Análisis Heurístico: detectando malware desconocido

Autores: David Harley, Security Author and Consultant
Andrew Lee, Chief Research Officer de ESET
Fecha: Martes 27 de marzo del 2007

ESET, LLC 1172 Orange Ave., Coronado, California 92118, USA
phone: (619) 437 – 7037, fax: (619) 437 – 7045
sales@eset.com, www.eset.com

Tabla de Contenidos:

Introducción

Viendo a los Detectives

- *Virus*
- *Gusanos*
- *Malware No Replicativo*

¿Qué significa Heurística, puntualmente?

- *Exploración por Firmas*
- *Lo opuesto a la Heurística*
- *Antivirus Genéricos*
- *Soy absolutamente Positivo*
- *Sensibilidad y Diagnóstico erróneo*
- *Cuestiones de la Evaluación*

Conclusión: Una paradoja Heurística

Acerca de los Autores

Referencias

Glosario

Introducción

"El problema no es lo que no sabes, sino lo que crees que sabes y no es así"

Algunos de los mitos más persistentes en computación se relacionan con la tecnología de malware y antivirus (AV). La ampliamente aceptada creencia de que el software antivirus sólo puede detectar códigos maliciosos específicos (conocidos), ha estado circulando desde los inicios de las investigaciones AV. No era verdad en su totalidad por aquel entonces; algunos de los primeros programas AV no estaban intencionados a detectar malware específico, sino más bien para detectar o bloquear comportamientos semejantes al de códigos maliciosos o bien, cambios sospechosos en los archivos. Ciertamente, tampoco es verdad en la actualidad.

Los sistemas comerciales AV complementan la exploración por firmas con una variedad de acercamientos más genéricos, que generalmente están agrupados bajo el nombre de análisis heurístico. Más aún, la mayoría de los nuevos productos AV son capaces de detectar un amplio espectro de software malicioso (malware es una conjunción de las palabras "malicioso" y "software"), no solamente virus. Esto puede estar combinado con otras tecnologías de seguridad como son: la detección de spam y phishing.

El objetivo de este artículo es reducir las confusiones en torno al funcionamiento de la tecnología AV y clarificar qué es lo que realmente debe esperarse de una protección antivirus, particularmente aquella que cuenta con análisis heurístico.

Las especificidades de la exploración heurística son discutidas en detalle. Por el momento, simplemente describiremos el análisis heurístico como un método para estimar la probabilidad de que un programa aún no identificado como malware sea, sin embargo, viral o malicioso.

Viendo a los Detectives

La mayoría de lo que vemos referido como virus podría ser mejor descrito como malware

¿Qué es lo que detecta un software Antivirus? Relativamente bastante, como sucede, incluyendo algunos ítems que técnicamente no son virus. La mayoría de lo que vemos referido como virus podría ser mejor descrito como malware. La ironía está en que muchos productos de detección especializada (por ejemplo, para detectar spyware o troyanos) se imponen en el mercado como necesarios porque los antivirus solo detectan virus.

De hecho, los antivirus comercializados detectan a una variedad más amplia de malware que la mayoría de aquellos servicios especializados. Un programa especializado puede detectar más amenazas dentro de su propia especialidad, pero esto depende no solo de la capacidad del programa para detectar amenazas específicas y tipos de amenazas, sino también de otros factores como pueden ser:

- Las capacidades de detección genérica del programa
- El criterio de diferenciación utilizado para las variantes de malware
- Los mecanismos entre los fabricantes de antivirus para compartir muestras (las casas antivirus cuentan con modos efectivos y bien establecidos de hacer esto, comparado con casas de fabricantes dentro de otras áreas de detección de malware)

Las siguientes secciones consideran tres principales tipos de malware. Una completa taxonomía de todo el malware excedería a los propósitos de este escrito.

Virus

Ciertamente, es razonable esperar que un software antivirus pueda detectar virus, y es en parte porque los antivirus han sido tan exitosos en esta detección durante años, que su capacidad para detectar otros tipos de malware ha sido subestimada.

Es en parte porque los antivirus han sido tan exitosos en esta detección durante años, que su capacidad para detectar otros tipos de malware ha sido subestimada.

Mientras que existen varias definiciones de virus, una definición aceptada por la mayoría de los investigadores de malware es “un programa informático que puede infectar a otros programas modificándolos de manera tal que incluya una (posiblemente evolucionada) copia de si mismo” [1,2].

Esta definición cubre varios tipos de virus, incluyendo:

- Infecciones del sector de arranque y/o tabla de partición
- Infecciones de archivos (virus parasitarios)
- Virus multipartitos
- Virus macro y scripts

Mientras que algunos de estos tipos de virus difícilmente sean vistos en la actualidad (por ejemplo infecciones del sector de arranque o de tabla de partición), los programas AV generalmente detectan todos los virus conocidos para la plataforma en la que son encontrados (y a veces para otras plataformas). Generalmente, también son bastante buenos en la detección heurística de nuevos y desconocidos virus “reales”.

Gusanos

La industria antivirus nunca ha llegado al consenso respecto de lo que son los gusanos, como alegó Cohen, "caso especial de virus" [1], pero cualquiera sea el caso, los software antivirus normalmente los detectan igualmente.

Hay al menos tantas definiciones de gusano como de virus, pero la mayoría de los investigadores AV definen un gusano como programa que se reproduce no parasitariamente, por ejemplo, sin adjuntarse a sí mismo al archivo que lo hospeda. Los mailers masivos podrían ser descritos como un tipo específico de gusano. La mayoría de las compañías AV describen este tipo de malware como gusano, pero algunos mailers y mailers masivos tienen las características de un virus "puro" (Melissa, por ejemplo, de hecho fue un virus "puro", un macro virus que se esparció como gusano, mientras que el W32/Magistr fue una infección de archivo).

Aquí también, los fabricantes tienen un buen manejo de la detección de nuevas variantes. Los nuevos envíos masivos, por ejemplo, generalmente son detectados por los proveedores de sistemas de seguridad de mensajes casi inmediatamente desde su aparición.

Malware No Replicativo

Se deduce de las definiciones anteriores que si un programa malicioso no se replica, no puede ser ni virus ni gusano. Pero eso no significa que el software antivirus no pueda detectarlos o que no sean dañinos.

Hay que tener en cuenta que aún cuando los fabricantes se quejaban frente a la detección de objetos no replicativos porque no se trataba de virus, algunos de estos objetos (varios de los cuales sin ser siquiera programas ejecutables, menos aún maliciosos) fueron no obstante, detectados y señalados. [3] Por ejemplo:

- *Intendeds* (virus que fallan en replicarse) y corrupciones
- Archivos basura
- Programas relacionados con virus pero que no lo son, como droppers y generadores de virus
- Programas de testeo legítimos como el archivo EICAR [4].

Muchos objetos no replicativos han circulado por años en colecciones de virus con poco mantenimiento que fueron utilizadas por algunas personas para evaluar software antivirus. La mayoría de los fabricantes dejaron de protestar hace rato y agregaron definiciones (firmas) para estos objetos

Muchos objetos no replicativos han circulado por años en colecciones de virus con poco mantenimiento.

a sus bases de datos, con la esperanza de evitar ser penalizados por no detectarlos. Desafortunadamente, la creciente sofisticación de exploraciones heurísticas apenas ha podido seguirle el paso a la habilidad de los examinadores AV para encontrar nuevos y no siempre apropiados modos de evaluación. Más adelante en este escrito, consideraremos brevemente los mecanismos técnicamente aceptables de examinar las capacidades heurísticas de un producto.

El más conocido de los códigos maliciosos no replicativos es el troyano. Un troyano es "un programa que aparenta realizar alguna función deseable o necesaria y de hecho puede que lo haga, pero además realiza alguna función o funciones no esperadas ni deseadas por el usuario que ejecuta el programa" [5]. Esto cubre un rango específico de malware, incluyendo:

- Droppers
- Keyloggers
- Troyanos destructivos
- Downloaders
- Spyware
- Adware
- Rootkits y Stealthkits
- Programas joke (algunos)
- Zombies (bots, troyano de acceso remoto, agentes DDoS, y más).

El malware replicativo, tal como virus, pueden también ser descritos como troyanos (o troyanizados, implicando que un programa previamente legítimo ha sido subvertido, alterado o suplantado para hacerlos, de algún modo, dañino), aunque la mayoría de las personas suelen considerar este uso más como confuso que como una ayuda. La detección de todas las versiones de malware no replicativo es aún menos alcanzable que la detección de todas las formas de virus, dado que debe evaluarse una más amplia variedad de funciones que la mera habilidad de replicarse.

Mucho del debate en cuanto a lo que es y no es un troyano (o malicioso) descansa no tanto en la función, sino más bien en el intento. Por ejemplo, un keylogger no es un troyano si ha sido legítimamente o consensuadamente instalado, aún cuando la función resulte idéntica. Esto deviene en problemas en la detección, ya que las computadoras son menos capaces que los humanos en determinar intentos.

Spyware y adware –tal vez debido al creciente interés de los medios de comunicación y los productos disponibles exclusivamente para su detección- han sido recientemente separados en su propia subclase de malware. Aquí, sin embargo, la distinción es mayoritariamente innecesaria si bien podría (y con frecuencia lo es) discutirse que el adware en particular no siempre es malware. No obstante, el mismo argumento puede aplicarse a la mayoría de los elementos de la lista, en que no es lo que el programa realiza que lo hace malicioso; es la brecha entre las malas intenciones de los programadores y la expectativa del usuario en cuanto al programa.

¿Qué Significa Heurística, puntualmente?

El término "Heurística" refiere al acto o proceso de encontrar o descubrir. El Oxford English Dictionary define Heurística como "permitiendo que una persona descubra o aprenda algo para ella misma" o (en el contexto de la informática) "proceder hacia una solución por ensayo y error o mediante reglas definidas sin ataduras" [6]. El Merrian Webster Dictionary la define como "una asistencia al aprendizaje, descubrimiento o resolución de problemas mediante métodos experimentales o de ensayo y error" o (nuevamente en el contexto de la informática) "vinculada con técnicas exploratorias de resolución de problemas que utilizan técnicas autodidactas (tales como evaluación o feedback) para mejorar el desempeño" [7]. La Real Academia Española la define como "Técnica de la indagación y del descubrimiento" y además aclara que: "En algunas ciencias, manera de buscar la solución de un problema mediante métodos no rigurosos, como por ejemplo, tanteo, reglas empíricas, etc".

Generalmente, la programación heurística es considerada como una de las aplicaciones de inteligencia artificial y como herramienta para la resolución de problemas. La programación heurística, tal como es utilizada en sistemas expertos, se construye bajo reglas extraídas de la experiencia y las respuestas generadas por tal sistema mejora en la medida en que "aprende" mediante experiencia a futuro y aumenta su base de conocimiento.

Si bien el modo en que es utilizado el análisis heurístico en el manejo de malware (y sin duda, spam e inconvenientes relacionados), está estrechamente relacionado con elementos de prueba y error y aprendizaje por experiencia, también posee una definición más estricta.

El análisis heurístico utiliza un acercamiento basado en reglas para diagnosticar un archivo potencialmente ofensivo (o mensaje en el caso de análisis de spam).

El análisis heurístico utiliza un acercamiento basado en reglas para diagnosticar un archivo potencialmente ofensivo (o mensaje en el caso de análisis de spam). Dado que el motor analítico trabaja a través de su base de reglas, chequeando el mensaje contra criterios que indican posibles malware, asigna cierto puntaje cuando localiza uno semejante. Si el puntaje iguala o supera el puntaje del umbral [8], el archivo es señalado como sospechoso (o potencialmente maliciosos o spammy) y procesado acorde a ello.

En cierto sentido, la heurística antimalware intenta aplicar los procesos de análisis humanos a un objeto. De igual modo que un analista humano de malware intentaría determinar el proceso de un determinado programa y sus acciones, el análisis heurístico realiza el mismo proceso de tomar decisiones inteligentes, actuando efectivamente como un investigador virtual de malware. Dado que el analista humano de malware aprende más acerca de amenazas emergentes, el/ella puede aplicar ese conocimiento al analista heurístico a través de la programación y mejorar los índices de detección futuros.

La programación heurística posee un doble rol en el desarrollo antivirus: velocidad y detección. De hecho, el término Heurística es aplicado en otras áreas de la ciencia [9] en un sentido similar; apuntando a mejorar el desempeño (especialmente en la velocidad de obtención de resultados) a través de un resultado lo "suficientemente bueno" antes que el más exacto. Dado que el número total de virus conocidos se ha incrementado, también creció la necesidad de mejorar la velocidad de detección. De otro modo, el incremento de tiempo necesario para explorar un número de programas maliciosos cada vez mayor, haría inutilizable el sistema.

Dado que el número total de virus conocidos se ha incrementado, también creció la necesidad de mejorar la velocidad de detección.

Aún los primeros exploradores heurísticos utilizaban patrones simples de detección beneficiada por las técnicas de optimización que investigaban solamente las partes de un objeto en las que podría esperarse encontrar un virus determinado. (Un ejemplo simple – no tiene sentido explorar un archivo completo en busca de un virus por firmas, si dicho virus siempre almacena el código de su núcleo al inicio o al final de un archivo infectado). Esto reduce la exploración a fondo y disminuye el riesgo de generar falsos positivos.

La detección inapropiada de una firma de virus en un lugar donde el virus nunca sería hallado en circunstancias normales no sólo es un efecto colateral de una metodología pobre de detección, sino también un síntoma de evaluaciones de detección pobremente diseñadas. Por ejemplo, algunos evaluadores han intentado examinar las capacidades de un programa antivirus insertando, de forma aleatoria, códigos de virus en un archivo u otro tipo de objeto infectable. En forma similar, un tipo particular de objeto tal como un archivo o sector de arranque puede ser explorado en forma selectiva únicamente para aquellos tipos de malware que pueden realmente esperarse de ser hallados en dicho objeto, proceso a veces descrito como "filtrado". Después de todo, no hay razón por la cual buscar códigos de virus macro en el sector de arranque.

Sin embargo, una correcta identificación de un tipo de archivo no es prueba concreta de un archivo no contaminable. Por ejemplo, los archivos de documento de Microsoft Word que contienen ejecutables maliciosos ocultos han sido, por largo tiempo, un vector principal de ataque para el robo de información y espionaje industrial. Similarmente, los autores de malware están constantemente en busca de ataques donde un objeto que normalmente no es capaz de ejecutar códigos, puede ser modificado para hacerlo como por ejemplo, modificando el ambiente de ejecución. W32/Perrun, por ejemplo, se agregó a si mismo a los archivos .JPG y .TXT, pero no podía ejecutarse a menos que se hicieran cambios específicos en el ambiente operativo para permitirle al código Perrun ser extraído y ejecutarse.

Exploración por Firmas

De hecho, muchos virus no pueden ser identificados buscando solamente una cadena estática.

La exploración por firmas se refiere al patrón de emparejar algoritmos en busca de una secuencia de bytes (una cadena de caracteres), característico de cada virus o sus variantes en las bases de datos de las definiciones, pero una que no sea propensa a encontrarse por accidente en un archivo que no ha sido infectado. Algunos investigadores de AV intentaron desincentivar [2] el uso de la descripción de exploración por firmas a favor de una "búsqueda de cadenas" o "exploración de cadenas", lo que resulta sin sentido cuando incluso las compañías antivirus utilizan rutinariamente la expresión.

Una objeción al término es que perpetúa la noción anticuada del trabajo de los exploradores, si bien el mismo argumento podría ser aplicado para otros términos similares.

Las dificultades reales con el uso del término "exploración por firmas" se deben a que la misma:

- Perpetúa el mito de que es el único modo de detección realizado por los exploradores AV. De hecho, muchos virus no pueden ser identificados buscando solamente una cadena estática.
- Sugiere que existe una única secuencia de bytes en cada virus que es utilizada por todos los exploradores para identificarla. De hecho, diferentes exploradores pueden utilizar cadenas de búsqueda muy diferentes (y algoritmos) para detectar el mismo virus.

Algunas fuentes [10] han confundido la cuestión aún más al dar la impresión de que los exploradores buscan cadenas simples de texto más que secuencias de bytes. Generalmente, tal método no es confiable, es totalmente inefectivo con muchos tipos de malware, y de ineficiente programación. Además resulta fácilmente explotable por el escritor de virus –o, de hecho, todo aquel capaz de editar un archivo- y potencialmente peligroso para generar numerosos falsos positivos.

Los comodines (wildcards) y expresiones regulares de UNIX permiten mayor flexibilidad en la búsqueda de cadenas. En vez de buscar una cadena estática (una secuencia estructurada de bytes) el explorador reconoce una cadena asociada a determinado virus aún cuando otros bytes o secuencias de bytes (bytes ruidosos) son interpolados entre los elementos de la cadena. Un ejemplo simple de un byte ruidoso es la inserción de la instrucción NOP (No Operación), que no realiza ninguna función, excepto consumir tiempo de procesamiento sin realizar una operación real.

Estas mejoras a la exploración básica por cadenas permite la detección de virus cifrados y polimórficos [8]. Sin embargo, aún con este tipo de mejoras la exploración por cadenas no es particularmente eficiente cuando se trata de explorar en busca de múltiples virus y el advenimiento de virus polimórficos

complejos en verdad perjudicó a algunos exploradores que fueron incapaces de moverse hacia técnicas más avanzadas de detección. [8, 11]

El advenimiento de virus polimórficos complejos en verdad perjudicó a algunos exploradores que fueron incapaces de moverse hacia técnicas más avanzadas de detección.

El análisis algorítmico para virus específicos en las tecnologías AV actuales generalmente se basa en código interpretado que se ejecuta dentro de una máquina virtual. La virtualización y emulación puede utilizarse, por ejemplo, para remover ofuscamientos accidentales o intencionados tales como: empaquetamiento, compresión o cifrado. Una vez que el archivo resulta des-ofuscado, puede ser analizado algorítmica –o heurísticamente- mediante un proceso de exploración antivirus.

Las máquinas virtuales también juegan un papel importante en la implementación de análisis heurístico y puede ser muy exitoso a pesar de los numerosos problemas asociados con la emulación de un ambiente tan complejo como es el de Windows moderno [12]. (Sin embargo, resulta necesario comprender que la emulación puede no ser perfecta y la penalización por latencia –incremento del tiempo de procesamiento- puede ser considerable y varía según la particularidad del archivo que está siendo analizado).

Lo opuesto a la Heurística

El mito de que los AV comerciales sólo pueden detectar instancias conocidas, variantes y subvariantes de malware conocido está, tal vez, menos difundido de lo que solía estarlo. Sin embargo, ha sido en parte suplantado por el mito, menos dañino, de que los exploradores de virus específicos y los exploradores por heurística son dos tipos completamente diferentes de exploradores. De hecho, el análisis heurístico tal como lo conocemos ha sido utilizado durante más de una década, pero las técnicas heurísticas para optimizar el control de los virus han sido utilizadas desde mucho antes por los exploradores "de virus conocidos". También han tenido un lugar en las contramedidas afines, tales como bloqueadores y monitores de comportamiento y chequeadores de integridad.

En un sentido, lo opuesto al análisis heurístico en AV no es la exploración por firmas sino la exploración algorítmica, en la que la exploración por firmas es un caso especial de la misma.

En un sentido, lo opuesto al análisis heurístico en AV no es la exploración por firmas sino la exploración algorítmica, en la que la exploración por firmas es un caso especial de la misma.

La exploración algorítmica, tal como otras formas de codificación algorítmica, está basada en procedimientos matemáticos verificables [13]. Lo que es referido en la industria como exploración algorítmica, normalmente es entendido como algo que se basa en un algoritmo (más que simplemente buscar una cadena estática –

una secuencia estructurada de bytes) que es específico de acuerdo al virus que se intenta detectar.

Ciertamente, en la vida real, el análisis heurístico descrito anteriormente, también es considerado algorítmico en un sentido más general. Sin embargo la utilización del término algorítmico en el sentido especializado, específico del virus (y por ello, en cierto modo confuso) resulta lo suficientemente utilizado dentro de la industria [12] como para ser ignorado. La heurística es normalmente caracterizada como un algoritmo específico de acumulación de puntaje que determina la propensión del objeto explorado a ser malicioso, más que por la identificación inequívoca de un programa malicioso especifico.

Antivirus Genérico

Generalmente el análisis heurístico es considerado como mecanismo de detección genérico del antivirus, y no como mecanismo de detección de un virus específico. Lo que no siempre se tiene en cuenta es el hecho de que la inversa de esta idea también es cierta: las soluciones genéricas utilizan reglas heurísticas como parte de sus procesos de diagnósticos.

Las soluciones genéricas utilizan reglas heurísticas como parte de sus procesos de diagnósticos.

Por ejemplo:

- Los filtros de los gateway de email utilizan reglas para especificar qué tipos y nombres de archivos son permitidos como adjuntos. Tales filtros resultan muy buenos para mostrar amenazas obvias tales como archivos con extensiones del tipo .LNK, .JPG o .EXE pero pueden ser algo inflexibles en el rechazo de la totalidad de clases de los ejecutables1. Algunos filtros utilizan técnicas más avanzadas, tales como revisar que las cabeceras de los archivos explorados coincidan con la extensión del archivo. Esto puede reducir, significativamente, el riesgo de falsos positivos (y falsos negativos).
- Los detectores de cambios utilizan la regla de que si las características de un objeto han cambiado, debería ser tratado como sospechoso. Dado que existen tantos contextos en los que un binario puede legítimamente modificar su checksum (como por ejemplo códigos que se modifican a sí mismos, código recopilado, reconfiguración por compresión en tiempo de ejecución, programas emparchados o actualizados), una cruda modificación del criterio de detección (por ejemplo, si el archivo fue modificado entonces que está infectado) puede exhibir

[1] *¿Por qué estas amenazas son obvias? En primer lugar, porque el sufijo .LNK denota un atajo del programa que generalmente carece de sentido como adjunto de un email porque no hay un enlace directo entre el atajo y el programa al cual debería enlazarse; sin embargo, un archivo de enlace directo en el adjunto de un email suele ser simplemente un archivo ejecutable de Windows, redefinido para evadir filtros de archivos ejecutables adjuntos. En segundo lugar, la doble extensión sugiere un intento de hacer pasar un archivo ejecutable como uno no ejecutable (gráficos), un truco común de los escritores de virus.*

un alto grado de detección de falsos positivos. Sin embargo, la detección de cambios puede trabajar bien junto con la exploración de virus específicos.

Una técnica bien demostrada de esto es comparar un objeto con su checksum y ejecutar una exploración a fondo de la misma solo si la checksum previamente calculada ha sido modificada, reduciendo el tiempo que lleva procesar un archivo que no ha cambiado. Esta es la razón por la que la exploración inicial del sistema puede llevar más tiempo que exploraciones subsecuentes con algunos software antivirus.

- Los bloqueadores y monitores de comportamientos, que evalúan el modo en que se comportan las aplicaciones, se encontraban entre las formas más antiguas de software AV. El monitoreo de comportamientos clásico del AV tiende a revisar en busca de dos tipos de comportamientos del código: Réplica y Daño potencial.
    - Por definición, el código replicable sugiere fuertemente la presencia de un virus (o gusano, dependiendo en el tipo de código y la definición que prefiera). Este acercamiento tiene la ventaja de que las llamadas al sistema, que pueden ser códigos replicables, son comparativamente fáciles de identificar en forma programática; especialmente donde el código no está significativamente ofuscado. Sin embargo, es más fácil identificar un virus que se reproduce escribiendo una copia directa de sí mismo que una copia evolucionada de si (por ejemplo, virus polimórficos).
    - El código potencialmente dañino refleja la posibilidad de un payload malicioso. Este acercamiento es ineficiente allí donde no hay payload o donde el payload no es evidentemente dañino. Algunas formas de daño, tales como eliminación de archivos, son más simples de detectar programáticamente que otros, tales como los indeseados y embarazosos mensajes o imágenes ofensivos. Por otro lado, la detección exitosa mediante payload tiene una ventaja cuando se trata de detectar malware no replicativo (tales como troyanos y otros programas no virales). Sin embargo, esto requiere de cierta precaución. Por ejemplo, eliminar un archivo es, por sí mismo, un indicador no fiable de malicia, ya que muchos programas eliminan o sobre escriben rutinaria y legítimamente archivos tales como configuración obsoleta o datos de archivos.

Soy absolutamente Positivo

La identificación de virus es un balance entre dos imperativos: evitar falsos negativos (un fallo en la detección de una infección en donde esta existe) y falsos positivos (detección de un virus donde no existe). Tal como queda demostrado por la cantidad de falsos positivos en varios de los principales productos antivirus en los primeros meses del 2006, los

La identificación de virus es un balance entre dos imperativos: evitar falsos negativos (un fallo en la detección de una infección en donde esta existe) y falsos positivos (detección de un virus donde no existe).

avances en la optimización de la tecnología de exploración, estos no han eliminado el riesgo de falsos positivos.

La eliminación de falsos positivos no siempre resulta posible utilizando la tecnología heurística que, por definición, conlleva un cierto grado de ensayo y error. Tal como fue discutido anteriormente, el objetivo de la programación heurística no es producir el resultado "perfecto" sino uno "lo suficientemente bueno". Entonces, ¿cuál es el problema?

El método "más seguro" de identificar un virus conocido es buscar la presencia de cada byte de código de virus que debería estar presente en un objeto infectado, mediante la generación de una checksum de cada bit constante en el cuerpo del virus. Generalmente este proceso es definido como "identificación exacta".

La identificación es una medida de la habilidad del software AV para detectar y reconocer una muestra de virus como virus específico o como su variante. Consecuentemente, la identificación exacta denota un nivel de precisión donde cada byte constante de virus es tenido en cuenta. Mientras que suena deseable para que esta precisión sea aplicada a toda exploración de códigos maliciosos, en la vida real esto rara vez es realizado debido al impacto potencial en el tiempo de exploración y recursos del sistema y porque este nivel de detalle no siempre resulta necesario.

El término "definición casi exacta" se aplica si la identificación "sólo es lo suficientemente buena como para asegurar que el intento por remover un malware no devendrá en un daño al objeto que lo hospeda por el uso inapropiado de un método de desinfección" [2]. La detección y eliminación no siempre poseen el mismo problema. Algunas compañías AV han alegado hace tiempo que un programa binario infectado debería ser reemplazado en vez de limpiado, prefiriendo concentrarse en la detección. Además existen escenarios (rootkits y stealthkits son buenos ejemplos) donde la sustitución o un programa troyanizado por un programa legítimo implica que el software de seguridad sólo puede eliminar, pero no limpiar. En tales casos, generalmente se requiere del administrador o usuario para restaurar el programa legítimo; la restauración automática puede ser una opción no segura.

En los últimos años, el movimiento del malware ha estado alejado de la clásica infección parasitaria de archivos, a la manipulación del sistema operativo (por ejemplo, modificación del registro). Esto puede dificultar la remoción de todos los rastros del malware una vez que ha tomado el control. La eliminación incompleta (o incorrecta) puede dejar el sistema dañado o aún inutilizable, a veces requiriendo medidas radicales tales como la reinstalación del sistema operativo y software de aplicación y restauración de datos del backup.

Sin embargo, donde el malware es detectado proactivamente (por ejemplo, antes de que tenga la oportunidad de instalarse en el sistema al que apunta) por métodos heurísticos o genéricos,

generalmente este problema no surge a menos que el objeto malicioso (viral o troyanizado) sea requerido en una forma no infecciosa (como, por ejemplo, cuando el objeto contiene datos).

La "detección genérica" es un término aplicado cuando el explorador busca una cantidad de variantes conocidas utilizando una cadena de búsqueda capaz detectar todas las variantes. Si bien puede detectar una variante actualmente desconocida en la que aparece la misma cadena de búsqueda, sólo es una detección heurística si implica la utilización de un mecanismo por puntaje. De otro modo, es realmente un caso especial de detección de virus específico. Algunos sistemas utilizan un acercamiento híbrido, donde un sistema de puntaje es agregado a las capacidades de la detección genérica para dar una probabilidad de la variedad o parentesco familiar con diferentes grados de certeza. Por ejemplo, si la similitud es lo suficientemente cercana, el explorador puede reportar "variante de X", o si está menos seguro puede reportar "probablemente una variante de X".

La "detección genérica" es un término aplicado cuando el explorador busca una cantidad de variantes conocidas utilizando una cadena de búsqueda capaz detectar todas las variantes.

Sensibilidad y Diagnóstico erróneo

La precisión en el análisis heurístico depende de cuan agresivamente se configure el criterio de puntaje. Si el malware buscado es nuevo para el explorador, la precisión del resultado del analista depende de una simple decisión dual (o "si, es un malware conocido como XXX" o bien "no, no es un malware conocido"). La certeza de esta respuesta descansa en un rango de valores que va de alto (manteniendo el número de falsos positivos tan bajo como se pueda) a bajo (detectando tantos malware como sea posible). Una respuesta agresiva prioriza la detección de posibles malware por encima del riesgo de falsos positivos, mientras que una respuesta menos agresiva resulta más apropiada cuando el impacto desfavorable de falsa alarma resulta inaceptable.

La certeza de esta respuesta descansa en un rango de valores que va de alto (manteniendo el número de falsos positivos tan bajo como se pueda) a bajo (detectando tantos malware como sea posible).

No es extraño que un producto ofrezca la opción entre una configuración por defecto (desactivación de la heurística) o una configuración con heurística. Considerando que ya se ha señalado que todos los exploradores son, de alguna forma heurísticos, tal vez resultaría más apropiado referirse a la configuración por defecto como aquella que tiene habilitada la heurística básica. Algunos fabricantes también distinguen entre heurística pasiva y activa. En ambos casos, el código es explorado en busca de características sospechosas pero en el modo activo, el explorador utiliza un ambiente simulado para ejecutar y rastrear al código. En el modo pasivo, solo inspecciona estáticamente al código.

Uno de los modos de observar cómo la tecnología del explorador mapea el rango mencionado puede verse en las siguientes líneas:

Nivel del umbral	Nivel de Heurística correspondiente
Máximo	Identificación exacta (o casi exacta) solamente; la heurística no es utilizada o bien, mantenida al mínimo.
Normal	La detección de malware conocidos utilizando la exploración algorítmica y emulación como identificación apropiada y exacta (o casi exacta) donde sea necesaria. Probablemente con firmas genéricas para identificar de modo confiable variantes relativamente parecidas.
Modo Heurística	Nivel medio de heurística, detección enriquecida; relativamente bajo riesgo de falsos positivos, uso de análisis pasivo más que de heurística basada en la emulación.
Mínimo	Heurística más alta (avanzada o más sensible) incluyendo alguna forma de emulación. Alta proporción de nuevos malware detectados, pero incremento del riesgo de falsos positivos.

Ni todos los exploradores cuentan con estos niveles de sensibilidad, ni todos permiten la configuración o reconfiguración manual de los umbrales; y aquellos que soportan niveles de sensibilidad pueden no documentarlos. Además, debería enfatizarse que algunas formas de emulación podrían utilizarse en cualquiera de los niveles descriptos.

Los fabricantes que deshabilitan su heurística avanzada por defecto, pueden no solo estar intentando reducir el riesgo de falsos positivos sino que en realidad podrían estar intentando mejorar la velocidad percibida del producto. Todos los niveles de análisis heurístico repercuten en el procesamiento y el tiempo de análisis y para algunos productos un desempeño muy bajo puede resultar demasiado obvio.

Todos los niveles de análisis heurístico repercuten en el procesamiento y el tiempo de análisis y para algunos productos un desempeño muy bajo puede resultar demasiado obvio.

Sin embargo, tal como fue mencionado anteriormente, aún a medida que aumenta el número de objetos maliciosos conocidos, con rutinas de código bien implementadas y el poder de computación actual, el impacto puede ser reducido a un nivel manejable. De hecho, existe un amplio grado de variabilidad en términos de degradación del desempeño en velocidad entre los exploradores de diferentes casas antivirus. Un motor heurístico apropiadamente implementado solo debería tener un impacto mínimo en el desempeño del sistema.

La sensibilidad heurística no es solamente una cuestión técnica vinculada a la precisión para diagnosticar la presencia de códigos maliciosos previamente desconocidos. Además, es una cuestión psicosocial; ¿cómo señalarle un malware al usuario final y qué aviso se le debería dar?

El modo en que un posible malware es advertido dice mucho al cliente acerca de la casa antivirus. Algunos productos son cautos utilizando mensajes que dicen, efectivamente, que podría ser una variante del malware X, pero sin estar completamente seguros. Esto elimina el riesgo del fabricante de generar falsos positivos, dejando el diagnóstico final y elección al cliente.

En la realidad, la mayoría de los clientes preferiría que el diagnóstico fuese realizado por el explorador. Los usuarios podrían sentirse incómodos con la posibilidad de que el software se esté equivocando, lo que podría sugerir que la tecnología es menos confiable de lo que resulta ser.

Otras casas antivirus ofrecen un mensaje más detallado que dice algo así como “malware XXX detectado y bloqueado” o “W32/troyano-peligroso-de-puerta-trasera detectado y eliminado”. Eso suena muy bien y el cliente puede encontrarse agradecido que el malware ha sido identificado y neutralizado, pero inicialmente muchos pueden no saber que estos nombres son simples nombres genéricos que indican la detección heurística de un posible malware y no el indicativo de un malware específico.

Desafortunadamente, no existen estadísticas confiables que indiquen cuántos programas legítimos, emails y demás han sido considerados maliciosos a causa de un explorador demasiado confiado en sí mismo.

Algunos fabricantes advierten que la heurística avanzada sólo debería ser habilitada en contextos en los que se sospecha la presencia de un nuevo código malicioso o con mayor propensión a ser encontrado, (en exploradores de gateways de correo por ejemplo). Esto reduce la confusión causada por el riesgo de falsos positivos en el escritorio, pero incrementa el riesgo de falsos negativos donde falla el perímetro de exploración.

Cuestiones de la Evaluación

Examinar los exploradores del malware por su desempeño en la detección siempre ha sido una cuestión polémica [14] y sólo unos pocos evaluadores y grupos de evaluación son reconocidos como competentes en esta área por otros miembros de la comunidad de investigadores AV.

Sólo unos pocos evaluadores y grupos de evaluación son reconocidos como competentes en esta área por otros miembros de la comunidad de investigadores AV.

Las organizaciones evaluativas generalmente consideradas competentes en esta área son:

- AV Comparatives (http://www.av-comparatives.org/)
- AV-Test.org (http://www.av-test.org/)
- ICSA Labs (http://www.icsalabs.com/)
- SC Magazine/West Coast Labs (http://www.westcoastlabs.org/)
- Virus Bulletin (http://www.virusbtn.com/)
- Virus Research Unit, University of Tampere (http://www.uta.fi/laitokset/virus)
- Virus Test Center, University of Hamburg (http://agn-www.informatik.uni-hamburg.de/vtc/naveng.htm.)

Nota: últimas dos organizaciones no han estado evaluando activamente en la actualidad.

A diferencia de evaluadores sin vínculos con la comunidad de investigadores AV, estas organizaciones generalmente tienen la confianza de la comunidad –aunque no necesariamente por todos sus miembros- para evaluar competente, éticamente y con seguridad mientras se mantienen independientes. Este status de confiabilidad implica que a veces cuentan con acceso a muestras de malware autentificadas tales como aquellas recolectadas, evaluadas y autentificadas por la WildList Internacional Organization (http://www.wildlist.org/) un grupo de investigadores representantes de la mayoría de las principales casas AV y un gran número de corporaciones e instituciones educativas.

La comunidad AV argumenta que la mayoría de otras evaluaciones realizadas por aquellos fuera del grupo de evaluadores avalados por la industria son potencialmente inválidos o inapropiados porque:

- No puede asumirse la competencia del evaluador y, por tanto, tampoco se puede asegurar:
    - que la metodología evaluativa sea la apropiada
    - la adhesión a una práctica segura, ética y con los estándares de la industria

Debido a estas cuestiones, miembros de la comunidad de investigadores no pueden, éticamente, compartir muestras con evaluadores desconfiables. Debido a esto, no puede asumirse la procedencia y autenticidad de estas muestras contra las que los productos son evaluados. A veces, los evaluadores incapaces de acceder a las muestras de la comunidad AV intentan sustituirlas con muestras extraídas del intercambio de malware entre sitios web y otros recursos (potencialmente dudosos) que pueden contener todo tipo de muestras no maliciosas (archivos basura, muestras corrompidas y otros generalmente denominadas intendeds). Algunas de estas cuestiones pueden ser superadas si la organización evaluadora subcontrata el examen a una organización aceptada (por ejemplo, AV-Test realiza diversos tipos de evaluaciones para revisiones de distintas revistas).

Resulta curioso que estas dificultades han contribuido (pero no causado) con una situación en la que los evaluadores, preocupados por la efectividad de un determinado explorador contra códigos maliciosos desconocidos, evaluaban variantes mediante heurística aún antes que la tecnología adquiriese la etiqueta de heurística y las capacidades del siglo XXI. Desafortunadamente esto incluyó, de forma típica, la utilización de generadores desconfiables de malware, simuladores irrelevantes de malware, ubicación aleatoria o enmascaramiento del código malicioso y las cadenas de texto y más.

Claro que evaluar las capacidades heurísticas de un explorador es un objetivo perfectamente válido (especialmente ahora que los exploradores poseen capacidades heurísticas). Sin embargo, es tan importante para dicha evaluación que se realice competentemente y con seguridad como lo es para evaluar cualquier detección de malware conocido.

Ante la ausencia de un lineamiento de las bases de una configuración competentemente administrada de la evaluación, no existe garantía de que los exploradores están siendo evaluados contra códigos maliciosos válidos y en actividad.

Ante la ausencia de un lineamiento de las bases de una configuración competentemente administrada de la evaluación, no existe garantía de que los exploradores están siendo evaluados contra códigos maliciosos válidos y en actividad.

Los evaluadores cuya competencia aún es cuestionable debido a la falta de comunicación directa con la comunidad de investigadores AV, crean futuras dificultades para ellos mismos y para aquellos que confían en sus evaluaciones, si es que no publican información acerca de su metodología de evaluación, especialmente en cuanto a la validación de las muestras.

Por validación nos referimos a que el código utilizado en la evaluación es efectivamente malicioso –por ejemplo, un código malicioso debe tener la habilidad de reproducirse, los gusanos deben poder esparcirse correctamente y así sucesivamente. En ciertos casos en los que se realizan evaluaciones sin esta validación; luego se descubre que muchas de las piezas del código no eran maliciosas, sino más bien archivos dañados o archivos legítimos utilizados erróneamente.

En un ejemplo reciente [16], se sugirió que el grupo comisionado para realizar la evaluación utilizó generadores de malware. Inmediatamente, esto causó que los investigadores AV dudasen de la competencia del evaluador, debido a que los equipos de generadores de malware resultan notoriamente desconfiables cuando se trata de producir códigos maliciosos viables. Puesto que no describieron su metodología de evaluación detalladamente, no se dio a conocer cómo o si es que verificaron las muestras seleccionadas para la evaluación.

La posibilidad de que algunas o todas las muestras no fueran maliciosas, invalidan los exámenes de los exploradores antivirus si se asume que las muestras contenían códigos maliciosos. Si este es el caso, el índice más elevado de detección no necesariamente implica el mejor desempeño, debido a que puede incluir una amplia cantidad de falsos positivos [15], aún suponiendo que todos los exploradores examinados fueron consistentemente configurados.

La industria AV es reacia a consentir la creación de nuevos malware o códigos de virus, aún si es sólo para evaluaciones. Existen muchas razones para esta instancia: la adhesión de la mayoría de los investigadores a un código de ética restrictivo, preocupación acerca de cuestiones de seguridad cuando nuevos códigos maliciosos son manejados por evaluadores inexpertos, dificultades de validación y más. Dicho esto, no resulta necesario que se creen nuevos códigos maliciosos para evaluar la tecnología heurística.

Una "Evaluación Retrospectiva" implica la evaluación de un explorador que no ha sido actualizado por un período de tiempo determinado (tres meses es un período comúnmente elegido), con malware validado que ha aparecido desde la última actualización aplicada al explorador.

Una "Evaluación Retrospectiva" implica la evaluación de un explorador que no ha sido actualizado por un período de tiempo determinado (tres meses es un período comúnmente elegido), con malware validado que ha aparecido desde la última actualización aplicada al explorador. Esto provee una certeza razonable de que la capacidad heurística está siendo evaluada, no la detección de malware conocido mediante algoritmos de códigos maliciosos específicos. De ningún modo, tal evaluación disminuye la necesidad de una evaluación competente pero evita las dificultades éticas y prácticas asociadas con la creación de nuevos códigos maliciosos para propósitos evaluativos. Sin embargo, no elimina la necesidad de validar las muestras o de construir cuidadosamente evaluaciones significativas.

Casi todos los principales fabricantes proveen actualizaciones de detección diarias (o frecuentes), de modo que evaluar un explorador cuando se encuentra tres meses desactualizado no dice mucho acerca de sus capacidades de detección corrientes. Un acercamiento más válido podría ser el de evaluar las capacidades en diferentes puntos o evaluar con un código malicioso específico para determinar el primer punto en que ocurre la detección. Claramente, vale la pena notificar si un explorador fue capaz de detectar códigos maliciosos antes de que se supiese de su existencia.

Conclusión: Una Paradoja Heurística

Es interesante que, aunque la tecnología heurística es actualmente más sofisticada de lo que fue en la década de los 90, los índices de detección global han caído drásticamente mientras que los índices de detección de códigos maliciosos de "la vieja escuela" (macro virus, emails masivos y más) se mantienen impresionantemente altos.

Si bien se suele sugerir que esta declinación global se debe a la ineficiencia de la industria AV, o el deseo de atenerse a un modelo de detección de malware específico, esto no resulta cierto. Un factor mayor de contribución es la incrementada sofisticación de los autores de malware que han desarrollado una amplia variedad de acercamientos para minimizar la susceptibilidad de su producto a la detección heurística y que evalúan la efectividad de estos acercamientos contra exploradores apropiadamente actualizados y configurados. El problema es más molesto ahora de lo que fue años atrás cuando fue considerado (al menos por las casas fabricantes) algo así como una bonificación si un producto AV detectaba cualquier cosa más allá de virus.

Hoy en día, los virus (léase, programas con una identificada funcionalidad replicativa) constituyen por lejos la menor proporción de todos los programas maliciosos [17]. En cierto sentido, esto dificulta mucho el trabajo del explorador heurístico; es conceptualmente simple detectar un virus heurísticamente si se puede desenredar lo suficiente el código para determinar que está intencionado a replicarse si bien no siempre es técnicamente posible detectar un programa replicativo. Determinar automáticamente que un programa es un bot o un troyano de algún tipo o simplemente su intención maliciosa, es un desafío mucho mayor [5].

Tómese el siguiente ejemplo clásico: un programa que formatea un disco no es malicioso por definición – de hecho, esa podría ser su única función. Sin embargo, si es ejecutado debido a que el usuario fue engañado al creer que exhibirá una película o mejorará el acceso a Internet, es razonable considerarlo malicioso. En tal caso, el problema real descansa en establecer un algoritmo que discriminará en base al conocimiento del usuario acerca del propósito del programa y la intención del programador, más que de la característica de su programación.

Si no podemos establecer una heurística confiable para un acto malicioso o sus intentos, de todos modos podemos aplicar otras heurísticas y asignar un puntaje a un programa. Una estrecha semejanza, en términos de programación, a un malware conocido recibirá un puntaje más elevado. Existen muchos otros comportamientos que pueden alarmar, según el contexto, tómese por ejemplo la apertura de un canal SMTP o IRC o un mecanismo de transferencia de archivos. El análisis de archivos ejecutables puede señalar muchas rarezas del código, tales como parches sospechosos y combinaciones de la señal, inconsistencias en el encabezado, indicaciones de tamaños discordes y más. El contexto más amplio en el que un posible programa malicioso es encontrado también puede proveer de valiosas pistas acerca de su naturaleza. Los análisis de mensajes pueden indicar semejanzas con un mailer masivo conocido o con un

email que contiene un troyano, e incluso podría contener información útil como puede ser la contraseña para un archivo cifrado.

Si bien algunos exploradores tienen esta capacidad, podría ser demasiado optimista esperar que un explorador heurístico automáticamente examine en busca de una frase de acceso secreta (passphrase), especialmente en un mensaje con un alto porcentaje de contenido gráfico. Las chances de detectar dicho passphrase pueden ser más altas en un mensaje que se asemeja a otros mensajes maliciosos. Los mensajes que contienen programas maliciosos o URLs también pueden asemejarse a otros tipos del tráfico de mensajes maliciosos tales como phishing y spam –los autores de malware y spammers han ido tomando prestadas técnicas los unos de los otros por muchos años; las evidencias muestran una creciente confluencia de interés entre estos grupos antes dispares. A veces, se espera que los exploradores de email detecten estas y otras formas de abuso vía correo electrónico así como también malware puro. El análisis de tráfico puede mostrar patrones asociados con actividad maliciosa, tal como mailers masivos, spam y scam (engaño) generados por una botnet y más. Por estas razones, la exploración de los gateway para spam (heurísticas y otros) puede sumar un aporte considerable a la efectividad de la detección de malware.

Sin embargo, de ningún modo es cierto que se verán los mismos porcentajes altos de detecciones proactivas en un futuro cercano que los que se vieron en los primeros tiempos de la exploración heurística, por muy bienvenido que esto sea para los usuarios y las casas AV.

Los autores de malware tienen diferentes prioridades. Más que un enfoque expansivo (máximo esparcimiento de una variante singular), ahora su foco está puesto en frecuentes ráfagas cortas en términos de una instancia dada del malware, que puede estar dirigida a individuos o grupos específicos. Aún con cambios simples tales como cambios generados por empaquetadores para modificar las huellas (footprint) del programa pueden reducir la detección (heurística o no) y limitar los recursos aún en los más grandes laboratorios antimalware. Las formas de malware que suelen hacer uso frecuente de la tecnología botnet para actualizarse y modificarse a sí mismo una vez instalado en una máquina comprometida puede ser muy difícil de detectar.

No obstante, no hay por qué entrar en pánico, hemos vivido con estos problemas por varios años. La higiene en sentido común de la computadora, buenas prácticas de uso de los parches y actualizaciones frecuentes del antivirus continúan aportando una muy buena protección. No sólo eso; una virtualización crecientemente sofisticada y técnicas de emulación, junto con análisis heurístico se mantienen como componentes fuertes y en continuo perfeccionamiento de la industria de seguridad. Sin embargo, ni los fabricantes antivirus ni los partidarios de las tecnologías alternativas “sabor del mes” pueden asegurar en realidad, que son capaces de detectar todas las amenazas futuras proactivamente.

El truco está en mantener las expectativas en un nivel realista.

Acerca de los Autores

David Harley

David Harley ha estado investigando y escribiendo acerca de software malicioso y otras cuestiones en materia de seguridad desde fines de 1980. Entre el 2001 y 2006 trabajó en el Servicio de Salud Nacional del Reino Unido como National Infraestructure Security Manager, donde se especializó en el manejo de software malicioso y todas las formas de abuso vía correo electrónico y también dirigió el Centro de Asesoramiento en Amenazas. Desde abril del 2006, trabaja como autor independiente y consultante en seguridad tecnológica.

Su libro principal fue Virus Revealed de Harley, Slade y Gattiker: la exhaustiva guía Osborne para protección de computadoras contra códigos maliciosos. Ha contribuido con capítulos y edición de muchos otros libros en seguridad y educación, así como también en una multiplicidad de artículos y whitepapers de conferencia. Su último proyecto de escritura es como editor técnico y principal colaborador en The AVIEN Guide to Malware Defense, a publicarse en el 2007 por Syngress.

SMALL BLUE-GREEN WORLD
8 Clay Hill House, Wey Hill, Haslemere, SURREY GU27 1DA
Teléfono: +44 7813 346129
http://smallblue-greenworld.co.uk

Andrew Lee

Andrew Lee, CISSP, es Chief Research Officer en ESET LCC. Fue miembro fundador de la Antivirus Information Exchange Network, AVIEN, (Red de trabajo de Intercambio de Información Antivirus) y también reportero para la Organización Internacional de la WildList, un grupo que mantiene un listado de los códigos maliciosos para computadores que aún se encuentran activos In- the-Wild. Antes de incorporarse a ESET, dedicó su tiempo al manejo de defensas contra malware como administrador Senior en seguridad para una importante organización gubernamental del Reino Unido.

Andrew es el autor de numerosos artículos acerca de cuestiones referidas al malware y es un orador asiduo en conferencias y eventos incluyendo AVAR, EICAR y Virus Bulletin.

ESET, LLC
610 West Ash Street, Suite 1900, San Diego, California 92101, U.S.A.
Teléfono: +1.619.876.5400
Fax: +1.619.876.5845
http://www.eset.com

Referencias

[1] "A Short Course on Computer Viruses 2nd Edition", pp 2, 49 (Dr Frederick B Cohen): Wiley, 1994.
[2] "VIRUS-L/comp.viru Frequently Asked Questions (FAQ) v2.00" (N. FitzGerald et al., 1995): http://www.faqs.org/faqs/computer-virus/faq/ (Date of access 12th January 2007)
[3] "Analysis and Maintenance of a Clean Virus Library" (Dr. V. Bontchev): http://www.people.frisk-software.com/~bontchev/papers/virlib.html (Date of access 12th January 2007)
[4] "The Anti-Virus or Anti-Malware Test File": http://www.eicar.org/anti_virus_test_file.htm
[5] "Trojans" (Harley), in "Maximum Security 4th Edition" (ed. Anonymous): SAMS, 2003
[6] Oxford Compact English Dictionary, Oxford University Press: http://www.askoxford.com/ (Date of access 12th January 2007)
[7] Merriam-Webster Online: http://www.m-w.com/ (Date of access 12th January 2007)
[8] "Viruses Revealed" (Harley, Slade, Gattiker) pp158-159: Osborne 2001
[9] "Evolution Discussion Group Fall 1996 Phylogenies and Evolution, Useful Terms" - University of British Columbia Zoology Department: www.bcu.ubc.ca/~otto/EvolDisc/Glossary.html (Date of access 12th January 2007)
[10] "Virus Proof" (P. Schmauder), page 187: Prima Tech (2000)
[11] Dr. Solomon's Virus Encyclopaedia (Solomon, Gryaznov), pp30-31: S&S International (1995).
[12] The Art of Computer Virus Research and Defense (Szor), page 441, pp451-466: Addison-Wesley (2005).
[13] "Heuristic Programming": http://www.webopedia.com/TERM/h/heuristic_programming.html (Date of access 12th January 2007)
[14] Anti-virus programs: testing and evaluation (Lee): in "The AVIEN Guide to Malware Defense in the Enterprise" (Ed. Harley): Syngress (2007, in preparation).
[15] "AV Testing SANS Virus Creation" (Harley): Virus Bulletin pp6-7, October 2006
[16] "Consumer Reports Creating Viruses?" (Sullivan): http://redtape.msnbc.com/2006/08/consumer_report.html (Date of access 12th January 2006).
[17] "Email Threats and Vulnerabilities" (Harley). In "The Handbook of Computer Networks" (Ed. Bidgoli): Wiley (2007 – in press).

Glosario

Adware	Programa que realiza alguna acción (tal como mostrar una pantalla pop-up o enviar un navegador a algún sitio web) en la que llama la atención del usuario con alguna publicidad o producto.
Archivo Basura	En la investigación AV este archivo no es un programa malicioso, pero se encuentra incluido en colecciones con mantenimiento deficiente de malware como si así lo fuese.
Archivo de evaluación EICAR	Archivo de programa modificado en forma única, que muchos programas AV reconocen como un programa de evaluación y responden a él de una manera muy similar al modo en que responden frente a virus.
Auto-ejecutables	Término utilizado para describir software malicioso que no requiere de ninguna acción por parte de la víctima para esparcirse, dispararse, o ambos.
Cadena de exploración/de búsqueda	Secuencia de bytes encontrados en un virus conocido que no debería ser hallado en un programa legítimo. El término no se limita a cadenas de búsqueda estática y puede incluir comodines y expresiones regulares o el uso de otros algoritmos de detección de virus específicos. También conocida como Firma.
Caracter comodín (Wildcard)	Caracter que puede ser utilizado para representar otro caracter o secuencia de bytes o que indica el uso de una forma especializada de expresión regular.
Checksum	En este contexto, es un valor computado que es dependiente del contenido de un archivo específico. Si el contenido de ese archivo cambia, el checksum será cambiado. Algunos métodos de la checksum son propensos a colisiones (dos archivos generan el mismo checksum) esto es suficiente para la mayoría de los propósitos de revisión de integridad/cambio.
Corrupción	Daño que causa un mal funcionamiento o un funcionamiento nulo (en el contexto específico de un virus).
DDoS	Ataque de Negación del Servicio Distribuida (Distributed Denial of Service). Un atacante remoto utiliza zombies o agentes de un software maliciosamente instalado en una red de para atacar otros sistemas de modo que su funcionalidad resulte alterada.

Detección de virus específico	Detección de virus conocidos utilizando búsqueda de cadenas específicas de aquellos virus o sus variantes.
Detección/exploración Heurística	Reconocimiento de un objeto que tiene características virales o maliciosas suficientes para sugerir que probablemente es un virus u otro malware.
Dropper	Programa (generalmente no viral) que instala otros programas maliciosos como virus o gusanos.
Evaluación Retrospectiva	Una técnica para evaluar las capacidades heurísticas de un explorador o exploradores sin actualizarlo por un período determinado de tiempo y luego utilizándolo para explorar malware que ha aparecido seguido de la última actualización
Exploración de virus conocidos/de virus específicos	Exploración en busca de virus conocidos que deriva en la identificación por nombre de un virus encontrado dentro del ambiente explorado.
Falso Negativo	Describe el escenario donde un explorador antimalware falla en detectar malware actual.
Falso Positivo	Describe el escenario donde un explorador antimalware detecta malware incorrectamente donde no lo hay.
Firma	Sinónimo de “Cadenas de búsqueda”. Puede ser aplicado a una búsqueda de cadenas estáticas.
Frase de acceso secreta (Passphrase)	A diferencia de una contraseña que generalmente es una “palabra” o cadena, un passphrase es generalmente un grupo más amplio de palabras utilizado como forma más segura de contraseña.
Genérico	Describe los programas de seguridad que no reconocen amenazas específicas pero protegen utilizando un método que bloquea una clase completa (o varias) de amenazas. Una firma genérica es un caso específico de esto; una serie completa de variantes son detectadas y procesadas por una firma única y no por firmas individuales para cada variante. Antónimo de “Virus específico”.
Heurística Positiva	Regla o criterio que incrementa la posibilidad de que el objeto analizado sea viral o malicioso.

Identificación casi exacta	Reconocimiento de un virus donde la identificación es lo suficientemente buena para asegurar que un intento de remoción del virus no resultará en daños al anfitrión utilizando un método de desinfección inapropiada. Cada sección de las partes no modificables del cuerpo del virus no es identificado unívocamente.
Identificación Exacta	Reconocimiento de un virus donde cada sección de las partes no modificables del cuerpo de virus es unívocamente identificado.
Intendeds	Virus (o, en menor frecuencia, otros programas maliciosos) que no trabajan por alguna razón u otra, generalmente debido a testeos insuficientes por parte del autor.
Keylogger	Un programa que monitorea tipeos, generalmente instalado para propósitos maliciosos o criminales tales como el robo de contraseñas.
Programa generador de virus	Programa que no es un virus en sí mismo, pero que genera virus. También puede ser referido como "virus kit".
Programas Joke	Programa que realiza alguna acción inesperada que puede resultar molesta pero que no es destructiva. El límite entre un joke y un troyano puede ser muy tenue.
Rootkit	Un programa o serie de programas instalados subrepticiamente de modo de permitir acceso privilegiado o sin autorización a un sistema. A veces el término Stealthkit es utilizado; sin embargo esto puede denotar acceso no autorizado y sin privilegios. [Para más información se puede acceder al artículo: "¿La Raíz de todos los males? Rootkits Revelados", de David Harley & Andrew Lee.]
Spyware	Programa que recopila información acerca del usuario y se la pasa a la persona interesada en ella.
Troyano destructivo	Troyano que causa (en general, deliberadamente) daño directo, en oposición a algo menos dañino, como puede ser el robo de contraseñas u otros datos.
Zombie	Programa backdoor en una computadora comprometida que espera y actúa bajo instrucciones de una máquina remota.