S11-L4

INDICE

Sommario

ESERCIZIO 1

Tipo di Malware 2

Identificazione del Tipo di Malware 2

Implicazioni di Sicurezza 3

Violazione della Privacy 3

Conclusione 3

Chiamate di Funzione Principali 3

Panoramica 3

Descrizione e Utilizzo 4

Tecniche di Keylogging 4

Rischi Associati e Raccomandazioni 4

Conclusione 5

Contesto e Premessa 5

Meccanismi di Persistenza 5

Implicazioni e Rischi 6

Raccomandazioni per il Contrasto 6

Conclusione 7

Analisi a Basso Livello delle Istruzioni del Malware 7

Introduzione 7

Analisi delle Istruzioni 7

Implicazioni Tecniche 8

Conclusioni e Raccomandazioni 8

ESERCIZIO

Traccia: La figura nella slide successiva mostra un estratto del codice di un malware. Identificate: 1. Il tipo di Malware in base alle chiamate di funzione utilizzate.

2. Evidenziate le chiamate di funzione principali aggiungendo una descrizione per ognuna di essa

3. Il metodo utilizzato dal Malware per ottenere la persistenza sul sistema operativo

4. BONUS: Effettuare anche un’analisi basso livello delle singole istruzioni

Figura 1:

```
.text: 00401010    push eax
.text: 00401014    push ebx
.text: 00401018    push ecx
.text: 0040101C    push WH_Mouse          ; hook to Mouse
.text: 0040101F    call SetWindowsHook()
.text: 00401040    XOR ECX,ECX
.text: 00401044    mov ecx, [EDI]         EDI = «path to
                                          startup_folder_system»
.text: 00401048    mov edx, [ESI]         ESI = path_to_Malware
.text: 0040104C    push ecx               ; destination folder
.text: 0040104F    push edx               ; file to be copied
.text: 00401054    call CopyFile();
```

Tipo di Malware

Introduzione

Questo report fornisce un'analisi approfondita di un frammento di codice sospetto estratto durante un'indagine forense informatica. Il codice in esame sembra essere parte di un software maligno (malware), con caratteristiche indicative di un keylogger o spyware.

Identificazione del Tipo di Malware

Keylogger e spyware sono tipologie di malware progettati per operare in modo furtivo, raccogliendo informazioni senza il consenso o la consapevolezza dell'utente. Queste informazioni possono includere pressioni dei tasti, movimenti e clic del mouse, dati di autenticazione, e altro.

Tecniche di Interfacciamento

Il frammento di codice analizzato mostra l'uso della chiamata di funzione SetWindowsHook(), che permette di inserire un hook. Un hook è un punto di intercettazione che può essere usato per monitorare o modificare il comportamento del sistema o di altre applicazioni.

Funzione WH_Mouse

L'uso di WH_Mouse indica che il malware sta cercando di intercettare gli eventi del mouse. Questo tipo di monitoraggio è spesso utilizzato per raccogliere dati sui comportamenti dell'utente, che possono includere:

- Movimenti del mouse che possono rivelare pattern di utilizzo o informazioni sensibili.

- Clic del mouse che possono indicare interazioni con interfacce specifiche, come moduli di login o pagamenti.

Metodo di Hooking

La chiamata a SetWindowsHook() è particolarmente significativa perché dimostra che il malware ha la capacità di inserirsi nei processi di basso livello del sistema operativo. Questo metodo di hooking può essere utilizzato per eseguire codice arbitrario in risposta a eventi di input, fornendo una potente tecnica per raccogliere dati in maniera segreta.

Implicazioni di Sicurezza

Rischio di Furto di Dati

Un malware che utilizza queste tecniche può catturare una vasta gamma di informazioni personali e sensibili. Il rischio di furto di dati è elevato, dato che le informazioni raccolte possono essere trasmesse a un attaccante.

Violazione della Privacy

La privacy degli utenti è gravemente compromessa da software che operano come keylogger o spyware. Gli utenti potrebbero non essere consapevoli che le loro attività sono sotto sorveglianza, portando a una violazione della fiducia e della sicurezza personale.

Conclusione

In base all'analisi delle funzioni utilizzate nel frammento di codice fornito, è ragionevole classificare il malware in esame come un potenziale keylogger o spyware. Questi tipi di malware rappresentano una minaccia significativa alla sicurezza e alla privacy degli utenti e richiedono misure immediate di mitigazione e rimozione da parte dei team di sicurezza informatica.

Si consiglia di procedere con ulteriori analisi comportamentali e statiche per confermare queste prime osservazioni e per identificare l'intera portata delle funzionalità del malware e le sue potenziali comunicazioni con server di comando e controllo. Questo consentirà di sviluppare strategie di difesa più efficaci e di attuare risposte incidentali appropriate.

Chiamate di Funzione Principali

Panoramica

L'analisi del frammento di codice sospetto ha rivelato l'uso di chiamate di funzione che sono di vitale importanza per l'operatività del malware in questione. In particolare, la chiamata a SetWindowsHook() gioca un ruolo cruciale nelle capacità di monitoraggio e

di registrazione del malware. Questa sezione fornisce un'analisi dettagliata di questa funzione, spiegando il suo ruolo e il potenziale uso nel contesto del malware.

SetWindowsHook()

Descrizione e Utilizzo

La funzione SetWindowsHook() è una chiamata di sistema fornita dalle API di Windows che permette di modificare il comportamento del sistema o di un'applicazione attraverso l'uso di hook. Un hook è essenzialmente un meccanismo di callback che viene attivato in risposta a eventi specifici, come input da tastiera o mouse.

In un contesto legittimo, SetWindowsHook() può essere utilizzata per personalizzare o estendere le funzionalità del sistema operativo. Tuttavia, nel codice del malware, questa funzione assume un ruolo più sinistro:

Potenziale di Sorveglianza

L'installazione di un hook attraverso SetWindowsHook() consente al malware di intercettare tutti gli eventi del mouse a livello di sistema, incluso al di fuori del contesto dell'applicazione maligna. Questo significa che il malware può registrare clic e movimenti del mouse su qualsiasi elemento dell'interfaccia utente, compresi i campi di input di password o altre aree sensibili.

Tecniche di Keylogging

Sebbene nel frammento specifico si faccia riferimento solo agli eventi del mouse (WH_Mouse), tecniche simili possono essere impiegate per catturare anche le pressioni dei tasti. Questo fa presumere che il malware possa avere la capacità di agire anche come keylogger, una seria minaccia per la sicurezza delle credenziali dell'utente.

Implicazioni della Sorveglianza a Basso Livello

SetWindowsHook() permette al malware di eseguire codice personalizzato in risposta a eventi di sistema, che può essere utilizzato per esfiltrare dati in tempo reale o per attivare processi maligni in specifici contesti, aumentando la difficoltà di rilevamento e rimozione.

Rischi Associati e Raccomandazioni

Rischio di Data Breach

Il monitoraggio degli eventi del mouse e potenzialmente delle pressioni dei tasti rappresenta un rischio elevato di data breach. Le informazioni raccolte possono essere facilmente esfiltrate e utilizzate per scopi fraudolenti.

Interventi di Sicurezza

Si raccomanda di effettuare scansioni approfondite del sistema con software antivirus aggiornato, di implementare soluzioni di Endpoint Detection and Response (EDR), e di condurre un'analisi forense per determinare la portata della compromissione.

Formazione e Consapevolezza

È essenziale educare gli utenti sulla tipologia di minacce rappresentate da tali malware e sulle buone pratiche di sicurezza informatica, come l'uso di autenticazione a più fattori, che può mitigare il rischio anche in presenza di keylogger.

Conclusione

La funzione SetWindowsHook() è un elemento chiave che rivela la natura invasiva e pericolosa del malware analizzato. La sua capacità di intercettare e registrare input dell'utente costituisce una seria minaccia alla privacy e alla sicurezza delle informazioni. Pertanto, è imperativo agire tempestivamente per identificare, contenere ed eliminare questa minaccia dal sistema compromesso.

Contesto e Premessa

L'analisi del frammento di codice sospetto evidenzia l'intenzione del malware di stabilire una presenza duratura sul sistema infetto. Questa sezione esamina le strategie adottate dal malware per ottenere e mantenere la persistenza, minando l'integrità e la sicurezza del sistema operativo colpito.

Meccanismi di Persistenza

Manipolazione dei Punti di Avvio Automatico

Il codice analizzato suggerisce che il malware tenta di garantirsi l'esecuzione automatica all'avvio del sistema. Questo è indicativo dell'uso di tecniche di persistenza che manipolano le funzioni di avvio automatico del sistema operativo Windows.

Tecniche Specifiche

La presenza di istruzioni che indicano il movimento di percorsi di file verso il startup_folder_system è particolarmente allarmante. Questa cartella è una directory monitorata dal sistema operativo che esegue automaticamente gli script e i programmi in essa contenuti ad ogni avvio del computer.

CopyFile()

L'uso della funzione CopyFile() suggerisce che il malware copia se stesso o alcuni suoi componenti in questa directory, assicurando che venga eseguito regolarmente. Questa

è una tecnica comune per assicurare che il malware venga riattivato dopo riavvii del sistema, logout o shutdown.

Implicazioni e Rischi

Resistenza alla Rimozione

Una volta che il malware ha stabilito la persistenza, diventa molto più difficile da rimuovere. Può eludere tentativi di disinfezione superficiali e continuare a operare anche dopo che l'utente crede di averlo eliminato.

Potenziale per Danneggiamenti Continui

La persistenza permette al malware di eseguire processi dannosi ogni volta che il sistema viene avviato, potenzialmente conducendo a danni continui o all'esfiltrazione di dati su un lungo periodo.

Raccomandazioni per il Contrasto

Analisi Forense Avanzata

È essenziale eseguire un'analisi forense completa per identificare tutte le tracce e le istanze del malware. Questo include la revisione delle chiavi di registro, dei task pianificati, e dei servizi di sistema che potrebbero essere stati manipolati per eseguire il malware.

Misure di Contenimento

Dovrebbero essere intraprese azioni per isolare e contenere il sistema colpito per prevenire la diffusione del malware e ulteriori danni. Questo può includere disconnessioni dalla rete, arresti temporanei del sistema e revoca dell'accesso alle risorse condivise.

Strumenti di Rimozione Specializzati

Utilizzare strumenti di rimozione di malware affidabili e aggiornati che sono specificamente progettati per affrontare le tecniche di persistenza. Tali strumenti possono rilevare e ripulire le modifiche al sistema che potrebbero non essere evidenti durante gli esami manuali.

Educazione degli Utenti

Gli utenti dovrebbero essere istruiti su come evitare comportamenti che potrebbero portare a reinfezioni, come l'apertura di allegati di email sospetti o il download di software da fonti non verificate.

Conclusione

La capacità di un malware di ottenere persistenza su un sistema operativo è un indicatore della sua sofisticatezza e della minaccia che rappresenta. Il malware in questione utilizza tecniche insidiose per assicurarsi che le sue operazioni dannose possano continuare indisturbate, rendendo critica una risposta immediata e comprensiva da parte dei professionisti della sicurezza informatica. La combinazione di strumenti automatizzati e competenze forensi è fondamentale per eliminare la minaccia e ripristinare la sicurezza del sistema.

Analisi a Basso Livello delle Istruzioni del Malware

Introduzione

Nel contesto della sicurezza informatica, la comprensione delle operazioni a basso livello eseguite da un malware è essenziale per sviluppare una risposta efficace. Questo report si concentra sull'analisi delle singole istruzioni assembly presenti nel frammento di codice del malware in questione.

Analisi delle Istruzioni

Registro EAX, EBX, ECX, EDI, ESI

I registri EAX, EBX, ECX, EDI e ESI sono utilizzati per vari scopi all'interno del codice assembly. Questi possono contenere indirizzi, valori temporanei, o possono essere usati in operazioni aritmetiche e logiche.

push eax/ebx/ecx: Queste istruzioni salvano i valori correnti dei registri generali sulla stack. Questo è spesso fatto in preparazione per una chiamata di funzione, per preservare lo stato dei registri prima dell'esecuzione di subroutine che possono modificarli.

Operazioni di Zeroing

XOR ECX,ECX: Un'operazione XOR del registro ECX con se stesso. Questo è un metodo efficiente per azzerare un registro, poiché l'operazione XOR di un valore con se stesso risulta sempre in zero. Azzerrare un registro è una pratica comune prima di utilizzarlo in operazioni successive.

Operazioni di Movimento

mov ecx, [EDI]: Questa istruzione copia il contenuto puntato dal registro EDI nel registro ECX. EDI è comunemente usato per contenere indirizzi, quindi questa operazione potrebbe essere usata per preparare un parametro per una chiamata di funzione o per un'operazione di trasferimento di dati.

mov edx, [ESI]: Analogamente, copia il contenuto puntato da ESI in EDX. ESI è spesso usato come puntatore di origine in operazioni di copia di stringhe o array.

Preparazione per Chiamate di Funzione

push ecx: Mette il contenuto di ECX sulla stack. Nella convenzione di chiamata stdcall, questo potrebbe essere un parametro per una chiamata di funzione imminente.

push edx: Mette il contenuto di EDX sulla stack. Similmente a ECX, questo potrebbe essere un parametro per una chiamata di funzione.

Chiamata di Funzione di Copia

call CopyFile(): Questa istruzione invoca la funzione CopyFile(), che è una funzione dell'API di Windows usata per copiare file. I parametri per questa funzione sono probabilmente preparati dalle operazioni di push precedenti, con ECX e EDX che forniscono i percorsi di origine e destinazione del file.

Implicazioni Tecniche

Ogni istruzione assembly ha uno scopo preciso e, nel contesto di un malware, potrebbe essere utilizzata per eseguire operazioni dannose. L'analisi a basso livello aiuta a comprendere come il malware manipola il sistema e quali risorse utilizza.

Conclusioni e Raccomandazioni

L'analisi a basso livello fornisce intuizioni preziose sulle strategie adottate dal malware per eseguire le sue operazioni dannose. Comprendere il flusso di istruzioni è fondamentale per sviluppare misure di mitigazione, strumenti di rimozione e per il reverse engineering del malware. Gli esperti di sicurezza dovrebbero usare questa analisi per rinforzare le difese e per educare su misure preventive più robuste.

In definitiva, l'analisi dettagliata a livello di istruzione è un componente chiave nella lotta contro il malware e nella protezione delle infrastrutture informatiche. Le istruzioni assembly possono rivelare non solo le intenzioni degli attaccanti, ma anche fornire indizi su come il malware può essere identificato, tracciato e neutralizzato.