complejidad del malware requiere un entorno de análisis más complejo. En [12] se presentan las técnicas de análisis dinámico más utilizadas.

D. Algunas Definiciones y Conceptos Básicos de Teoría de Grafos:

Grafo: Un grafo G es un conjunto de vértices $v \in V$ y aristas $e \in E$ tal que cada arista de E está relacionada con un par de vértices de V [14].

$$G = (V, E)$$

La definición anterior permite que un grafo contenga aristas paralelas y lazos o loops.

Grafo Simple: El grafo G es simple si y solo si en G no hay aristas paralelas ni lazos.

Grafo Completo de n Vértices: El grafo G es un grafo completo si G es simple y cada par de loa n vértices está conectado con una arista.

Grafo de Similaridad: Este consiste en agrupar objetos similares en clases basadas en las propiedades de los objetos.

Para construir un grafo de similaridad, debe presentarse una función de similaridad. Frecuentemente estas funciones usan algún tipo de distancia entre las propiedades de los objetos, definidas como sigue:

$$s(v, w) = |p_{v1} - p_{w1}| + |p_{v2} - p_{w2}| + \ldots + |p_{vn} - p_{wn}|$$

donde p son propiedades. Ambos grafos son similares, si $s < t$ (con t: un valor umbral)

Grafo con Pesos: Es un Grafo en el que cada contiene un número (el peso de la arista).

Peso de un Grafo: El peso de un grafo es la suma de los pesos de cada una de las aristas del grafo.

Subgrafo: Un graph $G' = (V', E')$ es un subgrafo de $G = (V, E)$ si y solo si:

I) $V' \in V$ y $E' \in E$

II) Por cada $e' \in E'$ si e' incide en v_i' y v_j', entonces $v_i', v_j' \in V'$

Árbol: Un grafo T es un árbol si para cada par de vértices de T existe un único camino, sin aristas paralelas ni lazos.

Árbol con Raíz: Es un árbol donde un vértice es designado como raíz y los restantes se orientan alejándose de la raíz.

Subárbol: Si T es un árbol, T' es un subárbol de T si T' es un subgrafo de T.

Grafo Estrella: Es un árbol con un vértice interno y k vértices hoja (vértices sin hijos)

Representación: Los grafos pueden representarse por conjuntos, por gráficas representando vértices con pequeños círculos y aristas con líneas. Los grafos también pueden representarse mediante matrices.

III. El Modelo Propuesto

A. Breve Descripción del Modelo:

Archivos y carpetas de cada muestra de crimeware forman un árbol simple sin pesos con raiz en la carpeta con el nombre del crimeware.

Para cada muestra del conjunto, puede obtenerse su correspondiente árbol de archivos y carpetas.

Por cada vértice (archivos o carpetas) se obtiene el hash MD5 y se lo almacena como nuevo nodo de un árbol análogo MD5. Con ésto, los archivos de los crimewares se los compara por su contenido, firmado mediante su hash MD5.

Basado en el árbol MD5 de cada muestra se intentará encontrar subárboles en el resto de las muestras y luego se interpretan los resultados hallados.

Hay subárboles compartidos entre varios crimeware. Entonces, un nuevo grafo de relaciones entre los crimeware puede construirse. Y el peso de este nuevo grafo indicará la intensidad de las relaciones entre los crimeware.

B. Ambiente Experimental:

Equipamiento: Los experimentos se realizaron en una notebook Dell Inspirion 3420 plataforma x86 de 64bits con 4 CPUs y 4GB RAM. Sistema operative: Kali Linux Rolling basado en un núcleo Debian. La algoritmia se programó en Python 2.7.

Muestras Usadas: El conjunto de muestras usadas consiste en 101 muestras de crimeware. Estas fueron entregadas por una empresa del sector: *MalwareIntelligence*, luego de ser pedidas a varias empresas del sector.

C. Usuarios Potenciales de este Modelo:

Empresas Anti-malware: Empresas que desarrollan soluciones para detectar y remover malware pueden desarrollar herramientas basadas en este modelo para detectar fácilmente crimeware instalado o una aplicación que intenta instalar crimeware en una computadora.

Agencias Públicas que Combaten al Cibercrimen: Divisiones de Cibercrimen de Policía, Fiscalías, Juzgados, u organizaciones que luchan activamente contra el cibercrimen pueden usar este modelo para ayudar en la persecusión de autores y distribuidores.

Centros de Monitoreo: Entidades responsibles del monitoreo de la evolución del cibercrimen y CERTs pueden usar este modelo para detectar apaciciones y analizar evoluciones en la dispersión del crimeware.

D. Ventajas de la Propuesta:

Fácil Entendimiento: La representación gráfica de los grafos permite que los grafos sean fácilmente interpretados y el fenómeno estudiado, fácilmente entendido. En este caso, los subárboles compartidos permiten construir grafos de relaciones entre ellos indicándonos el grado de familiaridad entre ellos, permitiendo además conocer su origen.

Fácil de Implementar: Los grafos son fácilmente programables. Los subárboles se construyen a partir de los árboles de archivos y carpetas.

IV. Resultados Experimentales

A. Resultados Generales:

El conjunto conteniendo 101 muestra de crimeware tiene un total de 23.662 archivos. De ellos, 3430 archivos aparecen en más de un crimeware. Esto significa que casi el 15% de los archivos y carpetas son archivos reusados.

B. Subárboles compartidos:

Un subárbol compartido consiste en un conjunto de vértices vecinos presentes en más de un crimeware. Y forman una estructura de árbol. El conjunto de subárboles compartidos consiste desde sólo un conjunto de vértices aislados hasta estructuras de árbol más complejas.

La Figura 1 muestra un árbol compartido entre varios crimeware del conjunto.

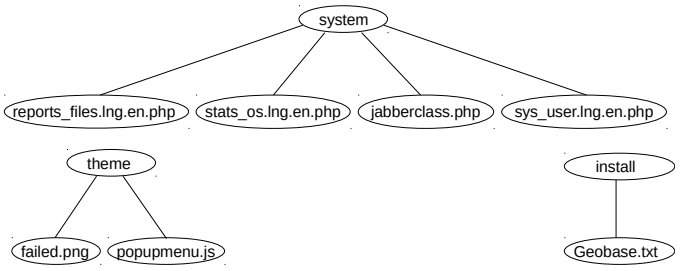

FIGURA 1: Subárboles compartidos entre 7 crimewares.

C. Grafo Estrella de Relaciones entre Crimeware con Árboles Compartidos:

Un grafo de más alto nivel puede construirse con crimeware que comparte subárboles: un grafo estrella. Los subárboles compartidos conformarán un nodo central y los crimewares serán las hojas o extremos del grafo estrella. Cada arista tendrá un peso cuyo valor será la cantidad de vértices compartidos.

Entonces, el peso del grafo puede ser un buen indicador de la intensidad de las relaciones entre crimewares: más crimewares compartiendo más vértices, mayor intensidad en las relaciones entre crimewares. La Tabla 1 muestra los pesos de los grafos estrella más destacados.

Tabla 1: Grafos Estrella más Destacados.

Shared Vertexes	Number of Repetitions	Weight of the star graph	Crimewares containing shared vertices
240	9	2160	CoinerHTTP Bot, BlackHole_v1.0.3, Db0t v2 (m\xc3\xb6glicherweise Infected) Vorsicht, LibertyExploitSystem_2.11, Atrax, BlackHole ExploitPack, coiner_2.0.1, Athena, AldiBOT
238	9	2142	[vOlk-Botnet]5.0.2(MX), vOlk-Botnet 4.0.2(MX), vOlk-Botnet 1.0.0(MX), Sapz(PE), [vOlk-Botnet] v1.6(MX), UELP@(LatAm), vOlk-Botnet 4.0(MX), Kyuss Exploit, vOlk_iquitos_mod(LatAm)
180	7	1260	DatalifeExploitPack-0.12, multisploit, BankBOT, ATDS, BlackEnergy.DDos.Bot.v1.8_VIP_, Baracuda, A-311DeathBackdoor
7	7	49	zeus Merry Chritsmas, Citadel 1.3.4.5 Botnet(ZeuS_Clon), Citadel_1.3.5.1(ZeuS_Clon), IceIX-1.5, Athena1, citadel_CP(ZeuS_Clon), panel

La figura 2 muestra al grafo estrella con la mayor intensidad en las relaciones.

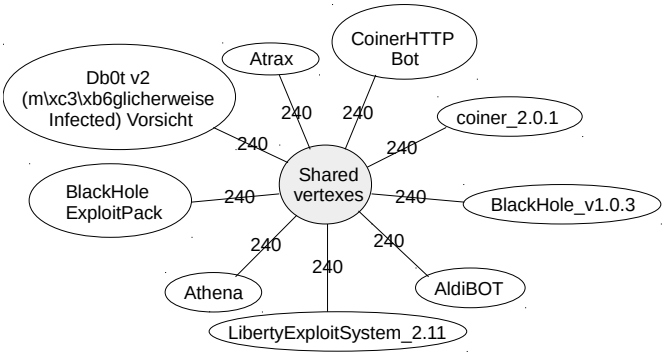

FIGURE 2: El grafo estrella de relaciones más pesado.

D. Grafo Estrella de Archivos Vacíos:

Durante el experimento se encontraron archivos vacíos con diferente nombre y extensión. Este puede ser un patrón de programadores. Con ello, un grafo estrella sin pesos puede graficarse, donde el nodo central es el vértice “archivo vacío” y las hojas o extremos lo conforman los nombres de los crimewares.

Los siguientes crimeware tienen al menos un archivo vacío: ZEUS_1.3.1.1, CZ Stat_1.1.1-7a, BlackHole_v1.0.3, SpyEye_Mario-Mod, Carbon Grabber, zeus Merry Chritsmas, BlackEnergy.DDos.Bot.v1.8_VIP_, BetaBOT_v1.5, Carbon form Grabber, ATDS, IceIX-1.5, SpyEye_v1.3, TinyBanker, DirpJumper_v2, BManager_1.4, AldiBOT, carberp_, YESSploitSystem_2.0, BetaBOT, multisploit, Athena1, Seo Sploit Pack, cc-grabbers admin panel bender edition, DirpJumper_v3, A-311DeathBackdoor, Citadel 1.3.4.5 Botnet (ZeuS_Clon), Citadel_1.3.5.1 (ZeuS_Clon), BetaBOT_v1.0.2.5, CrimeTime(LatAm), Mexico-Kit_ss2EXE (LatAm), BlackHole ExploitPack, Cry217, citadel_CP (ZeuS_Clon), Crimepack_3.1.3, Raterov1 [Mejorado] - (LatAm), Betabot_1.2.5, Db0t v2 (m\xc3\xb6glicherweise Infected) Vorsicht, DatalifeExploitPack-0.12, ReFF (LatAm), Carberp, Hugomixer_Drones_Master (LatAm), Kameleon, YESSploitSystem_1.0, BManager, panel.

La suma de los crimewares conteniendo archivos vacíos es

43. Por lo tanto, el grafo estrella tendrá un centro y 43 hojas o extremos.

V. Importancia de los Hallazgos

El reciclado y la reutilización de archivos es mucho más frecuente de lo esperado.

Según el enfoque presentado, los archivos reutilizados pueden rastrearse: es posible saber quiénes están detrás de las maniobras criminales. También es posible analizar la evolución de las versiones de crimeware y así tener una idea de las tendencias y proyecciones de las mutaciones del crimeware y así anticipar al cibercrimen.

Desde el punto de vista de la investigación, todos los instrumentos delictivos dejan evidencia que puede vincularse de manera inteligente a través de diferentes procesos. En consecuencia, el análisis en profundidad de estos paquetes de software es de suma importancia, principalmente porque facilita la detección y el estudio de nuevas versiones de crimeware basadas en otras; incluyendo nuevos y emergentes paquetes de software criminal que ayudan a investigar posibles maniobras de ataque, incluso durante su ejecución. En este sentido, un caso que motivó una línea de investigación reciente fue el crimeware latinoamericano conocido como "FlokiBot", que al estudiar la estructura de su código y compararlo con otras fuentes de crimeware, se estableció que está basado en la versión 2.0 . 8.9 de un crimeware muy peligroso llamado ZeuS cuyo código fuente fue lanzado en 2011.

En consecuencia, todos los procesos derivados del análisis de crimeware facilitan la documentación sobre su estructura y objetivos para comprender mejor su funcionamiento y así crear procedimientos de mitigación en tiempo real que se pueden acoplar a la Política de Seguridad de la Información y/o al Plan de Contingencia.

Por otro lado, durante cualquier proceso de investigación en delitos informáticos es de suma importancia tratar de encontrar patrones que a modo de prueba permitan la identificación del autor (es), así como cualquier otra indicación relacionada con el origen geográfico del desarrollo malicioso, zona de ataque objetivo y bajas potenciales.

VI. Conclusiones

Debido a que los crimewares tienen una estructura de árbol, un análisis basado en un enfoque de gráfico es natural y adecuado.

A partir de la búsqueda de subárboles compartidos, obtuvimos dos tipos de gráficos tipo estrella que permiten mostrar relaciones o familiaridad entre diferentes programas. La información obtenida de una pequeña muestra es muy importante, y su representación gráfica es muy fácil de entender, lo que permite mejorar la lucha contra el delito cibernético.

VII. Trabajos Futuros

En este trabajo, se presentó un modelo basado en la teoría de grafos. Su implementación en una muestra pequeña dio como resultado la obtención de algunos gráficos específicos. Su implementación en muestras de mayor tamaño y variedad puede dar como resultado la creación de gráficos nuevos, más reveladores y útiles.

Referencias

[1] Anti Phishing Working Group. “Phishing Attack Trends Report – 4Q2015”. [En Línea], (2016). Disponible en: <http:// docs.apwg.org / reports / apwg_trends_report_q4_2015.pdf>. Fecha de Consulta: 25 de Abril de 2016.

[2] ESET. “Seguridad antimalware para Partners”, (2009)

[3] Kaur, Simarleen, and Arvinder Kaur. "Detection of malware of code clone using string pattern back propagation neural network algorithm." *Indian Journal of Science and Technology* 9.33 (2016).

[4] Ruiz Azofra, Eduardo. “Técnicas Criptográficas Utilizadas en Malware”, [En Línea], (2015). Disponible en: <http: // oa.upm.es / 38772 / 1 / PFC_EDUARDO_RUIZ_AZOFRA_2015.pdf>. Fecha de Consulta: 15 de Abril de 2017.

[5] Wan, Justin. "Malware detection using pattern classification." *U.S. Patent No. 8,161,548*. (2012).

[6] Gazet, Alexandre. "Comparative analysis of various ransomware virii." *Journal in computer virology* 6.1 (2010): 77-90.

[7] Anti Phishing Working Group. “Phishing Attack Trends Report – 4Q2016”. [En Línea], (2017). Disponible en: <http:// docs.apwg.org / reports / apwg_trends_report_q4_2016.pdf>. Fecha de Consulta: 15 de Abril de 2017.

[8] Gordon, Sarah, and Richard Ford. "On the definition and classification of cybercrime." *Journal in Computer Virology* 2.1 (2006): 13-20.

[9] Zeviar-Geese, Gabriole. "The State of the Law on Cyberjursidiction and Cybercrime on the Internet." *Gonz. J. Int'l L.* 1 (1997): 119.

[10] Ben-Itzhak, Yuval. "Organised cybercrime and payment cards." *Card Technology Today* 21.2 (2009): 10-11.

[11] Jakobsson, Markus, and Zulfikar Ramzan. “Crimeware: understanding new attacks and defenses.” *Addison-Wesley Professional*, (2008).

[12] Egele, Manuel, et al. "A survey on automated dynamic malware-analysis techniques and tools." *ACM Computing Surveys (CSUR)* 44.2 (2012): 6.

[13] Moser, Andreas, Christopher Kruegel, and Engin Kirda. "Limits of static analysis for malware detection." *Computer security applications conference, 2007. ACSAC 2007. Twenty-third annual*. IEEE, (2007).

[14] Johnsonbaugh, Richard. “Discrete Matematics”. *Pearson Prentice Hall*, (2009).

Jorge Kamlofsky. Es Licenciado en Matemática graduado en la Universidad Abierta Interamericana (UAI). Es Especialista en Criptografía y Seguridad Teleinformática, posgraduado en la Facultad de Ingeniería del Ejército Argentino (EST - IUE). Se encuentra finalizando una Maestría en Tecnología Informática y se encuentra promediando un Doctorado en Ingeniería en la Universidad Nacional de Lomas de Zamora (UNLZ). Actualmente es profesor de Matemática Discreta en la UAI y de Algebra y Geometría Analítica en la Universidad Tecnológica Nacional (UTN). Es investigador del Centro de Altos Estudios en Tecnología Informática (CAETI) dependiente de la Facultad de Tecnología Informática de la UAI.

Jorge Mieres Jorge Mieres tiene más de 15 años de experiencia en investigación cibercriminal y malware. En 2006 fundó MalwareIntelligence (www.malwareint.com) y en los últimos años fue premiado por Microsoft como MVP en Seguridad Empresaria. A lo largo de estos años, colaboró con investigaciones en compañías y policías y fuerzas de seguridad en diversos países de Latinoamérica. Forma parte de varias comunidades privadas de investigación y frecuentemente es llamado a disertar en eventos de Seguridad informática y en Universidades.. Jorge sirvió como Investigador Senior de tretas y Responsable de Investigación para Iberoamérica en iSIGHT Partner’s, una empresa de ciberinteligencia. Anteriormente fue parte del equipo latinoamericano de investigación de las empresas Antivirus KarpeskyLab (GreAT) y ESET (Nod 32).