ANÁLISIS DE LAS CARACTERÍSTICAS DE INFECCIÓN, OCULTACIÓN Y PROPAGACIÓN DEL MALWARE ACTUAL

Jairo Alonso Corredor Montero

Universidad Pontificia Bolivariana
Bucaramanga, Colombia

Resumen

McAfee en su reporte de amenazas para el primer cuarto del 2012, muestra como en este periodo "se ha detectado el mayor número de malware por cuarto en los últimos cuatro años", al mirar los reportes de noticas sobre seguridad informática se puede inferir que el malware se hace cada vez más especializado y se propone como una posible arma para una guerra cibernética, causando preocupación principalmente en dos frentes, el robo de información en los sistemas bancarios y el ataque a infraestructuras críticas a través de sistemas SCADA.
Dentro de este documento se analizarán de las técnicas de infección, ocultación y propagación del malware actual haciendo un énfasis en particular sobre los que han causado mayor impacto en los últimos años dentro de la industria como Zeus, SpyEye, Stuxnet y recientemente Flame; finalmente se propondrán algunas recomendaciones a manera de conclusión para hacer frente a estas nuevas técnicas

Palabras claves

Malware, infección, ocultación, propagación, Zeus, SpyEye, Stuxnet, Flame.

Abstract

McAfee threats report for first quarter 2012, says that “in this period shows the largest number of malware detected per quarter in the last four years”, looking news reports about security information can be inferred that the malware is becoming more specialized and is proposed as a possible weapon for cyber warfare, causing concern mainly over two fronts, theft of information on banking systems and the attack on critical infrastructures through SCADA systems.
Within this paper will consider the infection, hiding and propagation malware techniques doing a particular emphasis on those who have caused the greatest impact over the industry in recent years like Zeus, SpyEye, Stuxnet, and recently Flame; finally propose some recommendations in conclusion to combat these new techniques.

Keywords:
Malware, infection, hiding, propagation, Zeus, SpyEye, Stuxnet, Flame.

1. Introducción

Como lo demuestra el reporte de amenazas de McAfee para el primer cuarto del 2012, el malware aumenta constantemente y se hace cada vez más especializado tanto en rootkits como en sus funcionalidades; de hecho en este cuatrimestre “se ha detectado el mayor número de malware por cuarto en los últimos cuatro años” [1], y no sólo se muestra como un riesgo para los sistemas tradicionales sino que actualmente ya se está clasificando como un arma para una guerra cibernética [2], causando preocupación principalmente en dos frentes, el robo de información en los sistemas bancarios, “se detectan un promedio de 7,800 nuevos troyanos bancarios por mes” [3] y el ataque a infraestructuras críticas a través de sistemas SCADA mediante malware como Stuxnet, Duqu y Flame [2].

A continuación se detallan las técnicas usadas por el malware actual, que les permiten infectar a sus víctimas, ocultarse y propagarse dentro de un sistema.

2. Vulnerabilidades de Día Cero

Son vulnerabilidades que no han sido detectadas hasta ahora o para las cuales el fabricante no cuenta con un parche disponible para solucionarlas. Esto brinda una ventana de tiempo dentro de la cual el ataque es totalmente efectivo.

Aunque el aprovechamiento de vulnerabilidades de día cero para la infección de sistemas no es una novedad, cabe destacar la sofisticación a la que se está llegando al combinarlas; *Stuxnet* por ejemplo, usaba cinco vulnerabilidades diferentes (MS10-046, MS10-061, MS10-073, MS10-092, CVE-2010-2772) [4] para infectar un sistema haciéndolo altamente eficaz contra los Sistemas Operativos Windows, éstas incluían ejecución de código desde dispositivos USB aún si el *AutoRun* estaba desactivado y copiado de archivos a través de impresoras compartidas en la red [5].

Las vulnerabilidades de día cero son usadas para lograr la infección y propagación inicial, y dependiendo de su naturaleza pueden permitir la ejecución de código arbitrario, elevación de privilegios, propagación a través de redes *WiFi*, *Bluetooth* e inclusive servicios de impresión.

3. Técnicas Anti-Análisis

Estas técnicas son usadas para proteger a los ejecutables del análisis de ingeniería inversa mediante la ofuscación del código y dificultan las labores de inspección de los Antivirus y los Sistemas de Detección de Intrusos (IDS). Dentro de las utilidades más comunes para proteger el contenido y generar nuevo programa se encuentran los *Packers* que se usan para ofuscar y empaquetar o comprimir el código, los *Crypters* que aplican un algoritmo de cifrado sobre el código y los *Binders* que mezclan el código con librerías propias del sistema operativo creando una aplicación portable [6].
Actualmente existen herramientas como *ExeStealth, ASPack, Molebox, Obsidium, PE Crypt 32, PolyCryptPE, Themida, etc.*, que automatizan esta labor y permiten además, añadir características adicionales como detección de debuggers, máquinas virtuales y *sandboxs*, así como validaciones de ejecución de la aplicación en modo seguro o como administrador.

Malware como *Zeus* además de estar empaquetado, también incorporó técnicas adicionales como la detección de *Firewalls* y la adición de una cantidad de bytes aleatorios al final del archivo cuando se copia así mismo para que su *Hash* sea distinto cada vez [7].

SpyEye por su parte destacó por su original forma de acceder a las funciones del *Kernel* sin generar alarmas, ya que no hace un llamado directo mediante el nombre de la función sino que calcula su *hash* y luego lo compara con el *hash* de todas las funciones del *kernel* hasta que coincida devolviendo su dirección [7], de esta forma lograba eludir a los antivirus y los IDS.

4. Do It Yourself Kits

Una tendencia que se viene generalizando en los últimos años es el desarrollo de *malware* modular y con ello el uso de los *Kits* de herramientas *Do It Yourself* (DIY) para ajustarlo a la medida. A través de un panel de control (C&C[1]), se pueden seleccionar las características deseables para el *malware* como el uso de *KeyLoggers*, robo de contraseñas de redes sociales, cuentas bancarias, cuentas *FTP* y de correo electrónico, historial de sitios *Web* visitados y lectura de la información del caché, conversaciones de *chat*, etc.

Adicionalmente estas herramientas traen incorporados *rootkits* y *exploits* que permiten automatizar la infección, ocultación y propagación. También cuentan con un archivo de configuración en donde se almacenan las opciones seleccionadas por el usuario y las direcciones web a donde se envía la información recolectada o de donde se descargan funcionalidades que no han sido incluidas en el archivo original.

En la actualidad, *BlackHole* and *Phoenix* son considerados los *kits* líderes del mercado; sin embargo, recientemente se ha dado a conocer *RedKit* que es el último DIY para *Zeus* y cuenta con *exploits* para dos vulnerabilidades. La primera explota una vulnerabilidad en la *LibTIFF* de los *PDFs* (CVE-2010-0188) y la otra explota una vulnerabilidad de *Java* denominada *AtomicReferenceArray* (CVE-2012-0507), su C&C permite monitorear el número de clientes infectados e inclusive ofrece la posibilidad de escanear con 37 antivirus diferentes el archivo de infección para asegurar su efectividad [8].

[1] C&C: Command & Control

5. Botnets

Es un "Conjunto de ordenadores infectados por un tipo de *malware*, que permite al atacante controlar dicha red de forma remota" [9] se usan generalmente para el robo de información, fraudes publicitarios, ataques de denegación de servicio distribuidos (DDoS) y envío de *spam*, entre otros.
Malware como *Zeus*, *SpyEye* o *Duqu* usa *botnets* para robar datos sensibles de sus víctimas, inicialmente este tipo de redes eran controladas a través del protocolo *Internet Relay Chat* (IRC), sin embargo tras la aparición de *Zeus* empezaron a orquestarse a través de la *Web*; a finales del 2010 este tipo de *botnets* doblaban su número cada 18 meses [10], actualmente siguen en aumento siendo *Cutwail* el malware con mayor número de equipos infectados [1].

El éxito de las *botnets* a través de *Web* radica en 2 aspectos fundamentales, el *Hypertext Transfer Protocol* (HTTP) "no suele ser filtrado hacia afuera en un sistema, puesto que impediría la navegación estándar" [10] y su facilidad para crear el C&C ya que sólo es necesario instalar un servidor para alojar las páginas Web, una Base de Datos *MySQL* y sobre ella, ejecutar los scripts del que son suministrados por el kit DIY [11].

6. Malware Firmado

Al finalizar el 2011 ya se empezaban a ver con más frecuencia en los reportes de noticias de seguridad informática, ataques que involucraban certificados digitales fraudulentos; autoridades de certificación como *DigiNotar* y *Digicert Malaysia*, así como casas antivirus como *Avira* y *Kaspersky* fueron víctimas de estas falsificaciones.

Posteriormente comenzó a aparecer *malware* firmado con certificados válidos como en el caso de la empresa china *Realtek* a quien le fue robada su clave privada y se usó para firmar los drivers de *Stuxnet* [5]. Esta tendencia ha tenido un aumento casi exponencial y para el primer cuatrimestre del 2012 ya se han detectado más de 200.000 binarios de *malware* con una firma digital válida [1].

Recientemente se conoció que *Flame* viene firmado por *Microsoft*; bajo alguna autoridad intermedia se consiguió manipular un certificado de *Terminal Server* y firmar su código, debido a que el software firmado por *Microsoft* pasa a las listas blancas de los antivirus este *malware* ha conseguido pasar durante 5 años inadvertido [12].

7. Flame

Flame es catalogado actualmente como una ciber arma de espionaje de sistemas industriales [2] e incorpora técnicas que lo hacen realmente único. Tiene características de robo de información similares a las del *malware* bancario como *Zeus* y *SpyEye*, entre ellas la discriminación de las aplicaciones de las que va a registrar información, captura de pantalla, registro de las pulsaciones del teclado y adicionalmente integra comunicación por *Bluetooth* y grabación por medio de la activación del micrófono [13].

Análisis realizados por expertos dejan ver la especialización que tiene Flame para el ciber espionaje. Tiene preferencias sobre archivos PDF, Office y gráficos de AutoCAD, los documentos cargados en el C&C se cifran, los documentos robados se comprimen, la infraestructura de C&C se ha movido múltiples veces los últimos 4 años y además se desconectó una vez se reveló la existencia del *malware* [14] y por si fuera poco el 8 de Junio de 2012 entró en ejecución un módulo denominado *suicide* que busca eliminar los archivos y las carpetas en donde se alberga el código malicioso [15].

La forma en que se creó el certificado con el que va firmado el código también es novedosa, se crearon 2 certificados sobre licencias de *Terminal Server*, uno normal y otro con los campos modificados, posteriormente se manipularon ambos certificados hasta que se logró una colisión en su *hash* MD5. Finalmente se envió a la autoridad certificadora (CA) la versión "normal" y cuando esta estuvo firmada, se copió la firma de la CA en el certificado falso. Al coincidir su *hash* el certificado se valida [16].

Otro aspecto novedoso es la propagación del *malware* en las redes internas, ya que *Flame* intercepta las llamadas a *Windows Update* y simula un *proxy*, si Internet Explorer está configurado para detectar automáticamente el *proxy*, *Flame* hace un ataque de hombre en el medio (MitM) para hacerse pasar por el servidor de actualización y descargarse en el nuevo servidor, como la actualización en el certificado falso se marcó como crítica *Flame* se instala inmediatamente [17].

Flame al igual que muchos otros malware usa técnicas anti-análisis como la ofuscación y el cifrado e incorpora un componente adicional, tiene partes lógicas escritas en lenguaje *Lua* y las ejecuciones se realizan por medio de librerías en C, *Lua* al ser un lenguaje interpretado y sin tipado de variables dificulta enormemente su análisis estático ya que los valores que pueden tomar las variables dependen del estado del programa en ejecución, "todo esto hace que Kaspersky estime que estudiarlo por completo les llevará un año" [13].

8. Conclusiones

La combinación de las técnicas descritas convierte al *malware* en una herramienta de infección y propagación altamente efectiva, por ello es de importancia crítica permanecer informado sobre las últimas vulnerabilidades reportadas y aplicar los parches o correctivos necesarios para mantener el software actualizado y no correr riesgos.

También se debe realizar un monitoreo constante del tráfico de la red y los servicios activos, ya que el *malware* especializado en robar información podría estar ejecutándose y no ser percibido por las herramientas de diagnóstico tradicionales, en este caso se hace fundamental la labor del analista de seguridad que basado en los datos obtenidos podrá detectar comportamientos anómalos y evitar la fuga de información.

Según las tendencias actuales, se van a seguir observando muchos más *malware* firmados y una mayor cantidad de ataques sobre los algoritmos de cifrado por eso es de vital importancia suprimir de las aplicaciones los algoritmos a los que se les han detectado debilidades y usar siempre los que se consideran seguros.

También se puede pronosticar que los malware incorporarán cada vez más segmentos de código escritos en lenguajes interpretados como Lua ya que esto les permite dificultar las labores de análisis de los antivirus y garantizar una mayor ventana tiempo durante la cual son indetectados. Finalmente, se aprecia una clara tendencia para el uso del *malware* como herramienta de espionaje y ataque a sistemas industriales.

Referencias Bibliográficas

[1] McAfee Labs. McAfee Threats Report: First Quarter 2012. [En línea]. Disponible: http://www.mcafee.com/us/resources/reports/rp-quarterly-threat-q1-2012.pdf

[2] Kaspersky Lab. Kaspersky Lab and ITU Research Reveals New Advanced Cyber Threat. [En línea]. Disponible: http://www.kaspersky.com/about/news/virus/2012/Kaspersky_Lab_and_ITU_Research_Reveals_New_Advanced_Cyber_Threat

[3] Kaspersky Lab. Kaspersky Lab Performs Best in Dedicated Testing Against Online Banking Threats. [En línea]. Disponible: http://newsroom.kaspersky.eu/en/texts/detail/article/kaspersky-lab-performs-best-in-dedicated-testing-against-online-banking-threats/

[4] ESET. Stuxnet Under the Microscope. [En línea]. Disponible: http://go.eset.com/us/resources/white-papers/Stuxnet_Under_the_Microscope.pdf

[5] Hispasec Sistemas. Éxitos y fracasos de Stuxnet. [En línea]. Disponible: http://unaaldia.hispasec.com/2010/10/exitos-y-fracasos-de-stuxnet-i.html

[6] C. H. Malin, E. Casey y J. M. Aquilina. (2008, Jun). Malware Forensics: Investigating and Analyzing Malicious Code. [En línea]. 340 - 351. Disponible: http://books.google.com.co/books?id=lRjO8opcPzIC&pg=PA340&hl=es&source=gbs_toc_r&cad=4#v=onepage&q&f=false

[7] SANS Institute. Clash of the Titans: ZeuS v SpyEye. [En línea]. Disponible: http://www.sans.org/reading_room/whitepapers/malicious/clash-titans-zeus-spyeye_33393

[8] Trustwave SpiderLabs. A Wild Exploit Kit Appears... Meet RedKit. [En línea]. Disponible: http://blog.spiderlabs.com/2012/05/a-wild-exploit-kit-appears.html

[9] INTECO. BOTNETS ¿Qué es una red de ordenadores zombis?. [En línea]. Disponible: http://www.inteco.es/file/p9cSCisIwwtRK6a0e7iZKg

[10] Hispasec. Las botnets controladas por web doblan su número cada 18 meses. [En línea]. Disponible: http://unaaldia.hispasec.com/2010/11/las-botnets-controladas-por-web-doblan.html

[11] Hispasec. Vídeo: Así funciona SpyEye (I). [En línea]. Disponible: http://unaaldia.hispasec.com/2010/11/video-asi-funciona-spyeye-i.html

[12] Hispasec. TheFlame, el sueño de todo creador de malware. [En línea]. Disponible: http://unaaldia.hispasec.com/2012/06/theflame-el-sueno-de-todo-creador-de.html

[13] Hispasec. TheFlame: reflexiones sobre otra "ciberarma" descubierta demasiado tarde. [En línea]. Disponible: http://unaaldia.hispasec.com/2012/05/theflame-reflexiones-sobre-otra.html

[14] Kaspersky. Expertos de Kaspersky Lab ofrecen un análisis en profundidad de la Infraestructura C&C de Flame. [En línea]. Disponible: http://www.kaspersky.com/sp/about/news/virus/2012/Expertos_de_Kaspersky_Lab_ofrecen_un_analisis_en_profundidad_de_la_Infraestructura_de_Flame

[15] BBC. Flame malware makers send 'suicide' code. [En línea]. Disponible: http://www.bbc.co.uk/news/technology-18365844

[16] Hispasec. La creación del certificado falso usado por TheFlame, ridiculiza la estructura PKI de Microsoft (y II). [En línea]. Disponible: http://unaaldia.hispasec.com/2012/06/la-creacion-del-certificado-falso-usado_11.html

[17] Hispasec. El ingenioso método de distribución de TheFlame en redes internas. [En línea]. Disponible: http://unaaldia.hispasec.com/2012/06/el-ingenioso-metodo-de-distribucion-de.html