SOPHOS

Los exploits a fondo:

Prevención de exploits completa

Los exploits aprovechan los puntos débiles de productos de software legítimos, como Adobe Flash y Microsoft Office, con el objetivo de infectar ordenadores con fines delictivos. Los ciberdelincuentes los suelen utilizar para penetrar las defensas de las empresas. Los objetivos de estos criminales son diversos: robar datos o tomarlos como rehenes, realizar operaciones de reconocimiento o simplemente como medio para distribuir más malware tradicional.

La explotación de vulnerabilidades suele formar parte de los ciberataques: en más del 90 % de las filtraciones de datos registradas, se utiliza esta explotación en uno o más puntos de la cadena de ataque. Incluir la prevención de vulnerabilidades como parte de un conjunto exhaustivo de defensas de seguridad tiene un valor muy claro.

Las vulnerabilidades han estado presentes durante más de 30 años, así que no es ninguna sorpresa que prácticamente todos los proveedores de seguridad importantes afirmen ofrecer algún nivel de prevención de vulnerabilidades. Sin embargo, la amplitud y el alcance de esta protección varía notablemente de un proveedor a otro. En algunos casos, es solo una función más; en otros, se trata de uno de los principales ejes. Lea este monográfico para conocer mejor los exploits y los distintos niveles de prevención que ofrecen destacados productos de seguridad.

Contenido

La industria de los exploits: el software delictivo como servicio

Gracias a los kits de exploits, los autores de malware no tienen que preocuparse por cómo encontrar errores en Java, Silverlight o Flash, cómo integrar esos errores en exploits activos, cómo encontrar servidores web no seguros para alojar los exploits o cómo atraer víctimas a las páginas web cargadas con código malicioso.

En la misma medida, los autores de kits de exploits no tienen que preocuparse por escribir programas maliciosos completos, no tienen que tener servidores para el seguimiento de los ordenadores infectados ni recaudar dinero de sus víctimas, y tampoco tienen que involucrarse en la exfiltración de datos robados ni en la venta de estos.

Hoy en día, se han industrializado todos los aspectos de un ataque, siendo los ciberdelitos una industria que mueve miles de millones de dólares y que se estima que provoque daños por valor de 2 billones de dólares de aquí al 2019.

Los criminales pueden permitirse el lujo de poder especializarse en uno o más ámbitos del panorama de las amenazas en lo que se conoce en broma como "software delictivo como servicio" (o CaaS, por sus siglas en inglés).

En esta industria tan lucrativa hoy en día, han aparecido los brókeres de exploits: compran exploits a los que los descubren y los venden a otros que quieran utilizarlos, ya sean agencias gubernamentales o hackers infames.

Los compradores nunca revelan sus objetivos. Tal como Kevin Mitnick, fundador de Mitnick's Absolute Zero Day Exploit Exchange, explicó a Wired: "Cuando tenemos un cliente que quiere una vulnerabilidad de día cero por el motivo que sea, no preguntamos, y de hecho, tampoco nos lo contaría. Investigadores las encuentran, nos las venden por X, nosotros las vendemos por Y, y nos quedamos con el margen".

"Cuando tenemos un cliente que quiere una vulnerabilidad de día cero por el motivo que sea, no preguntamos, y de hecho, tampoco nos lo contaría. Investigadores las encuentran, nos las venden por X, nosotros las vendemos por Y, y nos quedamos con el margen."

Kevin Mitnick

Técnicas de mitigación de exploits

Teniendo en cuenta que cada día se crean más de 400 000 muestras de malware únicas y que se descubren miles de vulnerabilidades nuevas cada año, el reto de prevenir ataques maliciosos es abrumador. Esta explosión de crecimiento de las variantes de malware requiere enfoques nuevos e innovadores a la hora de protegerse contra los ciberdelincuentes.

Un examen detallado de la industria moderna del cibercrimen muestra una oportunidad de defensa asimétrica. Resulta que, a pesar del aparentemente interminable desfile de ataques nuevos, solo hay alrededor de una veintena de técnicas que pueden usarse para aprovechar las vulnerabilidades de software. Así que un enfoque que pueda neutralizar este puñado de técnicas de explotación, en vez de dirigirse a todos y cada uno de los exploits, es sumamente potente.

Lo que es más: dependiendo de la vulnerabilidad, los atacantes a menudo acaban juntando unas cuantas técnicas de explotación para llegar al punto en el que puedan distribuir el malware. Estas técnicas no cambian mucho de un año a otro: quizá se añadan uno o dos trucos a la lista de técnicas disponibles.

A la hora de evaluar los principales productos de seguridad, puede sorprender la falta de técnicas significativas de mitigación de exploits. Y aunque algunos de los fabricantes más nuevos que afirman ofrecer tecnología de última generación cuentan con un soporte más amplio en materia de mitigación de exploits, incluso aquí la cobertura es irregular.

A continuación figura una lista de mitigaciones de exploits cuyo objetivo es eliminar toda clase de vulnerabilidades y doblegar las técnicas de explotación utilizadas por ciberdelincuentes y agencias gubernamentales. Las mitigaciones de cada técnica varían en función del fabricante. Es importante saber que cuando un proveedor afirma evitar los exploits, la mayoría simplemente protege contra una pequeña parte de los métodos de explotación usados habitualmente y las mitigaciones a menudo no son efectivas en aplicaciones de 64 bits. Solo Sophos ofrece una prevención de exploits realmente exhaustiva.

Aplicación de la prevención de ejecución de datos (DEP)

La prevención de ejecución de datos (DEP) es un conjunto de tecnologías de hardware y software que realizan comprobaciones adicionales en la memoria para evitar desbordamientos del búfer. Sin DEP, un atacante puede tratar de explotar una vulnerabilidad de software saltando al código malicioso (shellcode) en una ubicación de la memoria donde residen los datos controlados por el atacante, como la pila o el montón. Sin DEP, estas regiones suelen estar marcadas como ejecutables, de modo que el código malicioso podrá ejecutarse.

DEP es una opción activable en Windows XP y versiones posteriores que debe ser configurada por el fabricante de software al crear una aplicación. Además, hay ataques disponibles para sortear la protección DEP integrada y, por lo tanto, no se recomienda la dependencia de la implementación del sistema operativo.

Selección aleatoria del diseño del espacio de direcciones (ASLR) obligatoria

Algunos exploits atacan las ubicaciones de la memoria que se sabe están asociadas a procesos concretos. En versiones antiguas de Windows (incluido Windows XP), los procesos base solían cargarse en ubicaciones predecibles de la memoria al iniciar el equipo. La selección aleatoria del diseño del espacio de direcciones (ASLR) aleatoriza las ubicaciones de la memoria que utilizan los archivos del sistema y otros programas, por lo que un atacante lo tiene mucho más difícil para acertar correctamente la ubicación de un proceso determinado, como la base del ejecutable y la posición de la pila, el montón y las bibliotecas.

ASLR solo está disponible en Windows Vista y posterior y, al igual que DEP, debe ser configurada por el fabricante de software al crear una aplicación. Asimismo, como DEP, hay ataques disponibles para sortear la protección ASLR integrada y, por lo tanto, no se recomienda la dependencia de la implementación del sistema operativo.

ASLR de abajo a arriba

Si se activa, la ASLR de abajo a arriba mejora la entropía o aleatoriedad de la ASLR obligatoria.

La principal ventaja de la ASLR obligatoria y de la ASLR de abajo a arriba en Sophos Intercept X es que las direcciones base de las aplicaciones no solo se aleatorizan en cada reinicio, sino también cada vez que se inicia la aplicación protegida.

Página NULL (Protección de desreferencia NULL)

A partir de Windows 8 y posterior, Microsoft niega a los programas la habilidad de adjudicar o asignar la "página NULL" (memoria que reside en la dirección virtual 0x00000000 del espacio de direcciones). Al hacerlo, Microsoft mitiga eficazmente la explotación directa de toda una clase de vulnerabilidades llamadas vulnerabilidades "de desreferencia a puntero NULL".

En Windows XP, Windows Vista y Windows 7, la explotación de tal defecto permitiría al atacante ejecutar código en el contexto del kernel (en el nivel de privilegio de CPU ring0), lo que conllevaría un aumento de privilegios a uno de los niveles más altos. Este tipo de vulnerabilidades permiten que los atacantes accedan a prácticamente todas las partes del sistema operativo.

Preasignación de la pulverización del montón

Heap Spray es una técnica que en realidad no explota vulnerabilidades, sino que se utiliza para facilitar la explotación de una vulnerabilidad. Utilizando una técnica llamada Heap Feng Shui[1], un atacante puede colocar en el montón shellcode o estructuras de datos dirigidos, permitiendo así la explotación fiable de una vulnerabilidad de software.

Una mitigación de la pulverización del montón típica implica reservar o preasignar direcciones de memoria usadas habitualmente para evitar que se utilicen para alojar cargas. Los atacantes más creativos conocen estas direcciones, así que en el caso de un ataque del mundo real, esta mitigación resulta poco eficaz. La preasignación de la pulverización del montón, también conocida como preasignación de shellcode o imposición contra la pulverización del montón, suele ser eficaz contra exploits predeterminados utilizados por organizaciones de evaluación.

Pulverización dinámica del montón

Comparada con la preasignación de la pulverización estática del montón, la mitigación de la pulverización dinámica del montón suele desencadenarse por un incremento repentino del consumo de memoria.

De hecho, la mitigación de la pulverización dinámica del montón analiza el contenido de las asignaciones de memoria recientes para detectar patrones que indiquen pulverizaciones del montón que contengan rampas NOP, rampas NOP polimórficas, matrices en JavaScript y otras secuencias sospechosas que se colocan en el montón para facilitar ataques de exploits.

Eje de la pila

La pila de una aplicación es un área de la memoria que contiene, entre otras cosas, una lista de ubicaciones de direcciones de la memoria (las llamadas direcciones de devolución). Estas ubicaciones contienen el código propiamente dicho que el procesador necesita ejecutar en un futuro cercano.

Los ejes de pila son muy utilizados por explotaciones de vulnerabilidades para sortear protecciones como DEP, por ejemplo encadenando fragmentos de código (gadgets) ROP en un ataque de programación orientado a retorno. Con los ejes de pila, los ataques pueden pasar de la pila real a una nueva pila falsa que podría ser un búfer controlado por el atacante como el montón, desde el cual los delincuentes pueden controlar el flujo futuro de la ejecución de los programas.

1 https://cansecwest.com/slides/2014/The%20Art%20of%20Leaks%20-%20read%20version%20-%20Yoyo.pdf

Ejecución de la pila (MemProt)

En circunstancias normales, la pila contiene datos y direcciones que apuntan a un código para que lo ejecute el procesador en un futuro próximo. Usando un desbordamiento del búfer de la pila[2], los atacantes pueden sobrescribir la pila con código arbitrario. Para permitir que este código se ejecute en el procesador, el área de la memoria de la pila debe hacerse ejecutable para sortear DEP. Una vez que la memoria de la pila sea ejecutable, el atacante lo tiene muy fácil para proporcionar y ejecutar código de programación.

Mitigación de ROP basada en pilas (Autor de llamada)

Para superar tecnologías de seguridad como la prevención de ejecución de datos (DEP) y la selección aleatoria del diseño del espacio de direcciones (ASLR), los delincuentes suelen recurrir al ataque del flujo de control de aplicaciones vulnerables conectadas a Internet. Estos ataques en memoria son invisibles para los antivirus, la mayoría de los productos de "última generación" y otras ciberdefensas dado que no intervienen archivos maliciosos. En su lugar, el ataque se construye durante el tiempo de ejecución combinando breves fragmentos de código benigno que forman parte de aplicaciones existentes como Internet Explorer y Adobe Flash Player; es lo que se conoce como ataque de reutilización de código o de programación orientado a retorno (ROP).

Durante un flujo de control normal, la instrucción CALL invoca las funciones de API sensibles, como VirtualAlloc y CreateProcess. Al invocar una API sensible, las defensas de ROP habituales detienen la ejecución de código para determinar la dirección de invocación de la API, usando la dirección de "retorno" que se ubica en la parte superior de la pila. Si la instrucción de la dirección de invocación de la API no es CALL, el proceso finaliza.

Dado que el contenido de la pila es editable, un atacante puede modificar valores concretos en la pila para engañar al análisis de la defensa de ROP basada en pilas, que no puede determinar si el contenido de la pila es benigno o ha sido manipulado por un delincuente.

Mitigación de ROP basada en ramas (Integridad de flujo de control aumentada por hardware)

Como ya se ha explicado, las defensas basadas en pilas contra los ataques de programación orientados a retorno (ROP) son generales y proclives a ser manipuladas. Para mejorar esto, los defensores de la seguridad necesitan datos más detallados y a prueba de manipulaciones para analizar en tiempo de ejecución.

Sophos Intercept X introduce la integridad de flujo de control (CFI) aumentada por hardware al aprovechar una función no utilizada de hardware en los procesadores Intel® convencionales (de 2008 y más recientes). El propio hardware del procesador ofrece datos de solo lectura para incrementar la detección de ataques de exploits sofisticados en tiempo de ejecución. Utilizar registros con seguimiento realizado por hardware (ramas) ofrece una ventaja considerable respecto a los enfoques de software basados en pilas. La información de rama que puede obtenerse de estos registros no solo identifica el objetivo de la rama, sino también el origen. Así, de hecho muestra dónde se originó el cambio en el flujo de

2 https://en.wikipedia.org/wiki/Stack_buffer_overflow

control. Esta información específica no puede obtenerse con el mismo grado de confianza utilizando una solución basada en pilas, como Microsoft EMET o Palo Alto Networks Traps.

La información de rama en los registros con seguimiento realizado por hardware no se puede manipular; no hay forma de que un atacante la sobrescriba con datos controlados. Los enfoques basados en pilas se basan en los datos de pilas, que se encuentran bajo el control del atacante (sobre todo en el caso de un ataque ROP), que a su vez puede engañar al defensor. En cambio, los datos con seguimiento realizado por hardware examinados por Sophos Intercept X son más fiables y resistentes a las manipulaciones.

Una implementación alternativa de integridad de flujo de control aumentada por hardware (HA-CFI) de Endgame se basa en enseñar al flujo de control regular a detectar cualquier desvío de la ruta de código prevista por el programador. Debe ser entrenado continuamente para crear una lista blanca de direcciones válidas de puntero de código que reflejen todas las posibles funciones y versiones de la aplicación protegida. Sophos Intercept X no requiere entrenamiento y, además, funciona correctamente durante cambios de contexto de subprocesos y escalado dinámico de frecuencia.

Sophos Intercept X utilizará de forma automática el seguimiento de flujo de control aumentado por hardware cuando detecte un procesador (CPU) Intel® Core™ i3, i5 o i7. Si el hardware del procesador no es compatible, Sophos Intercept X recurrirá automáticamente a las comprobaciones de integridad basadas en pilas solo de software.

Sophos Intercept X no solo aprovecha los registros con seguimiento realizado por hardware para incrementar la detección de ROP, sino que también se utiliza para el filtrado de direcciones de importación (IAF) para proteger la tabla de direcciones de importación de las aplicaciones protegidas.

Nota: los parches para corregir las vulnerabilidades Spectre acerca del predictor de ramas dentro del hardware de CPU de Intel no afectan al buen funcionamiento de Sophos Intercept X.

Protección de sobrescritura del controlador de excepciones estructurado (SEHOP)

Un atacante puede sobrescribir, con un valor controlado, el puntero del controlador de un registro de excepciones en la pila. Cuando se dé una excepción, el sistema operativo pasará por la cadena del registro de excepciones y llamará a todos los controladores de cada registro de excepciones. Dado que el atacante controla uno de los registros, el sistema operativo saltará a donde quiera el atacante, dándole el control del flujo de ejecución.

SEHOP es una opción activable en Windows Vista y posterior y debe ser configurada por el fabricante de software al crear una aplicación. Hay ataques disponibles para sortear la protección SEHOP integrada y, por lo tanto, no se recomienda la dependencia de la implementación del sistema operativo.

Filtrado de acceso de tabla de direcciones de importación (IAF)

En última instancia, un atacante necesita las direcciones de funciones concretas del sistema (p. ej. kernel32!VirtualProtect) para poder llevar a cabo actividades maliciosas. Estas direcciones pueden obtenerse de distintas fuentes y una de ellas es la tabla de direcciones de importación (IAT) de un módulo cargado. La IAT se utiliza como tabla de búsqueda cuando una aplicación llama a una función de otro módulo. Dado que un programa compilado no puede saber la ubicación de la memoria de las bibliotecas de las que depende, se requiere un salto indirecto siempre que se realice una llamada a API. A medida que el enlazador dinámico carga módulos y los une, escribe direcciones reales en las ranuras de IAT para que apunten a las ubicaciones de la memoria de las funciones de la biblioteca correspondientes.

Sophos Intercept X introduce el filtrado de acceso de tabla de direcciones de importación aumentado por hardware al aprovechar funciones no utilizadas de hardware en los procesadores Intel® convencionales (de 2008 y más recientes). Además de los registros de ramas con seguimiento realizado por hardware para imponer la integridad de flujo de control, Sophos Intercept X también se sirve de la predicción de ramas de hardware para mejorar aún más la protección de la tabla de direcciones de importación.

Nota: los parches para corregir las vulnerabilidades Spectre acerca del predictor de ramas dentro del hardware de CPU de Intel no afectan al buen funcionamiento de Sophos Intercept X.

Carga de bibliotecas

Los atacantes pueden cargar bibliotecas maliciosas colocándolas en rutas UNC. Se puede usar la supervisión de todas las llamadas a la API LoadLibrary para impedir este tipo de carga de bibliotecas.

Inyección de DLL reflectiva

Generalmente, al cargar una DLL en Windows, se llama a la función LoadLibrary de API. LoadLibrary toma la ruta de archivo de una DLL como entrada y la carga en la memoria.

La inyección de DLL reflectiva hace referencia a cargar una DLL desde la memoria en lugar de desde el disco. Windows no tiene una función LoadLibrary que admita esto, de modo que para conseguir esta funcionalidad uno debe crear la suya propia. Una ventaja de escribir nuestra propia función es que podemos omitir algunas de las cosas que suele hacer Windows, como registrar la DLL como un módulo cargado en el proceso, lo que hace más escurridizo el cargador reflectivo al ser investigado. Meterpreter es un ejemplo de herramienta que utiliza la carga reflectiva para ocultarse. La mitigación se realiza analizando si una DLL se carga de forma reflectiva en la memoria.

Shellcode

Un shellcode es un pequeño fragmento de código que se utiliza como carga en la explotación de una vulnerabilidad de software. Se denomina "shellcode" porque históricamente iniciaba un shell de comandos desde el que el atacante podía controlar el equipo afectado, pero cualquier fragmento de código que realiza una tarea similar puede llamarse shellcode.

Normalmente, un exploit inyectará un shellcode en el proceso de destino antes o al mismo tiempo que explota una vulnerabilidad para hacerse con el control del puntero de instrucción del procesador (EIP/RIP). El puntero de instrucción se ajusta para apuntar al shellcode, tras lo cual se ejecuta y lleva a cabo su tarea.

Modo Dios de VBScript

En Windows, VBScript puede utilizarse en navegadores o en el shell local. Al usarse en el navegador, las capacidades de VBScript están limitadas por motivos de seguridad. Esta restricción está controlada por la bandera de modo seguro. Si esta bandera se modifica, VBScript en HTML puede hacerlo todo como si estuviera en el shell local. Por lo tanto, los atacantes pueden escribir fácilmente código malicioso en VBScript. Manipular la bandera de modo seguro en VBScript en el navegador web se conoce como "modo Dios"[3].

A modo de ejemplo, un atacante puede modificar el valor de la bandera de modo seguro sirviéndose de la vulnerabilidad CVE-2014-63324[4], un fallo causado por una manipulación inapropiada al cambiar el tamaño de una matriz en el motor VBScript de Internet Explorer. En el modo Dios, el código arbitrario escrito en VBScript puede salir del espacio seguro del navegador. Gracias al modo Dios, las protecciones de la prevención de ejecución de datos (DEP), la selección aleatoria del diseño del espacio de direcciones (ASLR) y el flujo de control (CFG) no están activas.

WoW64

Microsoft ofrece compatibilidad con versiones anteriores para software de 32 bits en ediciones de 64 bits de Windows a través de la capa "Windows sobre Windows" (WoW). Hay aspectos de la implementación WoW que proporcionan a los atacantes interesantes medios para complicar el análisis dinámico, el desempaquetado binario y sortear las mitigaciones de exploits.

El comportamiento de una aplicación de 32 bits en el entorno WoW64 es distinto en muchos sentidos de un sistema de 32 bits real. La habilidad de cambiar entre modos de ejecución durante el tiempo de ejecución proporciona al atacante métodos de explotación, ofuscación y antiemulación, como por ejemplo:

- Aparatos ROP adicionales que no están presentes en el código de 32 bits
- Codificadores de carga de modo de ejecución mixto
- Funciones de ejecución del entorno que pueden limitar la eficacia de las mitigaciones
- Sortear enlaces insertados por software de seguridad, solo en el espacio del usuario de 32 bits

3 https://en.wikipedia.org/wiki/Glossary_of_video_game_terms#God_mode
4 https://www.rapid7.com/db/modules/exploit/windows/browser/ms14_064_ole_code_execution

La mayoría de programas de protección para endpoints solamente enlazarán funciones de API sensibles con el espacio de memoria del usuario de 32 bits si un proceso se está ejecutando en WoW64. Si un atacante consigue cambiar al modo de 64 bits, logra acceso a las versiones de 64 bits no enlazadas de las funciones de API sensibles enlazadas con el modo de 32 bits.

En las ediciones de 64 bits de Windows, Sophos Intercept X prohíbe al código del programa cambiar directamente del modo de 32 bits al de 64 bits (por ej. usando ROP), si bien sigue permitiendo que la capa WoW64 realice esta transición.

Para obtener más información sobre el abuso de WoW64, consulte la investigación llevada a cabo por Duo Security: WoW64 and So Can You[5] y Mitigating Wow64 Exploit Attacks[6].

Syscall

Una llamada del sistema o syscall es la forma programática en la que un programa informático solicita un servicio del kernel del sistema operativo. Aquí se incluyen servicios relacionados con el hardware, como acceder al disco local y crear y ejecutar procesos nuevos.

Generalmente, el sistema operativo proporciona una interfaz de programación de aplicaciones (API) genérica que se sitúa entre los programas normales y el sistema operativo. En circunstancias normales, una aplicación siempre llamará a una API para solicitar una tarea concreta del kernel. El software de seguridad coloca enlaces en las funciones de API sensibles para interceptar y llevar a cabo comprobaciones como el escaneado antivirus antes de permitir al kernel encargarse de la solicitud.

Un atacante puede aprovecharse del hecho de que:

- No todas las funciones de API están enlazadas por el software de seguridad, solo las funciones sensibles.
- Los códigos auxiliares que se usan para llamar a las funciones del kernel son muy parecidos; solo el índice de la función es único.

Al llamar a un código auxiliar de funciones no sensibles sin supervisar en un desplazamiento (para dirigirse deliberadamente a un servicio de kernel sensible en su lugar), un atacante puede esquivar la mayoría de programas de seguridad o análisis de espacio seguro.

Sophos Intercept X incluye un enfoque novedoso para evitar que los atacantes se dirijan a funciones de kernel sensibles a través de funciones de API que no están protegidas.

Para obtener más información sobre los ataques a las llamadas del sistema, consulte la entrada de blog "Defeating Antivirus Real-time Protection From The Inside" en BreakDev.org[7].

Vaciado de procesos

El vaciado de procesos es una técnica en la que una aplicación de confianza (como explorer.exe o svchost.exe) se carga en el sistema con el único objetivo de hacer de contenedor de un código malicioso. Un proceso vacío suele crearse en estado suspendido; a continuación, se anula la asignación de su memoria y se sustituye por código malicioso. Al igual que la inyección de código, la ejecución del código malicioso se oculta en un proceso legítimo y puede eludir las defensas y el análisis de detección.

5 https://duo.com/blog/wow64-and-so-can-you
6 https://hitmanpro.wordpress.com/2015/11/10/mitigating-wow64-exploit-attacks
7 https://breakdev.org/defeating-antivirus-real-time-protection-from-the-inside

Duplicación de procesos

La mayoría de equipos Windows utilizan el sistema de archivos NTFS. En 2007, Microsoft introdujo una nueva función denominada NTFS transaccional (TxF). Esta función permite que múltiples operaciones de archivos se traten como un todo: pueden realizarse correctamente como un todo y confirmarse o bien fallar como un todo y revertirse. De este modo, una aplicación puede realizar muchos cambios en muchos archivos en el disco y revertirlos al estado original si se detecta un error. El uso más común de TxF es durante la instalación de las actualizaciones de Windows.

La duplicación de procesos explota el mecanismo de TxF para ocultar malware. Elige un archivo inocente, lo sobrescribe y ejecuta el malware a través de una API de perfil bajo para, por ejemplo, hacerse pasar por un archivo de confianza (similar al vaciado de procesos). Antes de ejecutar el malware, rechaza o revierte todos los cambios, evitando así que el software antivirus escanee el contenido del archivo que realmente se ejecuta. Si se abre, el archivo en el disco no tendrá contenido sospechoso. Además, este archivo puede ser una aplicación conocida firmada digitalmente.

Secuestro de DLL

Debido a una vulnerabilidad comúnmente conocida como secuestro de DLL, suplantación de DLL, precarga de DLL o siembra binaria, muchos programas cargarán y ejecutarán una DLL maliciosa incluida en la misma carpeta que un archivo de datos abierto por esos programas.

Intercambio dinámico de datos (DDE)

El intercambio dinámico de datos (DDE) de Windows es un protocolo cliente-servidor para la comunicación entre procesos (IPC) entre aplicaciones. Los atacantes pueden utilizar DDE para ejecutar comandos arbitrarios. Por ejemplo, los documentos de Microsoft Office pueden infectarse con comandos DDEAUTO y utilizarse para distribuir la ejecución de comandos de PowerShell a través de campañas de spear phishing o contenido web alojado, evitando el uso de macros de Visual Basic para Aplicaciones (VBA). También es posible incrustar comandos DDEAUTO en el cuerpo del mensaje de correos electrónicos o convocatorias de reunión, que se ejecutan al responderlos o aceptarlos en Microsoft Outlook.

Gracias al diseño de la mitigación del bloqueo de aplicaciones, Sophos Intercept X también evita la ejecución de código malicioso a través del intercambio dinámico de datos.

Bloqueo de aplicaciones

En caso de que un atacante explote y sortee con éxito todas las mitigaciones de memoria y de código, Sophos Intercept X limita las capacidades del atacante. Esta función, llamada bloqueo de aplicaciones, tiene como objetivo evitar que los delincuentes introduzcan código no solicitado.

El bloqueo de aplicaciones detiene los ataques que no suelen depender de los defectos de software en las aplicaciones. Un ataque de este tipo podría ser, por ejemplo, el uso de una macro maliciosa en un documento de Office adjunto de un mensaje de correo electrónico de (spear) phishing. Las macros en documentos son potencialmente peligrosas ya que se crean en el lenguaje de programación VBA (Visual Basic for Applications), que incluye la capacidad de descargar y ejecutar binarios desde Internet y también permite el uso de PowerShell y otras aplicaciones de confianza.

Esta función inesperada (o exploit de fallo de lógica) ofrece a los atacantes una ventaja obvia puesto que no tienen que explotar un defecto de software o hallar una forma de sortear las defensas de código y memoria para infectar equipos. Simplemente se aprovechan de las funcionalidades estándar que ofrece una aplicación de confianza de uso generalizado y solo tienen que utilizar técnicas de ingeniería social para convencer a la víctima de que abra el documento especialmente diseñado.

Sin que sea necesario mantener una lista negra de carpetas, Sophos Intercept X finalizará automáticamente una aplicación protegida según su comportamiento; por ejemplo, cuando una aplicación de Office se utiliza para iniciar PowerShell, acceder a WMI, ejecutar una macro para instalar código arbitrario o manipular áreas críticas del sistema, Sophos Intercept X bloqueará la acción maliciosa, incluso aunque el ataque no inicialice un proceso secundario.

Bloqueo de Java

Anteriormente, los kits de explotación eran instrumentos básicos de las descargas no autorizadas de malware. Se aprovechaban de vulnerabilidades en Java Runtime Environment (JRE) para entregar una carga de Windows PE. JRE se carga como un complemento en los navegadores convencionales.

Sophos Intercept X evita que JRE ejecute aplicaciones que no sean Java. Por ejemplo, Sophos Intercept X finalizará una aplicación Java cuando intente introducir y ejecutar un archivo binario de Windows PE. Además, los atacantes no pueden abusar de Java para manipular ubicaciones de inicio automático, como la carpeta de inicio, Run, RunOnce y otras claves del registro.

Nota: con la introducción de la actualización 20 de Java 8 en 2014, el nivel de seguridad para las aplicaciones Java está establecido en Alto de forma predeterminada. Esto ha hecho que los atacantes cada vez lo tengan más difícil para ejecutar exploits de Java con permisos suficientes para infectar el endpoint. Por consiguiente, los exploits de Java ya no son un favorito de los kits de explotación, por lo que la mitigación del bloqueo de Java ha quedado algo obsoleta.

Cueva de código

El uso de cuevas de código es una técnica utilizada por adversarios en la que estos modifican lo que probablemente es software legítimo de modo que contenga una aplicación adicional. Esta aplicación adicional se inserta en lo que se denomina cueva de código, una sección del archivo de la aplicación de destino que no utiliza el programa. Las cuevas de código existen en la mayoría de aplicaciones, y añadir código en estas secciones normalmente no altera el comportamiento de la aplicación principal.

A menudo, el código de ejecución que se inserta en una cueva de código es simplemente un iniciador de shell remoto o puerta trasera. Este suele ser muy pequeño y sencillamente concede acceso al adversario al endpoint en que puede realizar otras acciones. Este tipo de ataque requiere que el adversario tenga establecida una presencia en el endpoint, a fin de que pueda desplegar la aplicación de puerta trasera o engañar al usuario para que descargue e instale una aplicación que tiene la cueva de código ya explotada.

Una de las principales razones por las que utilizan cuevas de código los adversarios es evitar que los detecten los usuarios generales y administradores. La aplicación esperada sigue funcionando bien, pero la aplicación insertada también se está ejecutando.

Si la aplicación que se ha modificado es una herramienta empresarial legítima que el administrador espera encontrar en el dispositivo, es menos probable que la considere malware si un antivirus tradicional detecta un problema. Es posible que los administradores simplemente la añadan a la lista de exclusión al dar por supuesto que el motor antivirus ha generado un falso positivo. De esta forma, el adversario establece persistencia en el endpoint e incluso puede haber engañado al administrador para que permita la ejecución de la aplicación insertada.

En lo que se conoce como ataque a la cadena de suministro, un atacante también puede infiltrarse en los servidores de actualización de software y hacer que una actualización enlace a código malicioso para infectar a sus clientes de forma silenciosa con ransomware o malware limpiador (wiper).

Sophos Intercept X bloquea de forma automática la ejecución de aplicaciones que están vinculadas a una puerta trasera. Incluso detecta el shellcode añadido cuando la ejecución de código no fluye a una cueva de código o a una sección añadida en el archivo PE infectado. Ofrece una amplia protección contra herramientas de inyección de shellcode como Shellter y Backdoor Factory.

Migración de procesos (inyección de DLL reflectiva remota)

La migración de procesos es una técnica que aplica comúnmente el adversario cuando establece su presencia en un dispositivo al inicio y quiere pasar a otro proceso a fin de aumentar privilegios o hacerse con un acceso más perdurable. El adversario no quiere perder el control cuando el usuario final simplemente cierra el navegador o finaliza un proceso que se ha visto comprometido, de modo que le conviene migrarse a un proceso del sistema.

Un ataque de DLL reflectivo remoto es similar a una migración de procesos. El adversario ya ha comprometido un proceso y, a partir de ahí, manipula otro proceso para cargar archivos DLL y ejecutar código arbitrario.

Aumento de privilegios locales (LPE)

Sophos Intercept X evita que un proceso con privilegios limitados obtenga más privilegios a través de un token robado de otro proceso con privilegios más altos. Esta técnica suele utilizarse junto a otra vulnerabilidad para distribuir y ejecutar eficazmente el código malicioso de un atacante con permisos del sistema.

Inyección de código DoublePulsar

Originalmente, DoublePulsar era una herramienta de implantación de puerta trasera desarrollada por The Equation Group de la Agencia de Seguridad Nacional (NSA, por sus siglas en inglés) de los Estados Unidos que fue filtrada por The Shadow Brokers a principios de 2017. La implantación contiene una novedosa técnica de inyección que forma parte de varios exploits de la NSA, como EternalBlue y EternalRomance. Estos exploits también se utilizaron para el componente de autopropagación de tipo gusano en los brotes del WannaCry y NotPetya.

La técnica de inyección de código de DoublePulsar utiliza una llamada a procedimiento asincrónico (APC) para ejecutar código arbitrario (shellcode) dentro de un proceso de confianza habitual. Sophos Intercept X doblega el sistema fundamental utilizado por DoublePulsar y, por lo tanto, también detiene los ataques que se basan en la misma técnica para inyectar código.

Inyección de código AtomBombing

La inyección de llamada a procedimiento asincrónico (APC) implica adjuntar código malicioso a la cola de APC del subproceso de un proceso. Las funciones de APC en cola se ejecutan cuando el subproceso entra en un estado modificable. AtomBombing es una variante que usa APC para invocar código malicioso previamente guardado en la tabla ATOM global.

Inyección de código DoubleAgent

DoubleAgent explota una herramienta legítima de Windows llamada Microsoft Application Verifier. Esta herramienta está incluida en todas las versiones de Microsoft Windows y se utiliza como herramienta de verificación en tiempo de ejecución para detectar y solucionar errores en aplicaciones. Application Verifier puede definirse para cargar cualquier biblioteca desde el disco, abriendo así la posibilidad de cargar una biblioteca maliciosa a la que se le otorgarán los permisos del proceso de la víctima.

DoubleAgent está acuñado como vulnerabilidad y ataque de día cero en los productos antivirus pero, en realidad, la finalidad de Application Verifier es cargar código arbitrario en la aplicación elegida, incluso los procesos de Windows y de productividad de confianza.

Sophos Intercept X evita la inyección de código a través de la explotación de Application Verifier.

Funciones de Intercept X

Funciones	
PREVENCIÓN DE EXPLOITS	
Aplicación de la prevención de ejecución de datos	✓
Selección aleatoria del diseño del espacio de direcciones obligatoria	✓
ASLR de abajo a arriba	✓
Página NULL (Protección de desreferencia NULL)	✓
Asignación de pulverización del montón	✓
Pulverización dinámica del montón	✓
Eje de la pila	✓
Ejecución de la pila (MemProt)	✓
Mitigaciones de ROP basadas en pilas (Autor de llamada)	✓
Mitigaciones de ROP basadas en ramas (Asistidas por hardware)	✓
Sobrescritura del controlador de excepciones estructurado (SEHOP)	✓
Filtrado de tabla de direcciones de importación (IAF)	✓
Carga de bibliotecas	✓
Inyección de DLL reflectiva	✓
Shellcode	✓
Modo Dios de VBScript	✓
Wow64	✓
Syscall	✓
Vaciado de procesos	✓
Secuestro de DLL	✓
Omisión de AppLocker Squiblydoo	✓
Protección de APC (Double Pulsar / AtomBombing)	✓
Aumento de privilegios de procesos	✓
MITIGACIONES DE ACTIVE ADVERSARY	
Protección contra robos de credenciales	✓
Mitigación de cuevas de código	✓
Protección contra Man-in-the-Browser (Navegación segura)	✓
Detección de tráfico malicioso	✓
Detección de shell Meterpreter	✓

Funciones	
PREVENCIÓN ANTIRANSOMWARE	
Protección contra archivos de ransomware (CryptoGuard)	✓
Detección automática de archivos (CryptoGuard)	✓
Protección del registro de arranque y disco (WipeGuard)	✓
BLOQUEO DE APLICACIONES	
Navegadores web (incluido HTA)	✓
Complementos de navegadores web	✓
Java	✓
Aplicaciones multimedia	✓
Aplicaciones de Office	✓
DEEP LEARNING	
Detección de malware con Deep Learning	✓
Bloqueo de aplicaciones no deseadas (PUA) con Deep Learning	✓
Supresión de falsos positivos	✓
Live Protection	✓
RESPONDER INVESTIGAR ELIMINAR	
Análisis de causa raíz	✓
Sophos Clean	✓
Seguridad Sincronizada con Security Heartbeat	✓
IMPLEMENTACIÓN	
Puede ejecutarse como agente independiente	✓
Puede ejecutarse junto a un antivirus existente	✓
Puede ejecutarse como componente de un agente Sophos Endpoint existente	✓
Windows 7	✓
Windows 8	✓
Windows 8,1	✓
Windows 10	✓
macOS*	✓

* admite las funciones CryptoGuard, detección de tráfico malicioso, Seguridad Sincronizada con Heartbeat, análisis de causa raíz

Pruebe Sophos Intercept X gratis

en es.sophos.com/intercept-x

Ventas en España
Teléfono: (+34) 913 756 756
Correo electrónico:
comercialES@sophos.com

Ventas en América Latina
Correo electrónico:
Latamsales@sophos.com

18-03-06 WP-ES (DD)

SOPHOS