SOPHOS

Tutto ciò che occorre sapere sugli exploit:

Una prevenzione antiexploit completa

Gli exploit sfruttano le debolezze presenti in prodotti software legittimi quali Adobe Flash e Microsoft Office per infettare i computer a scopo criminale. Vengono comunemente utilizzati dai cybercriminali per superare le barriere di difesa delle aziende. Questi criminali hanno obiettivi di vario genere: furto o blocco dell'accesso ai dati per ottenere un riscatto, attacchi di ricognizione o semplicemente nuovi metodi per distribuire malware più tradizionale.

Non è insolito trovare attacchi informatici basati sugli exploit: più del 90% dei casi di violazione dei dati segnalati include l'uso di un exploit in una o più fasi della catena di attacco. Includere la prevenzione degli exploit in una strategia di difesa completa è chiaramente una mossa vincente.

Gli exploit sono in circolazione da più di 30 anni, per cui non c'è da sorprendersi se quasi tutti i più importanti vendor di soluzioni di sicurezza dichiarano di offrire un certo livello di prevenzione antiexploit. Tuttavia l'ambito di applicazione e l'efficacia di tale protezione varia notevolmente da vendor e vendor. Per alcuni si tratta solamente di un'opzione in più da includere, per altri invece è una delle funzionalità principali che merita la massima attenzione. Vi invitiamo a proseguire con la lettura di questo documento per ottenere maggiori informazioni sugli exploit e sui vari livelli di protezione disponibili nei principali prodotti di sicurezza attuali.

Whitepaper Sophos marzo 2018

Indice dei contenuti

Gli exploit come industria: Crimeware as a Service

Con l'arrivo dei kit di exploit, ora gli autori di malware non sono più costretti a cercare bug in Java, Silverlight o Flash; non devono neppure preoccuparsi di come sfruttare questi bug per creare exploit efficaci, né di come individuare server web privi della dovuta protezione, e nemmeno di come indurre le potenziali vittime a visitare pagine web piene di trappole.

Analogamente, gli autori dei kit di exploit non si trovano costretti a compilare malware completi, non devono gestire server per tener traccia dei computer infettati, né cercare di estorcere denaro dalle singole vittime; inoltre, non hanno l'obbligo di essere coinvolti direttamente nei processi di esfiltrazione o vendita dei dati prelevati illecitamente.

Oggi come oggi, il cybercrimine è una vera e propria industria che fattura miliardi di dollari e si prevede che nel 2019 genererà danni pari a quasi duemila miliardi di dollari. Di conseguenza, tutti gli aspetti degli attacchi sono stati industrializzati.

I criminali hanno l'imbarazzo della scelta e possono specializzarsi in uno o più ambiti del panorama delle minacce, offrendo quello che viene sarcasticamente denominato CaaS, ovvero "Crimeware-as-a-Service".
In questo settore ora molto lucrativo, sono emersi intermediari che acquistano exploit da chi li scopre e li vendono a chi desidera utilizzarli, sia che si tratti di enti governativi oppure di pericolosissimi hacker.
Inevitabilmente, gli acquirenti non divulgano le proprie intenzioni. Come ha dichiarato Kevin Mitnick, fondatore di Mitnick's Absolute Zero Day Exploit Exchange, in un'intervista a Wired: "Quando abbiamo un cliente che desidera una vulnerabilità del giorno zero per qualsiasi motivo, non facciamo domande, anche perché non risponderebbe. I ricercatori le trovano, le vendono a noi per un tot, noi le rivendiamo ai clienti per un po' di più, e nel frattempo otteniamo un guadagno".

"Quando abbiamo un cliente che desidera una vulnerabilità del giorno zero per qualsiasi motivo, non facciamo domande, anche perché non risponderebbe. I ricercatori le trovano, le vendono a noi per un tot, noi le rivendiamo ai clienti per un po' di più, e nel frattempo otteniamo un guadagno".

Kevin Mitnick

Tecniche di mitigazione degli exploit

Con più di 400.000 nuovi tipi di malware unici creati ogni giorno e con migliaia di nuove vulnerabilità rilevate ogni anno, la prevenzione degli attacchi malevoli è diventata una sfida insormontabile. Questa crescita esponenziale delle varanti di malware richiede approcci nuovi e innovativi per la protezione contro i cybercriminali.
Un'analisi accurata dell'attuale settore del cybercrimine rivela ottime opportunità per i sistemi di difesa asimmetrici. A quanto pare, nonostante la valanga infinita di nuovi attacchi, le tecniche che possono essere utilizzate per generare exploit dei software sono solo poco più di una ventina.
Di conseguenza, a differenza di un sistema che si proponga di risolvere ciascun singolo exploit, un approccio che sia in grado di rispondere a questa manciata esigua di exploit si rivela molto potente.
Ma c'è di più: a seconda della vulnerabilità, spesso gli autori degli attacchi finiscono per dover collegare alcune tecniche di exploit per poter giungere al punto di poter inviare il malware. Queste tecniche non variano molto di anno in anno: è possibile che all'elenco di tecniche disponibili vengano aggiunti uno o due stratagemmi nuovi all'anno.
Quando si considera la validità di prodotti di sicurezza essenziali, può sorprendere l'assenza di una tecnica di mitigazione degli exploit efficace. E anche se alcuni nuovi vendor che dichiarano di avere tecnologie next-gen offrono il supporto generale della mitigazione degli exploit, anche qui l'efficacia può essere sporadica.

Quello che segue è un elenco di strategie di mitigazione degli exploit che hanno lo scopo di eliminare categorie intere di vulnerabilità e di contrastare le tecniche di exploit utilizzate dai cybercriminali e dagli enti governativi. La strategia di mitigazione degli exploit per ciascuna tecnica varia a seconda del vendor. È importante tenere presente che, quando un vendor dichiara di offrire prevenzione degli exploit, nella maggior parte dei casi si limita a offrire protezione contro una frazione dei metodi di exploit più comunemente utilizzati nelle applicazioni a 64 bit. Solo Sophos è in grado di garantire una prevenzione degli exploit a 360 gradi.

Implementazione della Data Execution Prevention (DEP)

La Data Execution Prevention (DEP) è un set di tecnologie hardware e software che svolgono controlli aggiuntivi della memoria, per prevenire i buffer overflow. Senza DEP, un hacker può tentare di attaccare una vulnerabilità dei software saltando direttamente alla parte del codice malevolo (shellcode) in una posizione di memoria nella quale risiedono dati controllati dagli hacker, come ad es. heap o stack. Senza DEP, queste zone vengono normalmente contrassegnate come eseguibili, per cui in esse il codice malevolo ha la possibilità di avviarsi.

DEP è un'opzione che deve essere selezionata esplicitamente ed è disponibile per Windows XP e versioni successive. È il vendor del software a doverla impostare in fase di compilazione di un'applicazione. Inoltre, sono disponibili attacchi che bypassano la protezione DEP integrata, per cui si sconsiglia di utilizzare dipendenze per l'implementazione nel sistema operativo.

Uso obbligatorio di ASLR (Address Space Layout Randomization)

Alcuni exploit agiscono attaccando posizioni di memoria note per essere associate a processi specifici. Nelle versioni meno recenti di Windows (incluso Windows XP), i processi core tendevano a essere caricati in posizioni di memoria prevedibili all'avvio del sistema. L'ASLR randomizza le posizioni di memoria utilizzate dai file di sistema e da altri programmi, complicando per gli hacker il processo di deduzione della posizione corretta di un processo specifico, inclusa la base del file eseguibile e la posizione dello stack, dell'heap e delle librerie.

L'ASLR è disponibile solamente in Windows Vista e versioni successive e, come DEP, deve essere impostata dal vendor del software in fase di compilazione di un'applicazione. Inoltre, proprio come avviene per DEP, sono disponibili attacchi che bypassano la protezione ASLR integrata, per cui si sconsiglia di utilizzare dipendenze per l'implementazione nel sistema operativo.

ASLR bottom-up

Se attivata, l'ASLR bottom-up incrementa l'entropia o la randomizzazione dell'ASLR obbligatoria.

Il vantaggio principale dell'ASLR obbligatoria e dell'ASLR bottom-up in Sophos Intercept X è il fatto che gli indirizzi di base delle applicazioni non sono randomizzati solamente a ciascun riavvio, bensì ogni volta che viene aperta l'applicazione protetta.

Null Page (protezione contro Null Dereference)

A partire da Windows 8, Microsoft nega ai programmi la capacità di attribuire e/o mappare la "pagina NULL" (la memoria che risiede presso l'indirizzo virtuale 0x00000000 nello spazio degli indirizzi). In questo modo Microsoft riesce a mitigare gli exploit diretti di un'intera classe di vulnerabilità che si chiamano "NULL pointer dereference".

In Windows XP, Windows Vista e Windows 7, l'exploit di questa falla di sicurezza consentirebbe agli autori degli attacchi di eseguire codice nel contesto del kernel (sotto il livello di privilegio ring0 della CPU). Il risultato è l'escalation dei privilegi sino a raggiungere uno dei livelli più elevati.

Questi tipi di vulnerabilità concedono agli hacker l'accesso a quasi tutte le parti dei sistemi operativi.

Heap Spray Pre-Allocation

L'Heap Spray Allocation è una tecnica che non si serve direttamente delle vulnerabilità, bensì che rende le vulnerabilità molto più semplici da attaccare. Utilizzando una tecnica denominata Heap Feng Shui[1], un hacker è in grado di collocare con precisione strutture dei dati o shellcode specifici nell'heap, favorendo il successo di un attacco tramite l'exploit di una vulnerabilità del software.

Un tipico metodo dell'heap spray mitigation prevede la conservazione o preallocazione degli indirizzi di memoria più comunemente utilizzati, in modo che non possano essere adoperati per includere payload. Gli autori degli attacchi più creativi sono a conoscenza di questi indirizzi, per cui in situazioni realistiche questo tipo di mitigazione non è molto efficace. Nota anche come implementazione di Anti-Heap Spray o preallocazione dello shellcode, la heap spray pre-allocation è solitamente efficace contro gli exploit predefiniti utilizzati dalle organizzazioni che si occupano di test indipendenti.

Dynamic Heap Spray

Rispetto alla heap spray pre-allocation, che è statica, la mitigazione Dynamic Heap Spray viene solitamente attivata quando si verifica un aumento improvviso dell'utilizzo di memoria.

La mitigazione Dynamic Heap Spray agisce analizzando i contenuti delle allocazioni della memoria più recenti, per rilevare pattern che indichino disseminazioni nell'heap che contengono slitte di NOP, slitte di NOP polimorfiche, matrici JavaScript, e altre sequenze sospette che vengono collocate strategicamente nell'heap per favorire la buona riuscita degli attacchi di exploit.

Stack Pivot

Lo stack di un'applicazione è un'area di memoria che contiene, tra altre cose, un elenco di posizioni degli indirizzi di memoria (noti come indirizzi mittente). Questi percorsi contengono il codice che il processore deve eseguire nell'immediato futuro.

Lo stack pivoting viene spesso utilizzato dagli exploit delle vulnerabilità per bypassare sistemi di protezione come DEP, utilizzando ad esempio catene di gadget ROP in un attacco di programmazione return-oriented.

Con lo stack pivoting, gli attacchi possono utilizzare uno stack esistente per lanciarsi verso un nuovo stack fasullo, che potrebbe essere un buffer controllato da un cybercriminale, come ad es. l'heap, da cui poi gli hacker possono controllare il flusso di esecuzione dei programmi futuro.

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

Stack Exec (MemProt)

In circostanze normali, lo stack contiene dati e indirizzi che puntano a un codice che deve essere eseguito dal processore nell'immediato futuro. Con un buffer overflow dello stack[2] gli autori degli attacchi possono sovrascrivere lo stack con codice arbitrario. Per fare in modo che questo codice si esegua nel processore, l'area di memoria dello stack deve essere resa eseguibile per eludere la DEP. Una volta che la memoria dello stack diventa eseguibile, risulta facile agli hacker inviare ed eseguire codice per i programmi.

Mitigazione ROP basata su stack (chiamante)

Per sconfiggere tecnologie di sicurezza quali data execution prevention (DEP) e ASLR (Address Space Layout Randomization), gli autori degli attacchi di solito ricorrono all'hijack del flusso di controllo delle applicazioni vulnerabili connesse a internet. Questi attacchi in memoria sono invisibili agli antivirus, alla maggior parte dei prodotti "next-gen" e ad altre difese informatiche, in quanto non sono presenti file malevoli. L'attacco viene invece impostato a livello di runtime, unendo frammenti di codice innocuo che fanno parte di applicazioni già presenti nei sistemi, quali Internet Explorer e Adobe Flash Player. Si tratta di un attacco noto come attacco di riutilizzo del codice o di programmazione orientata al ritorno (Return-Oriented Programming, ROP).

Durante un normale flusso di controllo, le funzioni API di natura sensibile (come ad es. VirtualAlloc e CreateProcess) vengono richiamate dall'istruzione CALL. Quando viene richiamata un'API di natura sensibile, tipicamente i sistemi di difesa contro ROP arrestano l'esecuzione del codice per determinare l'indirizzo che effettua il richiamo, utilizzando l'indirizzo del "mittente" che si trova in cima allo stack. Se l'istruzione dell'indirizzo che richiama l'API non è CALL, il processo viene terminato.

Siccome i contenuti dello stack sono scrivibili, l'autore di un attacco può scrivere nello stack valori specifici per depistare le analisi dei sistemi di difesa contro ROP basati sullo stack. I sistemi di difesa contro ROP basati sullo stack non sono in grado di stabilire se i contenuti dello stack siano benevoli o se siano stati manipolati da un hacker.

Misure di mitigazione ROP branch-based (Control-Flow Integrity assistita con hardware)

Come abbiamo visto, i sistemi di difesa contro ROP (Return-Oriented Programming, ovvero programmazione orientata al ritorno) basati sullo stack sono generici e manipolabili. Per risolvere questo problema, i sistemi di difesa devono avere dati più specifici e a prova di manipolazione da analizzare in fase di esecuzione.

Sophos Intercept X introduce la Control-Flow Integrity (CFI) assistita con hardware, approfittando di una funzionalità hardware non utilizzata nei processori Intel® più diffusi (2008 e successivi). L'hardware del processore include dati in sola lettura che assistono le operazioni di rilevamento degli attacchi di exploit più sofisticati in fase di esecuzione. L'uso di record riconducibili all'hardware (branch) presenta, in termini di sicurezza, un vantaggio significativo rispetto agli approcci software basati sullo stack. Le informazioni relative al branch che possono essere recuperate da questi record identificano sia l'obiettivo del branch che la sua origine. Per cui mostrano da dove ha avuto origine il cambiamento del flusso di controllo. Queste informazioni specifiche non possono essere ottenute con la stessa precisione utilizzando una soluzione basata sullo stack, come Microsoft EMET o Palo Alto Networks Traps.

Le informazioni relative al singolo branch nei record riconducibili all'hardware non possono essere manipolate ed è impossibile che vengano sovrascritte da dati controllati dall'autore di un attacco. Gli approcci basati sullo stack si affidano all'attendibilità dei dati dello stack che, specialmente nel caso di un attacco ROP, vengono controllati da un cybercriminale, il quale a sua volta può confondere la vittima. I dati riconducibili all'hardware analizzati da Sophos Intercept X sono invece più attendibili e a prova di manomissione.

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

Endgame offre un'implementazione della Control-Flow Integrity assistita con hardware (HA-CFI) alternativa, che si basa sull'addestramento del normale flusso di controllo per rilevare eventuali deviazioni dal percorso del codice impostato dal programmatore. Il sistema deve essere continuamente addestrato, in modo che compili una whitelist di indirizzi puntatori per il codice validi, che riflettano tutte le possibili funzionalità e versioni dell'applicazione protetta. Sophos Intercept X non richiede training e mantiene la stessa operatività durante i cambiamenti di contesto del thread e il ridimensionamento dinamico della frequenza.

Sophos Intercept X utilizza automaticamente l'analisi del flusso di controllo assistita con hardware quando rileva un processore (CPU) Intel® Core™ i3, i5, o i7. Se non viene individuato un hardware del processore supportato, Sophos Intercept X eseguirà automaticamente il fallback ai controlli basati sullo stack esclusivamente per il software.

Sophos Intercept X utilizza record con monitoraggio hardware per potenziare il rilevamento di ROP, ma non solo: viene anche impiegata nel filtro degli indirizzi di importazione (Import Address Filtering, IAF) per proteggere la tabella di indirizzi di importazione delle applicazioni protette.

Nota: Le patch per le vulnerabilità di Spectre che riguardano la predizione delle diramazioni all'interno dell'hardware delle CPU Intel non hanno ripercussioni sul corretto funzionamento di Sophos Intercept X.

Protezione strutturata contro la sovrascrittura del gestore eccezioni (Structured Exception Handler Overwrite Protection, SEHOP)

Gli autori degli attacchi possono fruttare un valore controllato per sovrascrivere il puntatore del gestore di un record di eccezioni nello stack. Quando si verifica un'eccezione, il sistema operativo percorrerà la catena di record di eccezioni ed effettuerà la chiamata verso tutti i gestori in tutti i record di eccezioni.

Siccome l'autore dell'attacco controlla uno dei record, il sistema operativo salterà verso qualsiasi punto indicato dall'hacker, garantendogli pieno controllo sul flusso di esecuzione.

SEHOP è un'opzione che deve essere selezionata esplicitamente ed è disponibile per Windows Vista e versioni successive. È il vendor del software a doverla impostare in fase di compilazione dell'applicazione. Sono disponibili attacchi che bypassano la protezione SEHOP integrata, per cui si sconsiglia di utilizzare dipendenze per l'implementazione nel sistema operativo.

Filtraggio tabella di indirizzi di importazione (Import Address Table Filtering, IAF)

A un certo punto dell'attacco, gli hacker avranno bisogno degli indirizzi delle funzioni specifiche del sistema (ad es. kernel32!VirtualProtect) per poter svolgere attività malevole.

Questi indirizzi possono essere recuperati da fonti diverse, tra cui la tabella di indirizzi di importazione (Import Address Table, IAT) di un modulo caricato. La IAT viene utilizzata come tabella di ricerca quando un'applicazione chiama una funzione in un modulo diverso. Siccome un programma compilato non può conoscere la posizione di memoria delle librerie da cui dipende, è necessario un salto indiretto ogni volta che viene effettuata una chiamata API. Poiché il linker dinamico carica i moduli e li unisce, sovrascrive indirizzi nelle slot della IAT, in modo che si dirigano verso le posizioni di memoria delle rispettive funzioni della libreria.

Sophos Intercept X introduce il filtro in accesso della tabella di indirizzi di importazione assistito con hardware, approfittando delle funzionalità hardware disponibili nei processori Intel® più diffusi (2008 e successivi). Oltre ai record di diramazione monitorati dall'hardware per implementare la Control-Flow Integrity, Sophos Intercept X sfrutta anche la predizione delle diramazioni per incrementare ulteriormente la protezione della tabella di indirizzi di importazione.

Nota: Le patch per le vulnerabilità di Spectre che riguardano la predizione delle diramazioni all'interno dell'hardware delle CPU Intel non hanno ripercussioni sul corretto funzionamento di Sophos Intercept X.

Load Library

Gli autori degli attacchi possono tentare di caricare librerie malevole, collocandole in percorsi UNC. Per prevenire questo tipo di caricamento delle librerie, è possibile utilizzare il monitoraggio di tutte le chiamate verso l'API LoadLibrary.

Reflective DLL Injection

Di solito, quando si carica un file DLL in Windows, viene chiamata la funzione API LoadLibrary. LoadLibrary utilizza il percorso del file DLL come input e lo carica in memoria.

Il caricamento di Reflective DLL si riferisce al caricamento di un DLL dalla memoria, piuttosto che dal disco. Windows non dispone di una funzione LoadLibrary in grado di supportare questa opzione, per cui per utilizzarla occorre scriverne una. Una delle caratteristiche dello scrivere una funzione è la possibilità di omettere alcune delle azioni di Windows, come ad es. la registrazione del DLL come modulo caricato nel processo. In fase di analisi, questa opzione rende più difficile l'individuazione del reflective loading. Meterpreter è un esempio tipico di strumento che utilizza il reflective loading per eludere il rilevamento. La mitigazione avviene effettuando l'analisi di un DLL per scoprire se sia caricato in memoria riflessivamente.

Shellcode

Gli shellcode sono frammenti di codice che vengono utilizzati come payload negli exploit delle vulnerabilità dei software. Si chiamano "shellcode" perché un tempo avviavano un comando shell dal quale l'hacker poteva controllare il computer compromesso, ma il termine definisce qualsiasi frammento di codice che svolga operazioni simili.

Un exploit agisce tipicamente incorporando uno shellcode all'interno di un processo di destinazione prima o nel momento in cui attacca una vulnerabilità per ottenere controllo sul puntatore delle istruzioni per il processore (EIP/RIP). Il puntatore delle istruzioni viene modificato in modo da utilizzare lo shellcode, dopodiché viene eseguito e svolge l'operazione impostata.

VBScript God Mode

Su Windows, VBScript può essere utilizzato nei browser o nella shell locale. Se utilizzato nel browser, le funzionalità di VBScript sono limitate, per motivi di sicurezza. Queste limitazioni sono controllate dal contrassegno di modalità provvisoria. Se questo contrassegno viene modificato, VBScript in HTML potrà agire esattamente come se non si trovasse nella shell locale. Di conseguenza, gli autori degli attacchi potranno facilmente scrivere codice malevolo in VBScript. La manipolazione del contrassegno di modalità provvisoria su VBScript nel browser web viene detta God Mode[3].

Un hacker può, ad esempio, modificare il valore del contrassegno di modalità provvisoria sfruttando la vulnerabilità CVE-2014-6332[4], un bug generato da una gestione inadeguata del ridimensionamento di una matrice nel motore VBScript di Internet Explorer. Nella God Mode, viene scritto del codice arbitrario in VBScript, che consente di sfuggire alla sandbox del browser. La God Mode disattiva i sistemi di sicurezza della data execution prevention (DEP), dell'Address space layout randomization (ASLR) e dell'integrità del flusso di controllo (CFI).

WoW64

Il livello "Windows on Windows" (WoW) è un'opzione di Microsoft che offre retrocompatibilità per il software a 32 bit nelle edizioni a 64 bit di Windows. Alcuni aspetti dell'implementazione del livello WoW offrono agli hacker opportunità interessanti per complicare analisi dinamiche, decomprimere file binari e bypassare le strategie di mitigazione degli exploit.

Il comportamento di un'applicazione a 32 bit nell'ambiente WoW64 presenta molte differenze rispetto a un sistema a 32 bit originale. La possibilità di poter cambiare modalità di esecuzione a livello di runtime può offrire agli hacker diversi metodi per lanciare attacchi basati su exploit, offuscamento e anti-emulazione, come ad es:

- Gadget ROP aggiuntivi, che non sono presenti nel codice a 32 bit
- Codificatori di payload in modalità di esecuzione miste
- Funzionalità dell'ambiente di esecuzione che potrebbero ridurre l'efficacia delle tecniche di mitigazione
- La capacità di bypassare hook inseriti dal software di sicurezza, solo nello spazio utente a 32 bit

Nella maggior parte dei casi, i software di protezione endpoint forniscono hook solo per le funzioni sensibili dell'API nello spazio di memoria utente a 32 bit, se un processo si esegue in WoW64. Se l'autore di un attacco è in grado di passare alla modalità a 64 bit, può ottenere accesso a versioni a 64 bit prive di hook delle funzionalità API sensibili, che invece presentano hook nella modalità a 32 bit.

Nelle edizioni a 64 bit di Windows, Sophos Intercept X vieta l'uso di codice programma quando si passa direttamente dalla modalità a 32 bit a quella a 64 bit (ad es. utilizzando ROP), pur consentendo al livello WoW64 di svolgere questa transizione.

Per maggiori informazioni sull'uso improprio di WoW64, consultare i risultati delle ricerche svolte da Duo Security: "WoW64 and So Can You"[5] e "Mitigating Wow64 Exploit Attacks"[6].

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

Syscall

Il termine syscall (che sta per "system call", ovvero chiamata di sistema) definisce il modo programmatico in cui un programma richiede un servizio al kernel del sistema operativo. Include i servizi basati su hardware, quali ad es. l'accesso al disco locale e la creazione e l'esecuzione di nuovi processi.

Generalmente il sistema operativo fornisce un'API (Application Programming Interface, ovvero interfaccia di programmazione di un'applicazione) che si interpone tra i programmi normali e il sistema operativo. In circostanze normali, un'applicazione chiama un'API per richiedere al kernel un'operazione specifica. Il software di sicurezza inserisce hook nelle funzionalità API di natura sensibile per intercettare e svolgere controlli quali la scansione antivirus, prima di permettere al kernel di procedere con la richiesta.

Gli autori degli attacchi possono approfittare di quanto segue:

- Non tutte le funzioni API contengono hook del software di sicurezza, bensì solamente quelle di natura sensibile.
- Gli stub utilizzati per chiamare le funzioni del kernel sono molto simili, solo l'indice è univoco.

Chiamando uno stub di una funzionalità di natura non sensibile a un offset (per cercare invece di raggiungere un servizio kernel di natura sensibile), un hacker può eludere la maggior parte dei software di sicurezza e delle analisi di sandboxing.

Sophos Intercept X include un approccio innovativo alla prevenzione degli attacchi diretti alle funzionalità di natura sensibile del kernel che sfruttano la mancanza di protezione di alcune funzionalità API.

Per maggiori informazioni sull'uso improprio di syscall, consultare la voce del blog BreakDev.org intitolata "Defeating Antivirus Real-time Protection From The Inside"[7].

Process Hollowing

Il process hollowing è una tecnica in cui un'applicazione legittima, ad es. explorer.exe o svchost.exe, viene caricata nel sistema solamente allo scopo di fungere da contenitore per il codice malevolo.

Solitamente procede creando un processo "vuoto" (hollow) in uno stato sospeso, per poi annullarne il mapping della memoria e inserire al suo posto del codice malevolo. Analogamente alla tecnica code injection, l'esecuzione del codice malevolo viene nascosta in un processo legittimo, per cui può eludere i sistemi di difesa e le analisi di rilevamento.

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

Process Doppelgänging

Nella maggior parte dei casi, i computer Windows utilizzano file system NTFS. Nel 2007 Microsoft ha introdotto una nuova funzionalità che si chiama Transactional NTFS (TxF). Questa funzionalità permette di raggruppare operazioni sui file multiple e considerarle come una sola unità: possono quindi risultare completate come unità (nonché confermate o non riuscite come unità) ed essere ripristinate allo stato originale. In questo modo, un'applicazione può apportare diverse modifiche a vari file su disco e ripristinare tutti i file al loro stato originale se viene rilevato un errore.

La TxF viene più comunemente utilizzata durante gli aggiornamenti Windows.

Il process doppelgänging sfrutta il meccanismo della TxF per nascondere il malware. Sceglie un file innocuo, lo sovrascrive ed esegue il malware utilizzando un'API di basso livello, ad es. per spacciarsi per un file attendibile (in maniera analoga al process hollowing). Prima di permettere al malware di eseguirsi, rifiuta o ripristina tutte le modifiche effettuate, impedendo quindi al software antivirus di effettuare la scansione dei contenuti del file che viene eseguito. Se aperto, il file su disco non presenta alcun contenuto sospetto. Inoltre, questo file può essere un'applicazione comune e dotata di firma digitale.

DLL Hijacking

La vulnerabilità comunemente nota come DLL hijacking, DLL spoofing, DLL preloading o binary planting fa in modo che molti programmi carichino ed eseguano un codice DLL malevolo, contenuto nella stessa cartella del file di dati aperto da questi programmi.

Dynamic Data Exchange (DDE)

Windows Dynamic Data Exchange (DDE) è un protocollo client/server per la comunicazione tra processi (IPC) delle applicazioni. Gli hacker possono approfittare del DDE per eseguire comandi arbitrari. Ad esempio, i documenti Microsoft Office possono essere soggetti ad attacchi di poisoning con comandi DDEAUTO, per poi essere utilizzati per effettuare l'esecuzione di comandi PowerShell tramite campagne di spearphishing o contenuti web in hosting, evitando l'uso delle macro Visual Basic for Applications (VBA). I comandi DDEAUTO possono anche essere inseriti nel corpo del messaggio di e-mail o richieste di meeting, per poi essere eseguiti quando il destinatario risponde o accetta le richieste in Microsoft Outlook.

Grazie al design del sistema di mitigazione Lockdown delle applicazioni, la natura intrinseca di Sophos Intercept X impedisce anche l'esecuzione di codice malevolo tramite Dynamic Data Exchange.

Lockdown delle applicazioni

Se l'autore di un attacco dovesse riuscire a inviare un exploit e bypassare tutte le tecniche di protezione della memoria e del codice, Sophos Intercept X limiterà la libertà di azione dell'hacker. Questa funzionalità, che si chiama Lockdown delle applicazioni, serve a impedire agli autori degli attacchi di introdurre codice indesiderato.

Il Lockdown delle applicazioni blocca gli attacchi che non si basano tipicamente su bug del software nelle applicazioni. Tali attacchi potrebbero ad esempio prevedere l'uso di macro modificate (o malevole) in un documento Office allegato a un'e-mail di (spear)phishing.

Le macro nei documenti sono potenzialmente pericolose, in quanto vengono create nel linguaggio di programmazione Visual Basic, Applications Edition (VBA), che include la capacità di scaricare ed eseguire file binari dal web e di autorizzare l'uso di PowerShell e altre applicazioni attendibili.

Questa funzionalità inattesa (anche detta logic-flaw exploit) offre un netto vantaggio agli autori degli attacchi, che, per infettare i computer, non sono costretti a utilizzare bug dei software o a trovare un modo per bypassare i sistemi di difesa del codice e della memoria. Basta semplicemente che utilizzino in maniera impropria una funzionalità standard offerta da un'applicazione attendibile e di uso molto comune, adottando stratagemmi di ingegneria sociale per persuadere la vittima ad aprire il documento che funge da esca.

Senza costringere gli utenti a gestire una blacklist di cartelle, Sophos Intercept X termina automaticamente le applicazioni protette in base al loro comportamento. Per esempio: quando un'applicazione Office viene utilizzata per lanciare PowerShell, eseguire una macro per installare codice arbitrario o manipolare aree critiche del sistema, Sophos Intercept X blocca l'azione malevola, anche quando l'attacco non genera un processo figlio.

Lockdown di Java

Un tempo i kit di exploit erano il metodo principale per abilitare i download drive-by contenenti malware.

Sfruttavano vulnerabilità di Java Runtime Environment (JRE) per inviare payload Windows PE. JRE viene caricato come plug-in o componente aggiuntivo nei browser più comunemente diffusi.

Sophos Intercept X impedisce a JRE di eseguire applicazioni non Java. Sophos Intercept X termina, ad esempio, un'applicazione Java quando effettua un tentativo di introdurre ed eseguire un file binario Windows PE. Inoltre, impedisce agli autori degli attacchi di utilizzare Java in maniera impropria per manipolare i percorsi di avvio automatico, inclusi: la cartella Esecuzione automatica, Esegui, RunOnce e altri chiavi del registro di sistema.

Nota: Con l'introduzione dell'aggiornamento 20 di Java 8 nel 2014, il livello di sicurezza delle applicazioni Java è impostato di default su "Alto". Questo accorgimento ha reso più difficile l'esecuzione da parte degli hacker di exploit di Java dotati di autorizzazioni sufficienti per poter infettare l'endpoint. Di conseguenza, gli exploit di Java non sono più uno dei metodi più comunemente utilizzati nei kit di exploit, per cui il sistema di mitigazione Lockdown delle applicazioni Java sembra essere diventato obsoleto.

Code cave

Code cave è una tecnica sfruttata dai cybercriminali per modificare software legittimo in modo da inserirvi un'altra applicazione. Questa applicazione aggiuntiva viene inclusa in un elemento detto "code cave", ovvero una sezione inutilizzata del file dell'applicazione colpita. I code cave sono presenti nella maggior parte delle applicazioni e di solito l'aggiunta di codice a queste sezioni non interferisce con il comportamento dell'applicazione primaria.

Spesso il codice di esecuzione inserito all'interno di un code cave è semplicemente un metodo per avviare la shell in remoto o da una backdoor; può avere dimensioni molto limitate e può semplicemente fornire all'antagonista accesso a un endpoint in cui possa svolgere altre azioni. Per utilizzare questo tipo di attacco gli hacker devono prima stabilire una presenza nell'endpoint, per poter distribuire l'applicazione dalla backdoor o per indurre con l'inganno l'utente a scaricare e installare un'applicazione che è stata soggetta a exploit del code cave.

Uno dei motivi principali per cui i cybercriminali si servono dei code cave è eludere il rilevamento da parte di utenti generici e amministratori. L'applicazione principale continua a funzionare correttamente, ma esegue anche l'applicazione che è stata aggiunta.

Se l'applicazione che è stata modificata è uno strumento aziendale legittimo, la cui presenza nel dispositivo non segnala niente di insolito all'amministratore, è meno probabile che venga considerata come contenente malware se l'antivirus rileva un problema. Gli amministratori potrebbero semplicemente aggiungerla alla lista delle esenzioni, dando per scontato che si tratti semplicemente di un falso positivo generato dall'antivirus. In questo modo l'autore del malware garantisce la persistenza sull'endpoint e potrebbe persino aver indotto l'amministratore ad autorizzare l'esecuzione dell'applicazione malevola che è stata aggiunta.

In un attacco definito come attacco alla supply chain, un hacker può anche violare i server degli aggiornamenti del software per caricarvi un aggiornamento contenente codice malevolo e infettare in maniera invisibile i clienti con ransomware o malware wiper.

Sophos Intercept X blocca automaticamente l'esecuzione delle applicazioni contenenti una backdoor. Inoltre, è anche in grado di individuare shellcode aggiunti quando l'esecuzione del codice non porta a un code cave o a una sezione aggiuntiva del file PE. Offre ampia protezione contro strumenti di shellcode injection quali Shellter e Backdoor Factory.

Migrazione dei processi – reflective DLL injection in remoto

La migrazione dei processi è una tecnica di uso comune che viene adottata dai cybercriminali quando stabiliscono per la prima volta la propria presenza in un dispositivo e desiderano passare a un altro processo per ricevere privilegi più elevati o per ottenere un accesso più duraturo. L'obiettivo dei cybercriminali è mantenere il controllo anche quando l'utente finale chiude il browser o termina un processo compromesso, per cui cerca di migrare verso un processo di sistema.

Un attacco di reflective DLL in remoto agisce in maniera simile alla migrazione dei processi. Il creatore di malware ha già compromesso un processo e lo usa come punto di partenza per manipolare un altro processo, caricare DLL ed eseguire codice arbitrario.

Local Privilege Escalation (LPE)

Sophos Intercept X impedisce ai processi con privilegi bassi di passare a livelli superiori per mezzo di token prelevati illecitamente da processi dotati di privilegi più elevati. Questa tecnica viene spesso utilizzata insieme a un'altra vulnerabilità per consegnare ed eseguire il codice malevolo di un hacker con autorizzazioni di sistema.

Code Injection DoublePulsar

Originariamente, DoublePulsar era uno strumento per impiantare backdoor sviluppato dall'Equation Group dell'ente statunitense National Security Agency (NSA), la cui esistenza è stata resa nota da The Shadow Brokers all'inizio del 2017. La strategia di impiantazione prevedeva una tecnica innovativa di inclusione che rappresentava uno dei componenti di diversi exploit dell'NSA, inclusi EternalBlue ed EternalRomance. Questi exploit sono stati anche utilizzati per il componente worm a diffusione automatica rilevato nelle epidemie WannaCry e NotPetya.

La tecnica di code injection DoublePulsar utilizza le chiamate asincrone di procedura (APC) per eseguire codice arbitrario (shellcode) all'interno di un normale processo attendibile. La natura intrinseca di Sophos Intercept X interrompe il metodo di base di DoublePulsar e di conseguenza blocca anche gli attacchi che si affidano alla stessa tecnica per il code injection.

Code Injection AtomBombing

La tecnica di aggiunta tramite chiamata asincrona di procedura (APC) prevede l'aggiunta di codice malevolo alla coda dell'APC del thread di un processo. Le funzionalità dell'APC in coda vengono eseguite quando il thread entra in uno stato modificabile. AtomBombing è una variante che adopera le APC per richiamare codice malevolo precedentemente scritto nella tabella atom globale.

Code Injection DoubleAgent

DoubleAgent sfrutta uno strumento legittimo di Windows che si chiama Microsoft Application Verifier. Questo strumento è incluso in tutte le versioni di Microsoft Windows e viene utilizzato come strumento di verifica in fase di esecuzione per il rilevamento e la risoluzione di bug delle applicazioni. Application Verifier può essere impostato per caricare qualsiasi libreria dal disco, esponendo quindi i sistemi al potenziale caricamento di una libreria malevola a cui verranno attribuite le autorizzazioni dei processi della vittima.

DoubleAgent nasce come attacco del giorno zero e attacco delle vulnerabilità, ma in realtà Application Verifier serve a caricare codice arbitrario in un'applicazione selezionata, inclusi i processi di produttività e Windows attendibili.

Sophos Intercept X previene il code injection tramite utilizzo improprio di Application Verifier.

Le funzionalità di Intercept X

Funzionalità	
PREVENZIONE DEGLI EXPLOIT	
Implementazione della Data Execution Prevention (DEP)	✓
Uso obbligatorio di ASLR (Address Space Layout Randomization)	✓
ASLR bottom-up	✓
Null Page (protezione contro Null Dereference)	✓
Heap Spray Allocation	✓
Dynamic Heap Spray	✓
Stack Pivot	✓
Stack Exec (MemProt)	✓
Misure di mitigazione ROP basate su stack (chiamante)	✓
Misure di mitigazione ROP branch-based (assistite da hardware)	✓
Protezione strutturata contro la sovrascrittura del gestore eccezioni (Structured Exception Handler Overwrite Protection, SEHOP)	✓
Filtraggio importazione della tabella indirizzi (Import Address Table Filtering, IAF)	✓
Load Library	✓
Reflective DLL Injection	✓
Shellcode	✓
VBScript God Mode	✓
WoW64	✓
Syscall	✓
Hollow Process	✓
DLL Hijacking	✓
Squiblydoo Applocker Bypass	✓
Protezione contro le APC (Double Pulsar / AtomBombing)	✓
Privilege escalation dei processi	✓
BLOCCO DEGLI ATTACCHI DEI CYBERCRIMINALI	
Protezione contro il furto di credenziali	✓
Mitigazione di code cave	✓
Protezione contro gli attacchi man-in-the-browser (Safe Browsing)	✓
Malicious Traffic Detection (Rilevamento del traffico malevolo)	✓
Rilevamento shell Meterpreter	✓

Funzionalità	
PREVENZIONE ANTIRANSOMWARE	
Protezione antiransomware per i file (CryptoGuard)	✓
Recupero automatico dei file (CryptoGuard)	✓
Protezione del disco e del record di avvio (WipeGuard)	✓
LOCKDOWN DELLE APPLICAZIONI	
Browser web (incluso HTA)	✓
Plugin dei browser web	✓
Java	✓
Applicazioni multimediali	✓
Applicazioni Office	✓
DEEP LEARNING	
Rilevamento antimalware con tecnologie di deep learning	✓
Blocco delle applicazioni potenzialmente indesiderate (PUA) con deep learning	✓
Eliminazione dei falsi positivi	✓
Live Protection	✓
RISPOSTA INVESTIGAZIONE RIMOZIONE	
Root Cause Analysis	✓
Sophos Clean	✓
Synchronized Security Heartbeat	✓
DELIVERY	
Esecuzione possibile come agente standalone	✓
Esecuzione possibile insieme ad antivirus già esistente	✓
Esecuzione possibile come componente di un agente Sophos Endpoint già esistente	✓
Windows 7	✓
Windows 8	✓
Windows 8,1	✓
Windows 10	✓
macOS*	✓

* Funzionalità supportate: CryptoGuard, Malicious Traffic Detection, Synchronized Security Heartbeat, Root Cause Analysis

Prova gratuita di Sophos Intercept X

su sophos.it/intercept-x

Vendite per l'Italia:
Tel: (+39) 02 94 75 98 00
E-mail: sales@sophos.it

06/03/18 WP-IT (DD)

SOPHOS