Indice

Indice della guida dei controlli – stilata da bestemmie, 22/02/2023 - 22/01/2024

Questa guida è suddivisa in due parti, che pur essendo alla fine tutta teoria, ricadrà in argomenti pratici, perciò ecco la suddivisione dei vari capitoli e come dovrebbero essere seguiti

Vi auguro una buona lettura, prendetevela con calma e cercate di imparare tutto per bene, ma sopratutto, va bene la teoria, ma mi raccomando applicate tutto facendo tanta pratica.

- Bestemmie

Introduzione ai controlli

Introduzione alla guida dei controlli – stilata da bestemmie, 22/02/2023 - 22/01/2024

Un caldo benvenuto da parte mia e da tutta la community SS a chiunque tu sia, che hai deciso di leggere questa guida. Prima di iniziare ci tengo a precisare due cose:

- Questa guida è stata pensata per essere completa, non per forza semplice o intuitiva. Pur coprendo la maggiorparte degli aspetti dei controlli e alcuni dello studio dell'informatica forense, questa non è una guida che favorisce chiunque. Se si decide di leggere questa guida bisogna avere una mente libera da ogni preconcetto sentito sui controlli, oltre ad essere aperta all'imparare nuove cose, ad accettare nuovi concetti ma saper usare anche lo spirito critico e valutare anche degli eventuali errori, perché si, pur essendo questa opera di dimensioni mai viste e grandi attenzioni e cure sono state portate alla stesura di questo testo (eccetto l'italiano corretto maybe ._.), non tutte le nozioni fornite possono essere precise. Io sono aperto a commenti costruttivi sulla guida etc.
- Questa guida è stata scritta interamente dal sottoscritto (aka Bestemmie), ogni nozione fornita qua è puramente scritta in da me e solo da me, non significa che le cose imparate le ho inventate io, ci mancherebbe, ma sicuramente non sono stati altri a scriverle così, men che meno a insegnarle in questo modo. É una carta destinata al pubblico e deve essere, di conseguenza, di pubblico dominio. Ogni cosa presa da qua può essere reinterpretata e riutilizzata, affinché ci siano almeno crediti all'autore. Non è vietato copiare / prendere ispirazione, ma per favore come impegno comune .. siate onesti con me, con gli altri e con voi stessi.

Detto questo, Buona lettura …..

Controllo, definizione ed overview generale

Il controllo hack, anche conosciuto come screenshare in inglese, è una pratica che gli staffer utilizzano per dimostrare che un player sta utilizzando o meno dei cheats. Per avere un minimo di knowledge base su alcuni termini dei controlli partirei proprio da uno dei punti dolenti: il bypass. La definizione comune di bypass è la seguente: "Il bypass avviene quando lo staffer non riesce a trovare i cheats ad un player, nonostante lui li stava utilizzando. Questa è una cosa molto generica e secondo me è soltanto fonte di dibattito inutile. Come prima cosa, molte persone hanno diversi significati di "trovare i cheats", ma io penso che non esista una cosa generica come trovare i cheats. Intanto ci possono essere due tipi di server, i server che bannano per possesso e quelli che non lo fanno. Se parliamo di server che bannano per il possesso, trovare un cheat si limita a trovare delle tracce dell'esistenza di un cheat qualsiasi in un arco di tempo stabilito dal regolamento. Non ci si dovrebbe preoccupare più di tanto degli orari di esecuzione etc.. più che altro perché sono le prove dell'esistenza dei cheat in primo luogo a contare nella prova per il ban. Pochi server però decidono di bannare ancora con questi criteri, e qui entriamo nella categoria principale per la maggiorparte di voi: La dimostrazione dell'esecuzione. Essendo questa una introduzione non mi soffermerò su tutti i modi di dimostrare l'esecuzione di un programma (chiaramente), bensì ci terrò a precisare due cose in croce. Bisogna in questi casi, avere prove che un cheat (.exe, .jar, .py, etc ...) o una libreria (.dll e così via), sia stato eseguito e non solo. Bisogna avere un timestamp se si vuole essere precisi, perché un cheat eseguito può essere considerato in due modi: fuori istanza e dentro. Per istanza si intende il processo di minecraft attuale, quindi bisogna confrontare l'esecuzione del cheat, con il timestamp

di avvio del processo di minecraft. Ban fatti in altro modo sarebbero imprecisi e/o senza abbastanza prove. Tutto dipende, alla fine della giornata, dalle regole del server, ma è necessario in primo luogo, avere una conoscenza base delle casistiche in cui si banna e in cui no. Detto questo per ricollegarci al concetto dei bypass, un ban è valido finché rispetta le regole del server, non quelle inventate dalla "community". Questo lo specifico perché nel tempo si sono avuti molti episodi di flame altamente inutile tra staffer e bypasser, che si linciano perché uno che ha bypassato e che è stato "false bannato" mentre il povero staff cerca di fare il possibile per difendersi. Assumendo che lo staffer in questione abbia seguito alla lettera il suo regolamento e le prove che ha fornito sono necessarie, un bypasser non ha motivo neanche di parlare, se non per spawnare flame a random, ma una cosa è sicura, lo staffer in questione non è stato bypassato. Altra casistica invece, quando lo staffer non trova niente, il bypasser dice di aver usato un "client privato" e lo staffer si sente come giustificato a non aver trovato niente. Non funziona così, nelle varie tecniche per fare controlli ci sono infiniti metodi per dimostrare che un artefatto sia effettivamente un cheat, quindi lo staffer non è giustificato se il bypasser ha usato un "client privato". Come ultima cosa, farsi bypassare succede, va bene ogni tanto il flame da npc, ma alla fine sono solo esperienze costruttive, che ti insegnano cosa hai sbagliato, e quindi indicano cosa dovrai fare in un altro modo. Gli staffer non dovrebbero bannare per paura di essere bypassati, ne avere questa paura a prescindere. Si praticano i controlli per eliminare i cheater, non per costruire una reputazione. Alla fine tanto è solo un gioco.

Tool vari

I controlli non si fanno di certo evocando demoni pagani o leggendo nella mente del player, dei tool vengono utilizzati per praticare lo screenshare e quindi voglio fare anche qui un po' di chiarezza per chi è nuovo. I tool li possiamo dividere in 3 categorie (2 principali, una di queste sarebbe una sottocategoria, ma fa lo stesso):

- Tool per la condivisione schermo
- Tool per la ricerca di cheats:
    - Tool automatici
    - Tool manuali

Partendo dalla prima, i tool per la condivisione schermo penso siano i più ovvi; in questa categoria fanno parte programmi come Anydesk, Teamviewer e anche discord a volte. É preferibile che uno staffer abbia la possibilità di utilizzare mouse e tastiera del pc appartenente al player, quindi l'utilizzo di anydesk o teamviewer dovrebbe essere un must. In alcune casistiche si può usare discord ma è rara come occasione nonché comunque molto inefficace.
I tool per la ricerca dei cheats, sono tutti quei programmi e command line utilities (programmi per riga di comando), che utilizziamo per effettivamente dimostrare che il player è in possesso ed eventualmente utilizzo di cheats. Ci sono i tool manuali, che ci permettono di visualizzare una vasta gamma di artefatti più o meno tradotti in campi che possiamo capire e quindi interpretare. Manuali quindi perché richiedono la nostra interpretazione. Ci sono poi i tool automatici per controlli, come echo, avenge, ssdetector e tanti altri, che controllano automaticamente vari artefatti, e confrontano le prove con i loro metodi per verificare se un player ha utilizzato cheats. Questi tool sono tanto affidabili quanto non. É sempre meglio avere una conoscenza avanzata dei controlli prima di affidarsi a ss tools automatici, dato che si rendono veloce il controllo di tanto, ma allo stesso tempo bisogna stare attenti nei casi in cui non segnala niente, o segnala un cheat che sembra strano o non coincide

con le condizioni del player in quel momento. Ci sono tante cose da prendere in considerazione perciò sconsiglio l'uso dei tool automatici ai principianti.

Il viaggio è appena iniziato

Nelle successive guide troverete più nello specifico gli argomenti, spiegati per bene e in modo sopratutto completo. Vi auguro un buon proseguimento e mi raccomando rimanete aggiornati e curiosi.

Minecraft

Guida approfondita su cos'è minecraft dal punto di vista dei controlli

É giusto pensare ai controlli cheats come una branca dell'informatica forense, ma a differenza delle materie come DFIR e indagini varie, i controlli cheats sono legati molto anche al gioco che li ospita in primo luogo. Ecco perché qui vi spiego minecraft da un punto di vista di controlli, insieme ai vari tipi di cheats che ci possiamo aspettare e così via.

Il gioco e Java

Come ben sappiamo, minecraft è scritto in java, questo implica che come processo non esisterà mai un ipotetico "minecraft.exe" o robe del genere, bensì minecraft sarà una semplice task all'interno di un processo della Java Virtual Machine (java.exe o javaw.exe). Il gioco, non essendo appunto scritto in un linguaggio direttamente compilato (c/c++/c#/rust etc..), viene avviato tramite dei launcher. I launcher si occupano semplicemente di tenere un loro log delle attività, di offri alcune peculiarità rispetto a quello default e infine, la cosa più ovvia, scopo stesso dei launcher, è il fatto che hanno il comando loro per avviare il gioco (una java-jar con tanti argomenti in poche parole). I launcher dicono a minecraft anche dove ha la sua cartella radice. Per essere più precisi, minecraft, come tutti i giochi, non ha tutte le informazioni come texture e assets vari all'interno dell'eseguibile, bensì si appoggiano su una cartella radice, con all'interno un sistema di cartelle che ordina i vari file e risorse di cui necessita il gioco. La cartella di minecraft, con l'installazione più vanilla che si può avere, si trova in "C:\Users\{username}\AppData\roaming\.minecraft". Questo non significa che si può trovare esclusivamente li. Un esempio può essere MultiMC, un launcher usato da modders e player tecnici di minecraft, che ha come peculiarità quella che ogni versione di minecraft creata, crea una cartella di minecraft separata dalle altre e unica per quella versione, cosa molto comoda per i modders e tenere ordinate tutti i loro mod pack vari, ma ci indica come sia possibile che la cartella radice di minecraft si trovi in un posto diverso dalla posizione vanilla. Per accedere alla cartella di minecraft giusta, si consiglia di andare sul gioco > impostazioni > pacchetti risorse (resource packs) > apri cartella pacchetti risorse. Questa cosa aprirà la cartella resource packs sul disco, che ha come cartella parente, la cartella radice di minecraft.

I tipi cheats

I cheater con minecraft hanno a disposizione forse una tra le più vaste gamme di cheat rispetto a qualsiasi altro gioco, questo tutto per alcune peculiarità del gioco (foreshadowing: java). Prima di partire a scheggia su come trovare i cheats etc.., direi che è meglio avere una base teorica su cosa ci dobbiamo aspettare, cosa dobbiamo trovare in quanto staffer, sul pc del player. I client si dividono in quattro categorie principali, che poi analizzeremo per bene:

- Ghost clients
- Injection clients
- External clients
- Autoclickers

Ora i prossimi paragrafi saranno dedicati esclusivamente alla spiegazione nel dettaglio di questi client e di alcune cose fondamentali da sapere, non che metodi vari.

Ghost clients

I ghost clients, sono forse i client più semplici da trovare, per la quantità di prove che lasciano alla loro esecuzione, ma anche i più sottovalutati. Per definizione i ghost client possono essere delle versioni, delle mod o delle librerie, che vanno a modificare il gioco in modo che il player ha dei vantaggi sleali. Sono delle versioni perché molti tra i ghost client più famosi, sono delle vere e proprie versioni di minecraft che si scaricano e si mettono nella cartella minecraft/version, e si avvia quella specifica versione. Possono essere delle mod, perché tramite le mod noi possiamo implementare delle feature malevole nel gioco, senza andare ad intaccare però la versione originale del gioco. Possono infine essere delle librerie di minecraft, questa è una cosa meno vista in giro, solo un client particolarmente famoso usava questa tecnica, la Serenity. Le librerie sono tutti quei file esterni che contengono codice fondamentale per permettere a minecraft di essere eseguito. Se siete proprio bravi, avete notato un pattern particolare tra questi tipi di ghost clients, il loro obbiettivo è eseguire codice direttamente all'interno di minecraft, che poi sia tramite versione o tramite mod o tramite libreria conta poco, quello che conta è che il loro scopo è modificare il codice del gioco, dall'interno, sfruttando punti di accesso di minecraft, quali la possibilità di avere mods, librerie personalizzate o client custom. Una proprietà fondamentale di questi client, è che per la loro natura, non si possono trovare fuori dalla cartella radice di minecraft. Per finire, essendo accedendo internamente al codice di minecraft, si possono trovare tracce di essi nella memoria del processo di minecraft.

Injection clients

I ghost clients sono stati la prima tipologia di cheats ad essere inventati, ma presto gli anticheat diventarono migliori, come anche le persone che controllavano. Ci voleva qualcosa di nuovo e quel qualcosa si è rivelato essere gli Injection clients. Questi clients riescono ad accedere internamente al gioco, rimanendo però un eseguibile qualunque e non quindi una versione, libreria o mod. Questo li rendeva molto più difficili da trovare ma per capirci qualcosa forse è meglio capire a prescindere il loro funzionamento. Come detto all'inizio di questo capitolo, il gioco di minecraft è scritto in java, e java stesso ci fornisce dei tool con cui noi possiamo interfacciarci con la Java Virtual Machine, ed andare ad implementare le nostre funzioni scritte in linguaggio nativo (c/c++). Queste librerie sono JNI e JvmTI, rispettivamente Java Native Interface e Java Virtual Machine Tool Interface. Non mi soffermerò troppo su jvmti, ma vi basta sapere che è la versione aggiornata e più memory safe di JNI, che è invece molto vecchia come interfaccia. Tralasciando tutto questo, gli injection clients sono delle librerie dinamiche (.dll), che si attaccano al processo di minecraft, si iniettano al suo interno e grazie all'interfaccia fornita dalle librerie JNI.h e jvmti.h, riescono a modificare il codice del gioco a loro piacimento. É fondamentale capire che gli injection client sono solo e soltanto delle DLL. I cheat maker però per rendere l'utilizzo del cheat più user friendly, hanno creato dei semplici eseguibili che si occupano di injectare la dll, che è l'effettivo cheat, all'interno del processo di minecraft. Modificando il codice del gioco dall'interno, si possono trovare tracce di questi client nel processo di minecraft, ma è più difficile trovare altre tracce perché non sono limitati alla cartella radice di minecraft. Un tempo, si usava dire che gli injection client erano tali, perché avevano la GUI all'interno della finestra di minecraft. Col tempo è stata totalmente smentita questa cosa ma vorrei approfondire e dare un po' di crediti a questa vecchia "definizione". Non è del tutto sbagliata, ma per capire bene cosa intendo dobbiamo sempre guardare tutto dal lato del programmatore. Quando io vado a creare una gui interna al gioco, è perforza una modifica del codice originale di minecraft, perché tutti i rendering dei vari elementi grafici sono gestiti dal gioco stesso. Anche le più semplici overlay si attaccano internamente, anche se in modo differente, al gioco, e sono quindi

possibilmente individuabili nella memoria del processo. Una GUI esterna però non implica che sia un external client, questo perché come ho spiegato, il cheat in se per se è un .dll, che viene iniettato nel processo del gioco, ma il programmatore potrebbe non avere voglia di fare una gui interna al gioco, così sfrutta l'eseguibile che usa per iniettare la libreria .dll come GUI, quindi interfaccia per il cheat. Questo breve chiarimento lo volevo fare per sottolineare quanto gli argomenti dei controlli e della programmazione sono intrinsechi, perché alla fine sono tutti modi diversi di approcciare lo studio del computer e dei sistemi operativi. Come breve riassunto, voi potete dare come definizione la seguente: L'injection client è un client che agisce direttamente sulla memoria del gioco, modificando classi, il che lo rende individuabile all'interno della memoria del processo.

External clients

Come ultimi (ma non ultimi davvero spoiler) abbiamo gli External clients. La principale differenza e innovazione rispetto ai precedenti injection client, è il fatto che loro accedono in modo diverso alla memoria del gioco. Se prima i ghost e gli injection si riuscivano ad interfacciarsi internamente col gioco, quindi modificando classi e quindi costrutti di alto livello, gli external client agiscono su costrutti di basso livello, sui valori singoli della memoria. Non c'è la necessità che si iniettino all'interno di un processo, perché, almeno parlando di windows, la windows API fornisce funzioni per modificare i valori della memoria dato un indirizzo valido e l'Handle di un processo. Chiaramente non significa però che gli external client sono solo eseguibili. Agendo su valori singoli della memoria, è virtualmente impossibile per noi staffer capire se un player ha usato o no un external client se ci basiamo sulla memoria del processo di minecraft. Come definizione generale si può dire la seguente: Gli external clients sono dei client che agiscono su valori singoli della memoria. Per concludere, questo fatto di poter agire su valori invece che costrutti di alto livello, li limita in funzionalità.

Autoclickers

Gli autoclickers, non sono perforza legati a minecraft in se per se. Essi esistono da molto più tempo, e sono molto facili da nascondere. Interagiscono poco e niente con il processo di minecraft la maggiorparte delle volte, e sfruttano invece le API del rispettivo sistema operativo per simulare i vari click. Non essendo trovabili nella memoria del processo di minecraft, assumono un pattern di ricerca nel pc simile a quello applicato agli external clients, ma bisogna stare attenti, perché gli external possono essere solo degli eseguibili, ma gli autoclickers possono essere scritti in molteplici linguaggi (java, python e così via), il che aumenta la quantità di artefatti da controllare e casistiche possibili. Come sotto categoria degli autoclickers, verrei parlare un po' delle macro e del debounce time. Le macro sono molto semplici come concetto, sono delle istruzioni che si assegnano ad un tasto tramite un software esterno o dedicato al mouse stesso, e si utilizzano, almeno su minecraft, per aumentare la velocità di click del player. Il debounce time invece è un argomento caldo sulla quale discutere, ma da chad jitterclicker, vorrei chiarire un po' sulla posizione del debounce time come un cheat. Il debounce time è una feature dei mouse che è stata implementata tanto tempo fa, per prevenire un problema con gli switch dei mouse, ovvero essendo essenzialmente una molla, quando cliccati rischiavano di rimbalzare ed inviare al computer un secondo, terzo, quarto click e così via, senza però che l'utente stesse cliccando appunto tutte quelle volte. Il debounce time, è il tempo in cui il software del mouse, aspetta prima di inviare un altro segnale di click. Quando un click viene registrato, parte un timer che dura tot ms (generalmente 10 o 16 millisecondi), e che evita qualsiasi tipo di multiple click accidentale da parte del mouse. Questa feature su alcuni mouse è modificabile o totalmente assente, e che permette al player di cliccare a velocità assurde, facendo utilizzo di particolari metodi che triggherano un double click, tutte sotto la categoria di mouse abuse. Su alcuni server, il debounce time sotto a 10ms è bannabile, proprio perché viene considerato un vantaggio sleale nei confronti del player medio, che ha un mouse dei cinesi e fa al massimo 10 cps non per il

debounce time ma per la qualità dello switch palesemente indonesiano. Apparte tutto, su determinati server viene considerato cheating il fare uso di click sotto la categoria mouse abuse, quando si è in possesso, a prescindere dalle impostazioni del mouse, un mouse della roccat o bloody, perché questi due, come poi anche altre marche, non sono riuscite ad implementare per bene il debounce timer, rendendo il tutto molto sleale per il player che si trova contro la persona con tale mouse. Detto questo il debounce time si può modificare solo esclusivamente dal software del mouse, se esso lo permette in primo luogo. Non esistono software esterni, anche perché il firmware di ogni mouse è unico oltre che proprietario della società che lo produce.

Conclusioni

Abbiamo visto in modo si generale, ma anche approfondito, quali sono i possibili cheats che dobbiamo cercare su minecraft, e come riconoscerli, le loro caratteristiche e tanto altro. Nelle prossime parti vedremmo invece robe più correlate al sistema operativo, e quindi informatica forense in generale. Grazie della lettura e buon proseguimento.

Prefetch

Guida approfondita del prefetch e del suo uso in forensica

Il prefetch è un sistema pensato per originariamente per aumentare la velocità con cui un programma si avvia, e tutto questo è possibile perché ad ogni avvia di un programma, un file prefetch (.pf) viene salvato nella cartella "C:\Windows\Prefetch" con nome il nome dell'eseguibile, seguito da una serie di 8 cifre esadecimali. Questi due blocchi sono separati da "-". All'interno di questo file prefetch, vengono immagazzinate le seguenti informazioni:

- Il nome dell'eseguibile
- La hash del programma (che sarebbero le 8 cifre esadecimali)
- Le dimensioni del programma originale
- La versione/i di windows supportata dal programma
- Il numero di volte che è stato eseguito (in generale, non nella sessione del pc)
- L'ultima esecuzione (Data formato yyyy-mm-dd hh:mm:ss)
- Fino a 7 altre ultime date di esecuzione (Data formato yyyy-mm-dd hh:mm:ss)
- Il nome del volume dove il programma era localizzato durante l'ultimo avvio
- Il numero seriale del volume
- La data di creazione del volume
- La lista di cartelle aperte dal programma
- La lista di file (risorse) caricate dal programma

Queste informazioni sono molto utili per permettere a noi di avere una timeline abbastanza accurata dei programmi che sono stati eseguiti nel tempo. Tramite programmi come WinPrefetchView, noi possiamo analizzare in modo accurato i file presenti nella cartella prefetch, selezionarli e vedere ulteriori loro caratteristiche, come la lista di risorse di cui il programma ha fatto uso. Se invece dobbiamo analizzare in modo ancora più avanzato il prefetch, possiamo usare tool come PECmd, che ci permette di vedere ogni informazione di un file prefetch o di un insieme. Come suggerisce il nome quest'ultimo è un tool per command line, che risulta comunque molto potente e pieno di informazioni. A volte potremmo riscontrare errori nell'analisi di questi artifatti. Nel caso l'utente abbia disabilitato il prefetch, bisogna controllare la seguente chiave di registro, che è responsabile per lo stato attivo/inattivo del prefetch:

- "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management\PrefetchParameters".

Il servizio che si occupa della gestione del prefetch è il “sysmain”. La persona sotto controllo potrebbe aver stoppato il servizio. Nel caso controllare con il comando “sc query sysmain”, per controllare lo stato di esso. A volte però la persona controllata potrebbe giocare uno scherzo con i permessi. Prima cosa bisogna controllare che i permessi della cartella permettano la scrittura di file al suo interno. In caso contrario la persona potrebbe usare un noto metodo bypass detto cacls. Per verificare l’esecuzione del metodo cacls, abbiamo delle query al journal che possiamo fare per verificare la modifica dei permessi alla cartella del prefetch. A volte però invece di permormare modifiche di permessi su tutta la cartella, il controllato potrebbe modificare la visibilità di un singolo file prefetch, per non dare nell’occhio. In questo caso bisogna controllare che non ci siano file nascosti, con il comando “dir /ah”, che mostra ogni file, indipendentemente dalle impostazioni sui permessi di visualizzazione.

I processi e la memoria RAM

Una guida al mondo dei processi e di come la memoria ram viene gestita

I processi sono la parte più importante nel mondo dei controlli e della forensica digitale. I processi sono le istanze di un programma (.exe per windows) in esecuzione. I processi vengono gestiti da vari servizi, quello più noto è il PCB (Process Control Block) oppure il TCB (Thread control block). Questi gestiscono rispettivamente processi e thread, hanno informazioni sulla loro memoria, il loro ultimo avvio, le loro risorse etc. Una applicazione come "task manager" ci indica tutti i processi che sono in esecuzione ma si limita ad indicare i processi e il loro impatto sull'hardware (percentuale di ram usata, cpu etc..). Per avere una overview più dettagliata possiamo usare programmi come Process Explorer dalla sysinternal suite etc.. anche se il più utilizzato in ambito controlli è proprio Process Hacker (ora System Informer). Una informazione da notare e che parlando di windows, stiamo interagendo con un sistema proprietario e non saremo mai in possesso del codice sorgente di tale sistema operativo, quindi le nostre capacità di indagine si limitano a questi programmi offerti da queste persone. Non si riesce ad ottenere raw data se non quello che fornisce la windows API, che a volte da problemi di permessi, ma questo lo vedremo parlando di vari processi più avanti. Quando un programma viene avviato, viene creato un processo, che contiene le seguenti informazioni (P.S. non sono tutte ma sono la maggior parte):

- Il file eseguibile da cui deriva il processo
- Il path di quel file
- Il processo parente
- La console parente
- Gli argomenti con cui è stato avviato
- La directory a cui sta accedendo
- I threads
- I moduli
- La memoria stessa del processo che gli viene allocata
- Il programma stesso viene salvato nella memoria ram
- Gli handles

Cerchiamo di spiegare ora queste varie informazioni riguardo al processo. Piccola premessa, tutte queste informazioni sono riguardo hai processi che sono in esecuzione, quando un processo viene terminato, le informazioni elencate li sopra non sono salvate, almeno non raw. Sistemi come il prefetch salvano alcune di quelle informazioni con obbiettivo di ottimizzazione ad un successivo avvio del medesimo programma, ma le informazioni raw così non sono salvate da nessuna parte,

anche perché la ram è appunto una memoria volatile, oltre che incredibilmente limitata rispetto alla dimensione dei programmi.

Processo parente

Il processo parente, è il processo che ha avviato quel programma. Se noi facciamo doppio click su un eseguibile di qualsiasi tipo, il processo parente di quel processo sarà “explorer.exe”. Una cosa molto importante è il fatto che processi particolari come i servizi, hanno sempre dei processi parente noti, e quindi un ipotetico processo che prova a fingersi un servizio di sistema è facile da notare quando esso non ha lo stesso processo parente del servizio emulato e/o di qualsiasi servizio comune. Il 90% dei servizi hanno come processo parente “services.exe”, oppure “svchost.exe”. Servizi come il “csrss.exe” e “wininit.exe” non hanno un processo noto come parente, perché essi sono avviati direttamente dal sistema operativo.

Argomenti di avvio

Gli argomenti con cui un processo è stato avviato, sono le opzioni diciamo, che si possono dare ad un programma per fare varie cose. Alcuni processi hanno lo stesso eseguibile, ma si distinguino per il fatto che sono stati avviati con diversi argomenti e quindi compiono task differenti.

I threads

Nei sistemi moderni, ogni processo ha uno o più thread, ed è un modo di suddividere il lavoro della CPU. Il processo di minecraft ha tanti thread, come alla fine ogni processo di un programma moderno.

I moduli

I file eseguibili non hanno spesso tutto il codice al loro interno. Facendo un esempio semplice, chiunque abbia provato a vedere la cartella di gioco di fortnite, ha visto che ci sono “FortniteWinShipping64.exe”, che è l’eseguibile responsabile per l’avvio di fortnite, ma pesa relativamente poco, una cosa come 200 e qualcosa megabytes. Non sembra normale considerando che il gioco pesa 30 gigabytes e più. Questo perché fa utilizzo delle DLL. Le DLL o meglio conosciute come dynamic link libraries, sono librerie che possono essere caricate da degli eseguibile, e contengono codice ordinario. La comodità è che sono dinamiche, quindi possono essere caricate da un programma più volte e a piacimento. Questo permette un ottimizzazione del programma molto buona, oltre che una suddivisione più accurata delle varie parti del codice. I moduli sono importanti spesso per i cheater. Un esempio noto può essere appunto minecraft. I cheat maker, sia che si tratta di injection o external client, non vanno a mirare il processo “javaw.exe” in se per se, bensì un modulo particolare, il “jvm.dll”. Questo modulo contiene oltre alle istruzioni della jvm creata, quindi il codice del gioco, anche tutta la memoria di esso. Quando un injection client si collega a minecraft, crea un collegamento in particolare con questo modulo, e tramite librerie come “JNI.h” viene a contatto con il codice arbitrario del gioco. Gli external client invece per modificare i valori specifici della memoria, vanno a cercare gli offsets che puntano appunto al modulo “jvm.dll”, perché è dove effettivamente quei valori si trovano. I moduli di un processo sono fondamentali per capire cosa fa.

Gli handles

Meglio conosciuti come risorse in italiano, gli handles sono ogni singolo file aperto da un determinato processo. Quando noi andiamo a fare il self destruct di una mod, istantaneamente noi non possiamo eliminarla o spostarla, questo perché l’handle che il processo di minecraft ha su quel file è ancora attivo. Quindi da li viene poi il metodo bypass di unloaddare una mod, andando manualmente con process hacker a chiudere l’handle che la java virtual machine ha sulla mod,

rendendola così spostabile ed eliminabile. Gli handle riguardano mod come tutte le risorse. Qualsiasi cosa aperta dal processo si trova negli handles.
Alcune di queste informazioni vengono salvate nel prefetch, come i nomi degli eseguibili, le risorse e le directory aperte, ma i prefetch sono manipolabili in vari modi, e di conseguenza possono risultare di poco aiuto a volte. Un artifatto importante nelle indagini / controlli, è andare a controllare la memoria di alcuni processi. All'interno ci potrebbero essere segnali dell'esecuzione avvenuta di determinati programmi che noi valutiamo poi come cheats. Ora facciamo una breve lista di processi / servizi.

"Explorer.exe"

Forse il più noto, l'explorer è la shell che ci permette di interagire graficamente con il sistema operativo, e ha tante informazioni importanti. Osservando come la memoria del processo si comporta, si è notato che ogni file visualizzato nell'explorer appare nella sua memoria, come altre cose tra cui il PCA Client, utile per vedere gli ultimi 10 processi eseguiti (metodo deprecato ma comunque utile da controllare)

"csrss.exe"

Il "client server runtime process" è forse l'artefatto più importante di tutti. Ogni eseguibile, quindi .exe .dll etc.. viene trovato nella memoria di questo processo. Sembrerebbè quindi il metodo finale, purtroppo per noi però la windows API, come accennato prima, può dare problemi di permessi quando si tratta di accedere alla memoria, e questo è uno di quegli esempi. Molto spesso neanche process hacker può accedere a tale memoria e i programmi come eseguibili per ss tool si possono scordare di avere l'accesso a questa parte di memoria. Inoltre possiamo provare che un programma è stato eseguito, ma non quando, e in questi casi nei controlli è importante capire se un cheat era in istanza oppure no. Apparte tutto comunque rimane un processo utilissimo per i nostri scopi, nonstante la sua difficoltà nell'accederci. Metodi in sviluppo ci permettono di dumpare tutta la memoria RAM e isolare quella di determinati processi. Chissà in un prossimo futuro potremmo automatizzare il controllo del csrss e approfondire la nostra conoscenza di questi processi low level.

"msmpeng.exe"

Questo è il processo del servizio anti-malware del windows defender. Un processo poco conosciuto e poco esplorato dai methods finder, ma pieno di potenzialità, perché facendo parte della suite di antivirus di windows, ha informazioni su tutti i programmi eseguiti e quindi possibili flag di cheats in particolare.

Conclusione

Anche se questi sono 3 processi e ce ne sono tantissimi altri che vengono controllati, ho scelto quelli base perché almeno secondo me lo studio dei processi serve fino ad un certo punto. Il mondo delle "stringhe" magiche di process hacker è un mondo superficiale e che si basa su tanti false flag e metodi volatili, che un giorno vanno e altri no. Quando si ha a che fare con la memoria RAM, bisogna realizzare che volatile non è un attributo messo li a caso. Le stringhe sono si utili magari nel loro piccolo, ma non sono niente in confronto allo studio accurato di artefatti più importanti e che ci danno un overview generale su cosa è successo nel computer nel tempo. Lo studio della memoria RAM in modo avanzato sta andando avanti e si stanno scoprendo nuove cose. È un mondo complicato e difficile da tradurre in informazioni significative per noi, ma nonostante ciò è comunque un argomento fondamentale nelle indagini forensi che noi chiamiamo superficialmente controlli. Detto questo rimanete aggiornati e curiosi. Non vi limitate alle stringhe private e robe del genere. Cercate di capire come un computer funziona, per quindi poi ricavare informazioni 10 volte quelle che una banale stringa su process hacker ti può dare.

I file system e il journaling

Un viaggio all'interno del mondo dei file system e come funzionano

Cos'è un file system. Essenzialmente un file system è quel sistema di gerarchie che permettono al sistema operativo e di conseguenza a noi, la suddivisione in cartelle, i permessi dei vari file e l'ordine generale di esso. I file system sono lo scheletro di un pc e ogni sistema operativo adotta uno o più file system supportati. Alcuni file system sono: NTFS, FAT32, FAT16, exFAT, ext4, ext3, ext2, APFS, Mac FS, ReFS, etc …
Noi in questa serie di introduzioni ai controlli ed alla scienza più grande dietro tutto questo che è l'informatica forense, parleremo per semplicità di Windows, che adotta come file system il NTFS (New Technology File System).

MACB Timestamps

Prima di parlare di journaling e concetti più complessi, parliamo di una feature che viene sottovalutata ma è fondamentale la conoscenza e comprensione adeguata di essa per continuare. I timestamp sono delle date più orari salvati per scopi. I MACB timestamp sono una serie di 4 timestamp fondamentali che indicano degli eventi basici. Ecco il significato di MACB e i timestamp indicati:

- (M) Modified / Modificato, l'ultima volta che il contenuto del file è stato cambiato
- (A) Access / Accesso, l'ultima volta che il file è stato aperto (sembra utile in verità irrilevante)
- (C) Changed / Cambiato, indica un cambiamento del suo indice sulla Master File Table ($MFT)
- (B) Birth / Creazione, il momento in cui il file è apparso sul disco

Il timestamp access è irrilevante perché con tutti i servizi per compatibilità e altro, un accesso del file può avvenire anche con delle applicazioni in background, il che rende il tutto molto inaffidabile quando si dimostra l'esecuzione di un file, e quindi porta a false flag.

Master File Table

La Master File Table è un file che indica ogni singolo file presente sul disco fisso. Il parsing di tale artefatto è fondamentale per una conoscenza accurata dei file presenti sul disco.

$USNJrnl

Il file che contiene il comunemente noto "journal" è il $USNJrnl, un file che si trova nella cartella C:\$Extend, cartella nascosta e inaccessibile attraverso explorer normalmente. Il journal del file system è un log di ogni singola attività che è stato possibile registrare sul file system. Qualsiasi spostamento di un determinato file, un cambiamento di dati, una rinominazione o una eliminazione; tutto viene salvato sul journal. Una particolarità di questo artefatto, è che il file $USNJrnl in se è vuoto, ma esso ha due stream di dati alternative che contengono le seguenti informazioni:

- $Max: contiene informazioni come la dimensione massima del journal
- $J: contiene le informazioni effettive, quindi tutti gli USN records dei vari eventi

Tali informazioni vengono comunemente lette attraverso l'utilizzo dell'utility di sistema fsutil. Non è l'unico modo per leggere tali informazioni. Esse si possono leggere anche tramite utility esterne tipo OSForensic, etc…

$LogFile

Il $LogFile è un file particolare che è simile in funzionalità al $USNJrnl, ma invece di salvare USN Records, salva i cambiamenti ai metadata dei file, diverso dal salvare cambiamenti al file in se. Questo comprende i timestamp MACB del file relativo a varie casistiche e collegamenti alla Master File Table. Questo artefatto è estremamente utile e prezioso, anche se è particolarmente difficile da parsare e come argomento non c'è tanta documentazione. L'unico tool che parsa questo artefatto conosciuto almeno da me in questo momento si chiama NTFS Log Parser. Come tutti i tool indicati ci sarà una pagina apposta con i vari download.

Conclusione

Il filesystem è una parte da analizzare interessante, perché il tutto sta nel prendere queste informazioni nude e spoglie, e collegarle per avere un knowledge basico di come i file si sono spostati nel tempo. È un argomento vasto e ne sto coprendo solo la superficie, quindi di conseguenza invito a informarsi da ste basi e continuare ad interessarsi a questi argomenti. Qui chiudo.

Registri ed Eventi di windows

Guida approfondita sui registri, gli eventi e il loro utilizzo nei controlli

Abbiamo visto come analizzare i processi, il disco fisso, il prefetch e tanto altro nel dettaglio, ma ancora una o meglio due cose fondamentali mancano all'appello. I registri e gli eventi di windows. Sono entrambi degli artefatti che sono implementati dentro windows, e in questo capitolo vedremo cosa sono, come utilizzarli e alcune peculiarità.

Registri

I registri di windows sono stati introdotti nella versione 3.1, ed è essenzialmente un database interno al sistema e condiviso. Gli sviluppatori possono sfruttare questa risorsa per immagazzinare valori, sotto forma di "CHIAVI". Questa funzione viene usata sia da applicazioni di terze parti, sia applicazioni, programmi e servizi di windows. Questo significa e appunto ci indica che delle informazioni importanti di vari servizi, sono immagazzinate nei registri e noi nel capitolo "Metodi bypass e come trovarli" avremmo tutte le reference alle varie chiavi di registro specifiche per alcune informazioni. I registri possono essere visualizzati dalla applicazione di prima parte di windows. Per farlo semplicemente richiamare "regedit" dal pannello run (tasto win + R). Altre opzioni possono essere Registry Explorer di Eric Zimmerman, RegScanner, Advanced Regedit e così via. Un artefatto molto semplice nel suo cuore ma forse la parte veramente complicata che viene tralasciata dai vari staffer è quella di sapere le singole chiavi di registro a cosa fanno riferimento, oltre a da dove provengono e cosa significano per noi che dobbiamo ultimamente provare se un player sta usando i cheats o meno. Senza ombra di dubbio un argomento interessante.

Eventi

Gli eventi di windows sono sempre parte del sistema operativo, e come tante altre cose, ultimamente gli eventi sono stati pensati per registrare vari tipi di attività. Vari davvero, perché di eventi ce ne sono di tutti i tipi, ma piuttosto di vedere gli eventi nel singolo, cosa che accadrà nella parte più orientata alla pratica, in questo capitolo ci limitiamo a fondare delle buone idee su come questi eventi funzionano e vengono usati. Gli eventi vengono tutti salvati in file peculiari, con estensione .evtx. Questi file, sono dei database specifici per essere letti in un particolare modo, e per immagazzinare quindi i log dei vari eventi. Gli eventi più superficiali li possiamo visualizzare sul visualizzatore di eventi di windows, che viene evocato con la keyword "eventvwr" nel menù di run (tasto win + R). Questa applicazione ci permette di esplorare tutti gli eventi salvati nella cartella "%SystemRoot%\System32\Winevt\Logs*". I file .evtx è un formato si usato da servizi di windows, ma può essere utilizzato anche da applicazioni di terze parti, per tenere traccia di eventi importanti per il loro utilizzo. Noi usiamo questo artefatto per provare la occorenza di un qualche metodo bypass specifico o l'esecuzione di file etc. Tutti argomenti che riguardano comunque la sezione pratica ma come per i registri sopra spiegati, è un artefatto molto semplice e generico, che però nello specifico a tanto potenziale, ma viene spesso tralasciata tutta la teoria, quindi le persone usano a caso il visualizzatore eventi senza neanche sapere probabilmente cosa stanno cercando o cosa hanno trovato, quello che significa per loro e per il player stesso.

Conclusioni

Alla fine questi due argomenti hanno in comune una cosa, ovvero il fatto che sono degli artefatti che di un utilizzo generale ce ne si fa ben poco. Sono principalmente usati per cercare evidenze

specifiche a determinate conclusioni e quindi non molto importanti a livello di teoria, il che le lascia inesplorate da questo punto di vista. Questo capitolo ci è servito invece ad esplorare la teoria dietro questi artefatti e vi invita a seguire il prossimo capitolo, che entra nello specifico e analizza ogni singolo metodo bypass e/o detect mai trovata. Come sempre questa guida mira ad essere perfetta, ma come tutte le cose ultimamente non lo sarà mai e nemmeno voi, quindi informatevi, esercitatevi ed ultimamente miglioratevi. Grazie della lettura.

Guida ai tools manuali

Guida approfondita sui vari tools manuali che si possono usare durante un controllo

Come descritto brevemente nella introduzione, per aiutarci nel controllo, vengono in aiuto diversi tool, automatici e manuali. Essendo i tool automatici comunque per definizione e design, semplici ed intuitivi da usare, in questo capitolo ci concentreremo solo ed esclusivamente sui tool manuali.

WinPrefetchView

WinPrefetchView è forse il uno dei tool più usati dagli staffer nei controlli. Permette di visualizzare tutti i file prefetch e le informazioni di essi con un interfaccia grafica abbastanza user friendly. Fa parte della suite di programmi NirSoft e funziona praticamente su ogni pc (anche quelli 32bit). Le informazioni sono tante ma non sono del tutto complete, per avere una overview piena delle informazioni di un file prefetch, bisognerà usare altri tool. Una cosa fondamentale da tenere a mente quando si usa winprefetchview, è quella di ordinare i file per modifica più recente, così che si può avere una timeline mediocre dei programmi eseguiti sul pc. Riassumendo, serve principalmente a vedere le informazioni principali dei file prefetch (index, dipendenze e risorse, path degli eseguibili etc.).

Process Hacker

Questo programma è forse il più conosciuto e il più utilizzato tra tutti i programmi. Fin dai primi giorni questo particolare tool è stato usato per aiutare lo staffer nei controlli ed ha sempre saputo fornire una quantità notevole di informazioni. Process hacker può essere considerata come una versione super avanzata di gestione attività. Ha tutte le funzioni di un normale gestione attività, ma appena andiamo a fare tasto destro su un processo notiamo perché questo tool è utilizzato da tutti gli staffer nel mondo. La quantità di opzioni che offre è quasi troppo, ogni informazione di un processo, ogni peculiarità, anche quelle apparentemente inutili, sono visualizzate da questo fantastico programma. La funzione più utilizzata è quella dell'esplorazione della memoria del processo, dove noi possiamo filtrare delle stringhe all'interno della memoria di un processo, una feature che verrà abusata nel tempo per creare nuovi metodi, a destra e a sinistra. Senza dubbio sono tutte informazioni utili quelle che si ricavano dai processi, ma non ci si dovrebbe mai basare su delle prove che sono, per loro stessa natura, volatili. A lungo, specialmente verso gli inizi, gli staffer si limitavano a controllare il .minecraft, qualche stringa su un processo qua e la e finivano il controllo. Ora queste funzionalità ci aiutano molto nei controlli, ma finalmente non sono l'unico modo per noi di dimostrare se un player sta cheattando o no. Di process hacker e delle funzioni legate alla memoria ne ho già parlato in modo esaustivo nella parte più teorica, sui processi e la memoria ram, perciò ora mi soffermo su altre due funzionalità molto utili su process hacker. La prima è quella di vedere gli handle di un processo. Questo significa che ogni risorsa caricata da quel processo in particolare viene elencata sul programma. Questo è molto utile per verificare l'esecuzione di metodi bypass come le mod unloaddate ma questo lo vedremmo nel capitolo successivo. Un altra feature, invece questa più per i cheater, è quella di avere la possibilità, con process hacker 2 (nel 3 è stata rimossa), di iniettare delle librerie .dll all'interno di un processo. Inoltre permette di vedere l'utilizzo che fanno i vari processi di tutte le risorse del pc, questo include dischi, network (con tutte le connessioni aperte dai vari processi) e tanto altro.

Everything

Santa sia voidtools, la società che ha sviluppato questo fantastico tool, che aiuta noi a trovare i file in tempo minimo, e con minimo sforzo. Il programma carica la master file table nella sua memoria e la interpreta, così che noi riusciamo a cercare in modo veloce all'interno di tutto il file system senza troppi problemi. É questione di scaricare l'installer, installarlo ed aprirlo e si riuscirà a trovare qualsiasi file o cartella in tempo zero.

USB Deview

USB Deview è un programma sviluppato da NirSoft, che ci permette di visualizzare tutti i dispositivi usb che sono stati collegati e che sono collegati. Questo ci potrebbe avvertire se l'utente usava due mouse, e ne ha disconnesso uno, oppure se ha cambiato mouse per evitare di trovare le macro impedendo l'accesso al software del mouse, insomma un tool molto utile per capire sempre, cosa è successo e quando, un tema ricorrente in tutti i tool che andremmo ad introdurre qui.

Fsutil

Essendo un programma incluso con windows, non c'è nessun bisogno di scaricarlo o altro. Fsutil è una command line utility che ci permette di esplorare il $USNJrnl:$J, ci lista tutte le sue entry e quindi cosa è stato registrato in quanto eventi, cambiamenti al file system e al disco.

WinLiveInfo

Windows 10 Live Information viewer, o meglio conosciuto come winliveinfo, è un tool open source, che si occupa di interpretari una vasta gamma di artefatti che indicano le più svariate cose. Questi artefatti vanno dalla lista completa di processi attuali, alle entry del registro BAM, alla MRU cache e così via. É un tool di estrema utilità e valore in quanto alle prove che egli ci fornisce, oltre a rendere più veloce il controllo.

PECmd

Una utility per command line, che permette di visualizzare ogni informazione di uno o più file prefetch.

OS Forensics

OS Forensics è un programma professionale (a pagamento) che offre in un solo tool, una vastissima gamma di sotto tool, tutti con lo scopo di ottenere informazioni riguardo il pc e cosa è successo, quando, dove e come. Si protrebbe considerare un tool a tutto tondo, offre davvero ogni funzionalità. Include un explorer della memoria dei processi, un parser per i file prefetch, un generatore di timeline dove analizza molti artefatti allo stesso tempo, per generare una timeline complessa ma sopratutto completa, un journal viewer, una ricerca di estensioni spoofate e così via. Insomma ogni tool immaginabile si trova all'interno di questo programma, il cui utilizzo è anche relativamente facile, grazie alla sua interfaccia user friendly.

FTK Imager

FTK Imager, fornito da access data, è un tool che permette l'analisi di immagini di dischi fissi. Montando al suo interno il disco del player controllato, abbiamo accesso ad ogni file, compresi quelli di sistema. Questo tool viene utilizzato per due scopi principali: L'estrazione di artefatti appartenenti al sistema operativo, quindi nasconsti e non accessibili normalmente; mentre l'altro è ottenere un dump intero della memoria ram.

NTFS Log Tracker

Un tool che ci permette di convertire i file raw della master file table, del journal e del $LogFile in file csv, leggibili da un umano e quindi interpretabili da programmi appositi per leggere file di quel genere.

Timeline Explorer

Un programma molto potente ma semplice, che ci permette di visualizzare e interpretare i file csv, che possono essere considerati un po' come la versione lite e primitiva dei file excel. Riesce ad aprire file di grandi dimensioni, molto importante per parsare artefatti complessi come quelli del journal e della master file table.

JD-Gui

Un programma che ci permette di visualizzare il contenuto degli archivi .jar, e quindi ci permette di decompilare tutti questi file java. Ci lascia dare un occhiata al codice sorgente, ammeno che esso non sia offuscato, ma comunque è fondamentale per capire se artefatti come le mod o altro sono legit o meno.

MemProcFS

Un programma che ci permette di trasformare un dump della memoria ram, in un file system esplorabile, con tutte le informazioni che potrebbero servirci, anche con delle feature già implementate per la conversione a csv e così via. Fondamentale per avere ogni aspetto di cosa contiene la memoria ram in quel preciso momento. Questo utlimo è una command line utility, ma solo un comando è già abbastanza per avere una serie di prove molto schiaccianti e affidabili

Registry (Windows)

L'explorer default per le chiavi di registro di windows, un tool fondamentale e incluso appunto con windows, che ci permette di visualizzare la gerarchia dei registri, insieme al valore delle loro chiavi

Programmi non menzionati

In questa guida ai vari tools fondamentali, abbiamo saltato molti altri tools che sono si utili, ma almeno io penso siano di seconda importanza rispetto a questi elencati. Una maestria dei programmi sopra elencati è necessaria per condurre dei controlli soddisfacenti. Detto questo lascio qua un elenco di programmi non elencati che potreste incontrare nel vostro viaggio nel mondo dei controlli:

- Hibernation recon
- Last Activity View
- User Assist View
- OpenSave File View
- Registry Explorer
- Kape
- EZ Viewer

- System Informer (Process hacker 3)
- …..

Questa lista come ho già specificato prima, potrà essere modificata in futuro, ci sono così tanti programmi ed è quasi impossibile listarli tutti, ma noi comunque ci proviamo.

Conclusioni

In questo capitolo abbiamo visto tutti i tool principali, insieme a delle informazioni riguardo a loro e alcune spiegazioni sul loro utilizzo. Come specificato in ogni capitolo, queste informazioni sono pensate per essere il più complete possibili ma niente è perfetto, perciò informatevi, e sopratutto rimanete interessati. Buon proseguimento.

Metodi bypass e come trovarli

Guida approfondita su ogni metodo detect e bypass di cui la community è a conoscenza

In questo capitolo della guida, abbiamo un elenco accurato con ogni metodo detect e bypass mai stato usato nella community dagli inizi ad oggi. Questa parte non sarebbe stata possibile senza l'aiuto di alcuni collaboratori, tutti elencati con i crediti rispettivi nella conclusione. Questa parte della guida andrebbe visualizzata solo e soltanto se i concetti base sono stati ben assimilati. É importante perché questa parte è molto pratica e le spiegazioni non possono essere troppo lunghe, quindi è necessario che le tecniche che si vorranno usare in controllo, siano almeno capite al massimo delle loro potenzialità, ciò richiede dunque uno studio attento della teoria generale. Detto questo dividerò questo capitolo in due grandi categorie:

- Metodi detect generici
- Metodi bypass (con rispettivi detect)

Senza perderci in ulteriori argomenti, buona lettura. P.S. sono tutti metodi per sistemi Windows.

Metodi detect generici

In questa parte, troviamo metodi detect che non sono strettamente correlati a particolari metodi bypass. I detect specifici per i bypass si trovano nella stessa sezione, se esistono …. É tutto diviso in sottocategorie quindi non dovrebbe essere un problema orientarsi.

Cartelle particolari

Ci sono cartelle molto utili che noi possiamo andare a controllare per avere delle prove di utilizzo cheats, o degli artefatti da analizzare. Per viaggiare a queste cartelle è suggerito l'utilizzo dell'utility di windows, run (tasto win + r). Ecco un elenco di queste directories:

- C:\$Recycle.bin
    - É la cartella dove i file del cestino si trovano. Si utilizza per controllare l'ultima eliminazione e svuotamento del cestino, ciò si va con l'attivazione in primo luogo della visualizzazione di elementi nascosti e, per andare sul sicuro, opzioni > visualizzazione > nascondi file protetti di sistema deve essere off come opzione nell'explorer. Fatto questo possiamo visualizzare l'ultimo svuotamento del cestino.
- shell:recent
    - É la cartella degli oggetti recenti di windows, una funzione che ci permette di visualizzare, se attiva la stessa, alcuni file con cui windows e l'utente ha avuto a che fare. A volte ci si potevano trovare .dll etc…, ma rimane al giorno d'oggi un metodo detect deprecato.
- %temp%

- É la cartella dei file temporanei di windows e si utilizza principalmente per trovare il file .dll che rilasciano gli autoclicker in java che utilizzano la libreria con il medesimo nome: JnativeHook. I metodi per il Jnativehook li vedremmo nella parte successiva, ma per ora ci basta sapere che viene utilizzata principalmente come detect per questo artefatto.

- Prefetch

  - Prima che qualcuno lo dica, no non è un errore scrivere sulla casella di run “prefetch”, questo infatti ci apre la cartella che windows utilizza per immagazzinare i file prefetch. Vedremmo più avanti metodi bypass che vengono applicati a questa cartella e come averne le prove.

Macro e debounce time

I mouse sono un grande argomento di dibattito spesso, io stesso molte volte mi trovo a discutere con alcuni player di varie ragioni per cui le loro impostazioni sono bannabili etc… Chiaramente ho fatto un po’ di ricerche quindi oggi lascio in questa sottocategoria tutte le nozioni sui mouse, il detect di varie cose etc.. Partirei proprio dall’argomento più semplice, ovvero le macro. Partendo dal fatto che programmi come Xmouse e altri simili sono bannabili (Timer resolution non rientra in questa categoria), i software proprietari dei vari mouse salvano le loro impostazioni delle macro nelle seguenti cartelle:

- Logitech

  - %localappdata%\Logitech\Logitech Gaming Software\settings

  - %localappdata%\LGHUB\settings

- Red Dragon

  - %homepath%\Documents\M--- Gaming Mouse\MacroDB: I trattini indicano la possibilità di numeri diversi in base al modello del mouse

- Glorius

  - %appdata%\BY-COMBO2

- Roccat

  - %appdata%\ROCCAT\SWARM\macro\custom_macro_list

- Steel Series

  - %localappdata%\steelseriesengine-3-client\Local Storage\LevelDB

- Razer

  - C:\ProgramData\Razer\synapse\Accounts

  - %localappdata%\Razer\Synapse3\Log

Queste erano le cartelle dei mouse fin ora analizzati. Ora vi parlo due secondi di un argomento molto scottante al momento, sto parlando del debounce time. Partiamo da qualche nozione base prima: Il debounce time è una feature dei mouse che è stata implementata, originariamente, per prevenire un difetto dei mouse: Il tasto dei mouse è essenzialmente una molla, e come tutti noi sappiamo le molle rimbalzano. Se il mouse viene cliccato in un modo particolare, questa molla rimbalza così vigorosamente che il mouse registra più click. Questo problema è noto da tempo come double clicking e sottolineo è un problema nei mouse. Minecraft è uno dei pochi giochi, se non l'unico, in cui la velocità di click fa la differenza tra vincere o perdere, perciò i player cercavano sempre più modi per cliccare più velocemente. Un giorno dei player con mouse difettosi, non disposti di DC prevent (la feature generale che implementa il debounce time in modo da evitare il double click), hanno iniziato a cliccare in modi particolari, e hanno scoperto che in questi modi il mouse registrava più click di quante volte il tasto veniva realmente premuto. Questi metodi fanno parte della categoria che oggi è conosciuta come mouse abuse. Questi metodi vengono accumunati dal fatto che abusano questo problema dei mouse, aka il double click. Il debounce time, non è altro che un timer, che quando il mouse registra un click, parte e durante il tempo impostato su questo timer (generalmente o 10ms o 16ms), il mouse blocca ogni altro input registrato. Alcuni mouse hanno la possibilità di modificare il debounce timer, in modo da abbassare il threshold e quindi aumentare la possibilità di double clicking. Solo quindi con i metodi di click mouse abuse, di cui butterfly click e drag click fanno parte, i player possono abusare di questo difetto del mouse. Venendo a noi, su alcuni server è bannabile il debounce time sotto 10ms. In questo caso per controllare che il player abbia il deobunce time a 10ms e non l'abbia modificato, si procede con un controllo simile a quello delle macro. Prima si controllano i file raw, dopo si apre il software. I mouse che possono modificare tramite software il debounce time sono i seguenti: P.S. Il debounce time è una feature che ogni manifattura di mouse implementa in modo proprietario, perciò non esiste nessuno programma generico per modificare tale funzionalità.

- Glorius
    - I mouse glorius hanno il debounce time modificabile a livello software tramite uno slider, questo implica che la modifica di BYCOMBO-2 come abbiamo visto prima per le macro è tutto quello necessario per trovare qualsiasi modifica prima del controllo delle impostazioni
- Cooler Master
    - Stesso discorso per i glorius, i cooler master implementano il debounce time modificabile tramite slider, e tutto questo si verifica nello stesso modo delle macro
- Roccat
    - I roccat sono dei mouse che implementano il debounce time con uno switch (zero debounce time on/off). Il problema dei roccat è che sono tutti, dal primo all'ultimo, difettosi, e indipendentemente dalla funzionalità indicata, hanno un debounce time base inferiore a 10ms. Se si mette in controllo un player perché cliccava 20+ cps, ha un roccat e sul server il debounce time sotto 10ms è bannabile, allora si può direttamente bannare. Una casistica dove non andrebbe bene è quando, per esempio, metto sotto controllo uno per es. reach, e trovo una situazione simile, ma non ho visto che cliccava 20+ cps. In quel caso non è detto che stava mouse abusando, quindi la prova non può essere ritenuta valida per un ban. Massimo massimo un aggravamente ma finita li.

- Bloody
    - I mouse bloody implementano il debounce time modficabile con uno slider, ma hanno lo stesso problema dei roccat, sono tutti difettosi, e anche se lo slider è a 16ms, il debounce time vero è sotto 10ms. Stesso discorso dei roccat in quanto a motivazioni per ban etc..
- Red Dragon
    - I red dragon implementano il debounce time su alcuni mouse, tramite slider, e il metodo di controllo è il medesimo di quello delle macro
- Mad Catz
    - I mouse mad catz, per quanto rari, hanno questa peculiarità di non avere direttamente nessun tipo di debounce timer e/o double click prevent. Stesso discorso quindi per i mouse roccat e bloody, viene applicato anche qua.
- Mars Gaming
    - I mouse della mars gaming implementano il debounce time tramite slider, e il metodo di controllo è uguale a quello per le macro

Ci vuole, come in tutti i controlli e situazioni in generale, sempre un minimo di contesto e bisogna capire in primo luogo, se quella persona è da sanzionare o meno. Usate umanità quando fate controlli non siate dei robot per farmare ban.

Process Hacker

Process hacker è il tool che in assoluto è il più utilizzato dagli staffer nel tempo. Pur essendo sconsigliato basarsi principalmente su prove provenienti dalle memorie dei processi, non si può negare che siano incredibilmente utili e se utilizzate con destrezza, possono essere un gran time saver durante un controllo, sia per lo staffer che per l'utente. Come tutte le cose ci vuole prudenza, ma in questo caso più che mai, visto che spesso il concetto di memoria volatile e cosa fa process hacker viene spesso percepito nel modo scorretto, o non capito proprio. Ipotizzando che voi abbiate letto attentamente la guida ai processi e la memoria ram, oltre alla guida dei tools e quindi sull'utilizzo di process hacker, senza altri indugi partire con l'elencare i processi e i rispettivi pattern di ricerca, con relative spiegazione al perché le facciamo in primo luogo:

explorer.exe

Il primo processo che si va a controllare, essendo quello che ha il rate di volatibilità più alta, ecco l'elenco di pattern usati:

- Parola (case insensitive) > pcaclient
    - in output ci dà più risultati a volte, ma quello che ci importa è una stringa di output che ha come contenuto, parte della memoria del pca client stesso (Program Compatibility Assistant). Questo ci da una lista degli ultimi 10 programmi eseguiti. Questo è un metodo vecchio e deprecato ma nonostante la sua inefficacia contro tutti i cheat moderni, rimane un ottimo artefatto iniziare un controllo

- Parola (case insensitive) > file:///
    - in output ci ritorna la lista di file che sono stati visualizzati dall'esplora risorse di windows. Questo è molto utile per avere una overview generale, ma bisogna stare attenti a non concentrarsi troppo su questi risultati. Rimangono pur sempre dei risultati con utilizzo deprecato, ciò significa che non sono sempre così tanto utili e spesso la gente si perde nel controllare tali risultati. Chiaramente questa ricerca ritorna una mole di file enorme, perciò su process hacker si può sempre eseguire una ricerca di qualsiasi tipo su un subset di risultati, per chi non lo sapeva.
- Regex (case insensitive) > ^[A-Z]:\\.+:
    - Questo da in output vari risultati, che bisogna analizzare attentamente, ma essenzialmente è un vecchio modo per detectare un metodo bypass (WMIC, lo vediamo dopo). Questa è una stringa deprecata e se ne sconsiglia l'utilizzo

csrss.exe

Questo è forse il processo simultaneamente più utile e inutile che c'è. Utile perché è un servizio di livello molto basso, più basso perfino di cose come svchost o altro, ed è inoltre proprio il servizio che si occupa del paradigma client - server, perciò nella sua memoria possiamo trovare ogni eseguibile mai avviato, da .dll a .exe a .pif etc .. Il problema è che essendo un servizio di livello basso, è molto spesso protetto da antivirus esterni, rendendo la sua memoria inacessibile, o ancora peggio lockato nei sistemi come windows 11 e superiori. Nonostante ciò su sistemi come windows 7 è perfettamente utilizzabile e all'interno della sua memoria con alcuni semplici filtri regex, possiamo trovare tantissime informazioni:

- Regex (case insensitive) > ^[A-Z]:\\.+.exe$
    - Questo filtro trova ogni singolo path degli eseguibili .exe che sono stati avviati nel sistema
- Regex (case insensitive) > ^[A-Z]:\\.+.dll$
    - Questo filtro trova ogni singolo path degli eseguibili .dll che sono stati avviati nel sistema
- Regex (case insensitive) > ^[A-Z]:\\((?!Exe|dll|jar|ini).)*$
    - Questo filtro trova ogni singolo path degli eseguibili che hanno un estensione diversa da .exe .dll e così via, e sono stati avviati. Se qualcosa appare qua, è un ban quasi sicuro.

svchost.exe (-s dps)

Il -s dps è un servizio particolare che si fa parte dell'svchost, e tramite regex troviamo anche qui alcune informazioni utili:

- Regex (case insensitive) > ^!!(?!svchost|dwm|csrss|explorer|taskhostw|ctfmon|rundll32|conhost|lsass|

  usoclient|sihost|dashost|nissrv|smss|sc|servicehost|settingsynchost|consent|dllhost|sppsvc|wermgr).+.exe

    - Questo regex di dimensioni comicamente grandi, trova ogni file .exe eseguito di cui il dps tiene traccia, escludendo dalla lista i nomi dei processi che si trovano tra le parentesi.

- Regex (case insensitive) > ^!![A-Z]+(.*)[A-Z]:

    - Questo regex trova tracce dell'esecuzione di un metodo bypass (WMIC) che vedremmo più tardi. Come metodo è deprecato

- Regex (case insensitive) > ^!![A-Z](.)((?!Exe|dll).)*$

    - Questo regex trova ogni file che è stato eseguito, ma non ha l'estensione di un eseguibile (metodo bypass estensioni spoofate, le vediamo dopo)

Fix dei permessi per process hacker

A volte process hacker non permette l'accesso a memorie di processi di basso livello (eg. csrss.exe). Perciò, per rendere nuovamente accessibile quella regione di memoria, dobbiamo andare nelle opzioni di process hacker / system informer ed abilitare la "kernel-driver" mode, che ci da pieno accesso alle risorse del computer. Questo consente anche ai pc con versioni moderne di Windows di accedere ai vari processi bloccati.

Journal (fsutil)

Il modo più comune per interagire con il $USNJrnl:$J journal è senza ombra di dubbio tramite la utility cli fornita da windows fsutil. Il problema degli staffer è inseriscono i comandi, ma non sanno il loro funzionamento, o non sanno inventare comandi specifici loro stessi. Inventare pattern di ricerca per process hacker è complicato, e serve tanta pazienza, ma per il journal, tutto si limita nel filtrare per bene i vari risultati con il tool findstr.exe. Molto spesso i comandi journal che troviamo in giro sono suddivisi in questo modo:

1. fsutil usn readjournal c: csv
2. |
3. findstr /….
4. (opzionale) > output.txt

Ecco alcune nozioni sui comandi: la prima parte è un comando tutto a se, e essenzialmente quello che fa è tradurre i dati contenuti in modo compresso all'interno del file $USNJrnl:$J, in un contenuto leggibile, per l'esattezza gli specifichiamo di mettere in output, i dati sotto forma di un file csv, di cui dovremmo già sapere tanto, ma per ripassare, è essenzialmente una versione primitiva delle tabelle excel. La seconda parte del comando, è una cosa geniale, perché quella barretta è un alias di nome "pipe" e come suggerisce il nome, ogni output del primo comando viene intubato, fornito come input al comando successivo. La terza parte è sempre un comando, questa volta però filtriamo i dati del csv, con l'utility findstr.exe, fornita anche questa dal sistema. Quando facciamo il findstr, ci sono degli argomenti, come /i /R /c e così via, ecco a cosa servono:

- /i
    - indica se il filtro che specifichiamo deve essere case sensitive o no

- /R
    - indica se il filtro che specifichiamo è una stringa REGEX (regular expression) oppure no
- /C
    - indica la stringa effettiva per cui noi filtriamo. Segue infatti subito dopo un :"filtro"

Alcune volte, in comandi molto lunghi, si effettua più volte il pipe con il | e un altro findstr, che quindi filtra tutti i risultati del comando precendente. Tutto questo può terminare normalmente, o con la sintassi indicata nel punto 4. Tale sintassi indica che l'output deve finire nel file indicato dopo il simbolo maggiore (>). Li si può specificare un path relativo o assoluto, attenzione solo a dove siete nel file system con il command prompt quando fate il comando, specialmente con questa parte finale. Una cosa da sottolineare è che per questo comando, è obbligatorio avviare il cmd in modalità amministratore. Detto questo, ecco alcuni comandi già scritti, molto utili per risparmiare tempo, ma la cosa che suggerisco è inventare sul momento i comandi del journal. Fatevi furbi, poi semplificatevi la vita copiando questi comandi ma mi raccomando prendeteci destrezza.

- fsutil usn readjournal c: csv | findstr /i /C:"0x80000200" /i /C:"0x00001000" /i

  /C:"0x00002000" | findstr /i /C:.exe\^" /i /C:.pf\^" /i /C:.com\^" /i /C:.cmd\^" /i /C:".jar" /i

  /C:.pif\^" /i /C:.bat\^" /i /C:"?"

    - Questo mastodontico comando, indica ogni file di tipo eseguibile e le loro azioni, quindi eliminazione, spostamento o modifica.

- fsutil usn readjournal c: csv | findstr /i /C:"0x00000800 /i /C:"0x80000800" | findstr /i

  /C:"Prefetch"

    - Filtra risultati per un metodo bypass (CACLS), ma essenzialmente cerca modifiche di permessi alla cartella prefetch

- fsutil usn readjournal c: csv | findstr /i /C:"0x--------" /i /C:"0x--------"

    - Trova tracce dell'esecuzione del metodo bypass (WMIC). Trova delle modifiche al flusso di dati di un file.

Powershell

La history dei comandi dati al cmd non è salvata da nessuna parte, ma non si può dire lo stesso di powershell. Per trovare tutti i comandi eseguiti da powershell, basta fare il seguente comando su di essa:

- cat (get-PSReadlineoption).Historysavepath

Cmd

Alcuni comandi da riga di comando sono molto utili per avere informazioni fondamentali. Ad inizio controllo, per esempio, bisognerebbe sempre fare un veloce:

- sc query dps
- sc query sysmain

Questo comando, sc query “nome servizio”, controlla lo stato del servizio, e ci indica se sono stati riavviati o no. Se uno di questi risulta stoppato o riavviato, si può facilmente bannare. L’utilizzo del comando dir è poi fondamentale per trovare metodi bypass con i permessi dei file. Per trovare la maggior parte dei file nascosti, con pochissime eccezioni, ecco due comandi fondamentali (per cmd):

- dir /ar
- dir /ah

Regedit

I registri di windows sono pieni di informazioni importanti, quindi ecco alcune chiavi di registro e raccolte di chiavi da analizzare durante un controllo

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management\PrefetchParameters
  - Questa chiave di registro indica se il prefetch è attivo o meno (0 = inattivo)
- HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\Compatibility Assistant\Store
  - Questa cartella indica un po’ di programmi che sono stati loggati dall’utility di windows per la retro compatibilità (non indica l’esecuzione di essi, solo la loro presenza sul disco).
- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\bam\State\UserSettings\
  - Una delle detect per il metodo bypass (WMIC). Deprecato
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\OpenSavePidlMRU\
  - Indica gli ultimi 20 file aperti e salvati
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\
  - Tutte le estensioni dei file che sono presenti sul pc
- HKEY_CURRENT_USER\SOFTWARE\WinRAR\ArcHistory

- Tutti gli archivi che winrar ha aperto / utilizzato (Chiaramente presente solo se l'utente ha installat winrar)

Event viewer

Nel visualizzatore eventi possiamo trovare tanti eventi generici che ci indicano azioni malevole e quindi possibile ban:

- Registri windows > Sicurezza > 4616
  - Indica un cambio di orario
- Registri windows > Applicazione > 3079
  - Eliminazione del journal
- Registro windows > Sicurezza > 4798
  - Cambio dell'eredità dell'eventvwr
- Registro windows > Sicurezza > 1102
  - Eliminazione di eventi dall'eventvwr

Metodi bypass (con rispettivi detect)

Ora in questa sezione ci concentriamo nella parte dei metodi bypass, con tanto di metodi detect etc… Sapere il nome comune in community di questi metodi, oltre al loro funzionamento è fondamentale per una buona conoscenza per iniziare, e in questa sezione miriamo ad elencare ogni metodo bypass più o meno conosciuto che siamo riusciti a trovare. Ancora una volta grazie ai collaboratori, crediti alla conclusione finale.

Partizioni

Possiamo creare delle partizioni all'interno del disco, in modo da non flaggare in alcun modo il journal del disco di default, per poi successivamente eliminare le partizioni. Questo sarebbe un buon metodo, se non fosse che viene segnalata ogni creazione ed eliminazione di partizione sul database degli eventi:

- Registri applicazione e servizi > Microsoft > Windows > VolumeSnapshot-Driver > 116 o 117
  - Eliminazione o creazione di una partizione

Javaedit

I javaedit, sono delle versioni di minecraft apparentemente vanilla, ma che sono state modificate apposta in modo da contenere, per esempio, una reach integrata, difficile da individuare in primo luogo, ma molto facile da controllare. Essendo i javaedit, molto vicini all'esperienza vanilla, essi provano in tutti i modi a fingersi una versione vanilla, ma una cosa le tradisce tutte: la SHA256. Per trovare un javaedit, bisogna confrontare la SHA256 del client utilizzato dal player, con la SHA256 della stessa identica versione che il player pretende di utilizzare. Esempio: player ha la forge-1.7.10-

29174, noi dobbiamo confrontarlo con la sha 256 della forge-1.7.10-29174, non con la forge-1.7.10-27865 o la forge-1.8.9 o la vanilla 1.7.10 e così via, ci vuole la esatta versione, non una a random sennò si ottengono dei falsi positivi.

Java -jar

Tramite il comando java-jar, noi possiamo avviare qualsiasi file che ha al suo interno, del java bytecode. Questo comando ci evita di utilizzare l'explorer, cosa molto utile ma sopratutto nel prefetch è meno evidente l'esecuzione di un .jar. Il jar può essere rinominato anche con un estensione spoofata, ma l'importante è che il contenuto sia del java bytecode. Non posso avviare un vero eseguibile con java-jar, ma un 7clicker.jar rinominato tagliani.exe lo possiamo avviare. Per trovarlo, basta controllare i file prefetch del java.exe, e nei vari argomenti, controllare per file sospetti.

Wmic

Il wmic è un metodo bypass che consiste nel utilizzare due comandi e una funzionalità particolare di windows. I file possiedono diverse stream (flussi) di dati. Questi flussi possono essere creati a piacimento e possono contenere diverse informazioni. Ciò può essere un problema perché tramite questo metodo noi possiamo mettere cheat come perfino la vape all'interno di un file .txt, senza che esso sembri strano allo staffer o niente, tutti questo perché si trova all'interno di un altra stream di dati, che in primo luogo è invisibile allo staffer. Il metodo bypass sfrutta il comando type, per scrivere tutti i bytes di un file, nella stream alternativa di un altro, e il comando wmic per richiamare i bytes della stream alternativa. Questo metodo si può detectare in due modi principali:

- Analizzare il disco per file con più stream alternative (AlternativeStreamView di NirSoft)
- Controllare il journal, per la modifica anomala di data stream per alcuni file sospetti

Tutti questi due metodi sono molto efficaci, diversi in esecuzione ma pursempre efficaci. Ci sono vari metodi deprecati che includono stringhe su process hacker e noi li sconsigliamo, ma intanto sapete che esistono.

Cacls

Il cacls è una utility per modificare in modo particolare, i permessi delle cartelle e dei file. Questa utility può essere utilizzata per mettere in sola lettura la cartella prefetch e quindi far si che i file prefetch non si registrino, rimanendo però apparentemente indetectata come azione. Piccola premessa prima del detect, il cacls gioca con i permessi di sistema, rendendo il tutto estremamente pericoloso da effettuare, specialmente se non sapete cosa fate. State attenti se lo provate. Il metodo detect però non rompe il sistema operativo, e consiste nel controllare tramite il journal, la modifica di permessi di determinate cartelle, tutto alla fine relativamente semplice. Questa query al journal può servire per trovare eventuali modifiche ai permessi avvenuti alla cartella prefetch:

- fsutil usn readjournal c: csv \ findstr /i /C:"0x00000800" /i /C:"0x80000800" \ findstr /i /C:"Prefetch"

Replace

Il replace è un metodo generico, che consiste nel rimpiazzare un file con un altro, in modo da confondere la persona che ti sta controllando. Esistono tante varianti di replace, ma il modo più facile per individuare ogni movimento sospetto è senza dubbio l'utilizzo del journal. Possiamo usare

le informazioni del journal per vedere che movimenti ci sono stati dentro una cartella o relativi ad un singolo file. Alcune nozioni necessarie per il journal e il corretto utilizzo durante la ricerca di un replace, è l'utilizzo delle file hash e del comando "fsutil file queryFileNameById 0x…..", che indica la cartella nella quale un determinato file, flaggato nel journal, si trovava. Per maggiori informazioni, guarda la parte dei detect generici per capire meglio di cosa si sta parlando.

Spoof estensioni

Un metodo per nascondere l'esistenza di un file eseguibile, cambiando la sua estensione da .exe a .tmp o altro, qualsiasi estensione. Poi utilizzando l'utility di windows "Start-Process", si può eseguire tale file. Il detect è molto semplice, essendo che nella cartella prefetch appare un file che non ha l'estensione .exe, esempio: "TEST-EXECUTABLE.TMP-366708B5.pf". Se si vede una roba del genere, verificare che tale file sia effettivamente un cheat e procedere con le giuste misure.

Godmode

Un tempo, i cheater più divertenti del mondo di minecraft rinominavano il cheat in un modo particolare, facendolo risultare come un utility del pannello di controllo.

Cambi d'ora

Molto spesso i cheater sono simpatici e si divertono a cambiare l'orario del computer, per trollare lo staffer. In questo caso ci sono diversi metodi per trovare il cambio di orario:

- Utilizzo dell'eventvwr:
    - Registri windows > Sicurezza > 4616
- Utilizzo del journal:
    - // TODO

Process hollowing

Il process hollowing è un metodo molto simpatico, dove un processo viene sfruttato per "nascondere" per così dire un altro processo, generalmente il caro e vecchio cheat. Ci sono tanti modi per sgamarlo ma fin ora quelli più generici e affidabili per detectarlo, riteniamo siano due:

- vedere se il path dell'eseguibile indicato in un file prefetch è vuoto o meno. Se esso è vuoto c'è una buonissima possibilità che sia avvenuto un qualche tipo di process hollowing

- fare un "live kernel dump" con process hacker / system informer, e successivamente filtrare i risultati utilizzando un programma di Sysinternals "strings64.exe" e la seguente stringa: "strings64.exe -accepteula -s -n 4 %path del file dumpato con process hacker% | findstr ".exe" > %file filtrato di output%". Successivamente aprire quel file e cercare la presenza della seguente stringa: "nul 2>&1" e voilà, in caso hai dei risultati, controlla tutti i file che appartengono alla stessa riga e vedi se sono dei cheat o meno (spesso li puoi trovare sotto estensioni come .dat o altro).

Registrazioni nascoste

Molto spesso un bypasser poco divertente decide di utilizzare dei software custom o diversi dai soliti OBS, shadowplay etc. che siamo soliti a controllare. Per trovare qualsiasi tipo di registrazione nascosta che sta accadendo in background, ci basta vedere l'utilizzo del disco tramite process hacker. Il disk usage ci mostra tutti i file che stanno venendo letti / scritti in tempo reale, perciò ci basta individuare estensioni video comuni come .mkv, .mp4, .mov etc..

Cmd disabilitato

A volte uno sporco cheater decide di farci uno scherzetto, "disabilitando" il prompt dei comandi. Non avere paura però, non ha disabilitato niente, semplicemente ha deciso di mettere una procedura che spegne il processo del cmd ogni volta che esso viene aperto. Per trovare questa simpatica tecnica, ci basta visitare la chiave di registro: "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Command Processor" e vedere cosa si può trovare nella chiave Autorun. Se c'è roba come taskkill etc.., eliminate la chiave di registro con tasto destro.

Trovare delle DLL sospette

Una questione a lungo dibattuta negli SS è il metodo per trovare i cheat DLL. Dopo tanto tempo e molte discussioni, un metodo è stato trovato. Le DLL non sono altro che librerie dinamiche, e sono molto pericolose se usate per scopi maligni, come virus o altro. Da questa esigenza nel campo dei virus e malware, si è iniziato ad implementare un sistema di firme digitali, dove le DLL certificate e ufficiali vengono firmate digitalmente per contrassegnare il fatto che sono sane e non dei possibili virus o altro. Una DLL con mancata firma digitale è spesso simbolo di uno sviluppatore scarso che non si è preoccupato di firmare digitalmente la propria libreria. Perciò per trovare tutte le DLL sospette, conviene analizzare il processo "csrss.exe" e dumpare tutte le stringhe con i filtri C: e .dll. Successivamente assicurarsi che i risultati siano in linea e tutti ordinati, per poi procedere e utilizzare il programma BinarySignatureStatus, per verificare la presenza di una firma digitale e tutti i suoi attributi. Dopo aver ottenuto tutti i risultati, ricercare le DLL dove la firma non è presente e controllarle manualmente.

Chiavette esterne

Una tecnica molto comune tra i bypasser, almeno quelli scarsi e di una volta, era usare un cheat che si trovava in una chiavetta esterna. Questa cosa agiava il bypasser in due modi differenti:

- Le chiavette sono generalmente formattate con FAT32, che non prevede alcuno sistema di journaling

- Si poteva estrarre la chiavetta e ripulirla senza fare risultare niente

Per evitare di farsi bypassare in questo modo, bisogna controllare come prima cosa, la presenza in passato di una chiavetta o altro, facilmente consultabile con il programma UsbDeview, di NirSoft. Se si individua una chiavetta recentemente scollegata prima del controllo, sarebbe da ban. Se invece la chiavetta è ancora collegata ed essa è formattata con FAT32, anche li i può bannare, mentre invece nel caso di una chiavetta formattata con NTFS, basta usare il tool fsutil per leggere il journal della chiavetta interessata, specificando la drive letter corretta.

Conclusioni

In questa ultima sezione della guida controlli, abbiamo visto metodi bypass e detect reali, che vengono utilizzati tutt'ora. Tutto questo faceva parte della parte di pratica, e si spera che arrivati a questa conclusione, voi abbiate una conoscenza già approfondita dei capitoli precedenti sulla teoria. Questa parte è inutile senza la teoria e le basi quindi mi raccomando. Per il resto buona lettura e buon proseguimento.

Conclusione?

Conclusione della guida dei controlli – stilata da bestemmie, 22/02/2023 - 22/01/2024

Se siete arrivati a questo punto, intanto, congratulazioni. Congratulazioni intanto per esserci arrivati, perché dedicare il tempo che avete dedicato a leggere questo mini libro sui controlli, solo per migliorare in questa materia, allora davvero complimenti. Come tutte le cose, non le puoi leggere una volta tutte di seguito e sei il maestro finale dei controlli, le cose vanno messe in pratica, ma anche in discussione. Questa guida è lo step iniziale, che a lungo nella community SS mancava. La persona più vicina a questo obbiettivo, è stato forse MrCreeper2010, e la sua serie di video, in cui ha spiegato, seppur in modo diverso, ogni singola cosa riguardo ai controlli dell'epoca, e di conseguenza i suoi video sono stati usati come punto di riferimento nella community per tanto tempo. Purtroppo quei video ora sono stati unlistati o cancellati, nemmeno noi per certo lo sappiamo, ma non è questo il problema. Questa guida non vuole essere un rimpiazzo alle altre guide strane che vagano in giro, o alla serie di video del signor crepeer, bensì vuole essere un po' l'enciclopedia dei controlli, dove tutto quello che voi potete sapere sui controlli cheats di minecraft, si trova qui. Il problema vero è che come con tutti gli argomenti, ci sono infinite sfaccettature alla già presente scena dei controlli, quindi per quanto questa guida cerca di prendere il tutto, ordinarlo e spiegarlo, non saremo mai in grado di unire tutte le conoscenze dal 2013 a oggi. Un altra cosa importante da sottolineare è che gli ss, per quanto possano sembrare oggettivamente morti, sono in continua evoluzione. Questa guida se non altro vi mette in pari, almeno parlando del Q1 del 2023. Questa guida è pensata per novellini e persone interessate ad entrare, ma anche agli OG che sono rimasti alle stringhe su process hacker per la vape del 2019. La dovete prendere probabilmente come una vera e propria enciclopedia più che guida, perché le guide sono dei semplici manuali che spiegano con i paraocchi cosa fare, questa almeno ti da tutte le basi, teoriche e pratiche per poi iniziare il tuo percorso in questo mondo parte dell'ecosistema di minecraft. Se vi siete letti anche questa conclusione finale il triplo più complimenti e non mi resta altro che augurarvi un buon proseguimento.

Non finisce qua ...

Se volete continuare questa parte di minecraft e volete fare parte della community SS, allora vi conviene iniziare a trovare i posti giusti dove mettervi in contatto e conoscere altre persone dedicate a questa sfera di minecraft. Ecco alcuni link utili a voi nell'elenco che segue:

- Canali e gruppi
  - CheatReleaseItaly V3
  - ScreenShareTeam IV
  - SSQuiz&OtherStuff
  - DoggoBypass

- ScreenShare Guide V2

Crediti

Ecco per concludere, la lista di persone che hanno contribuito, più o meno a questo fantastico progetto:

- FreakMyth
- Chein
- SpyCyyh
- xFloppa
- Omqr5912
- Unknown

Ancora una volta, grazie della lettura e detto questo … Buon proseguimento.

- Bestemmie