




Spis treści

Przedmowa do wydania trzeciego	11
„Szkoła Hakerów??? Czy wyście już do końca powariowali?!	11
Kim jest haker?	12
Misja Szkoły Hakerów	12
Słowem wstępu	15
Sposób przekazywania wiedzy, który działa	15
Dla kogo jest ta książka?	16
Czego nauczysz się z tej książki?	17
Jak korzystać z podręcznika Szkoły Hakerów?	17
Szkoliowy System Operacyjny i Filmy Instruktażowe	18
O Autorach	18
Podsumowanie	19
Wskazówki prawne	21
Art. 267.	21
Art. 268.	22
Art. 268a.	22
Art. 269.	23
Art. 269a.	23
Art. 269b.	23



Rozdział 1. Odzyskiwanie zagubionych haseł	25
Odzyskiwanie haseł RAR i ZIP – Advanced Archive Password Recovery	27
Odzyskiwanie plików DOC – Advanced Office Password Recovery	31
Rozdział 2. Przechwytywanie informacji w sieciach LAN	33
Co będzie nam potrzebne?	35
Krótkie przedstawienie używanych programów	35
Czym jest protokół ARP?	37
Jak to działa?	38
Skąd wziąć cel?	41
Rozdział 3. Przechwytywanie danych szyfrowanych, atak na sesję SSL	45
Rozdział 4. Tylne drzwi Twoją furtką do systemu	51
Modyfikacja /etc/passwd	52
Dodanie nowej usługi	52
ICMP Backdoor	56
Modyfikacja daemona	58
Rozdział 5. Ukrywanie plików i katalogów przy pomocy modułu jądra serii 2.6	59
Budowa modułu	61
Kompilacja modułu jądra	62
Obsługa modułów przez jądro	63
Wywołania systemowe	65
O rejestrach nieco więcej	67
Zestawienie rejestrów	67
Funkcje obsługi i tablica sys_call_table	68
Dostęp do sys_call_table w nowych jądrach serii 2.6	69
Podmiana funkcji w sys_call_table	69
Ukrywanie plików i katalogów przy użyciu modułu jądra	71
Rozwijamy nasz moduł	75
Ukrywanie modułu jądra	77
Rozdział 6. Ataki typu Buffer Overflow	81
Pamięć	81
Stos	82
Czym jest bufor?	84
Prosty przykład wykorzystania buffer overflow	87
Zaawansowany przykład buffer overflow	89
Użycie shellcode	93
Jak nie popełnić błędu?	101

Rozdział 7. Praktyczny przykład ataku zdalnego	103
Zbieranie informacji	104
Badanie strony www	105
Wybór języka programowania	107
Język Python	107
Moduły Pythona	110
Piszemy exploita	112
Praktyczne wykorzystanie exploita	116
Rozdział 8. Ataki typu Heap Overflow	121
Segmenty pamięci	121
Serta	125
Przepełnienie bufora	126
Przykład heap overflow	127
Przykład bss overflow	130
Rozdział 9. Ataki typu Format String	135
Czym jest ciąg formatujący?	135
Błędne użycie funkcji printf()	137
Zastosowanie znacznika %n	139
Praktyczne wykorzystanie błędu typu format string	145
Wykorzystanie shellcode	147
Nadpisanie kopii EIP	148
Nadpisywanie sekcji GOT	151
Nadpisywanie sekcji DTORS	153
Jak uniknąć błędu?	155
Rozdział 10. Praktyczny przykład ataku typu Format String	157
Wybór atakowanego oprogramowania	157
Uzyskanie dostępu do przekazanego adresu shellcode	160
Określenie najlepszego miejsca dla shellcode w pamięci	163
Znalezienie miejsca nadającego się do nadpisanie	164
Nadpisanie określonego miejsca adresem shellcode	166
Problemy związane z długością zapytania	171
Rozdział 11. Nadpisywanie wskaźnika strumienia pliku, File Stream Pointer Overwrite	173
Omijanie wskaźnika strumienia pliku	173
Wykorzystanie wskaźnika strumienia pliku	177
Atak na FreeBSD	178
Atak na system Linux	183
Rozdział 12. Błędy na poziomie jądra systemu	193
Błędy jądra	193
Buffer overflow – krótkie przypomnienie	194

Podatny moduł jądra	196
Tworzenie shellcode	202
Exploit	205
Przykład z życia – bluetooth	207
Tworzenie exploita	209
Brak adresu tablicy bt_proto	212
Rozdział 13. Wykorzystanie protokołu ICMP	
z punktu widzenia Hakera	217
Budowa pakietu IP	218
Narzędzie ping	221
Określanie drogi pakietu przy użyciu programu traceroute	223
Wykorzystanie ICMP w atakach typu DoS	224
Ping flooding	224
Tworzymy własny ping flooder	225
Backdoor wykorzystujący ICMP	230
Przesyłanie danych za pomocą ICMP	234
Skanowanie ICMP	242
Rozdział 14. Zdalne rozpoznawanie systemu operacyjnego	243
Era kamienia łupanego	243
OS Fingerprinting	245
Aktywny Fingerprinting	245
Jak to działa	247
Pasywny Fingerprinting	253
Wykorzystanie p0fa przeciw nmapowi	256
Rozdział 15. Netfilter w służbie bezpieczeństwa systemu	259
Fingerprinting – przypomnienie	259
Stack fingerprinting	260
Moduły kernela	261
Netfilter	263
Filtracja pakietów	266
Filtracja portu	268
Modyfikacja pakietów w locie	271
Podszywamy się pod FreeBSD	275
Rozdział 16. Zabezpieczanie systemu krok po kroku	281
Przygotowywanie dysku twardego	281
Wybór instalacji	283
Hasło administratora	283
Firewall	283
Superserwer xinetd	285
Konfiguracja SSH	286
Ukrywanie informacji oraz bit SUID	287
Oprogramowanie poprawiające bezpieczeństwo systemu	290

Kernel – czuły punkt systemu	291
Co dalej?	291
Rozdział 17. Skanery bezpieczeństwa	293
Czym są skanery	294
Nmap	294
Techniki skanowania	295
Skanowanie z wykorzystaniem connect()	295
Skanowanie TCP SYN	296
Skanowanie TCP FIN	296
Skanowanie TCP Ident	296
Skanowanie UDP przy pomocy ICMP	296
Skanowanie UDP przy pomocy write() i recvfrom()	297
Skanowanie ICMP Echo	297
Skanowanie fragmentacyjne	298
Skanowanie z wykryciem wersji	298
Skanowanie protokołem IP	298
Skanowanie ACK	298
Skanowanie rozmiarem okna TCP	299
Skanowanie RPC	299
Skanowanie listy	299
Skanowanie przy pomocy pinga	299
Skanowanie bezczynne	299
Praktyka	300
Wprowadzenie do Nessusa	302
Konfiguracja	302
Praktyka	303
Tworzenie zasad dla użytkowników	304
Użytkowanie	304
Nikto	307
Instalacja	307
Użytkowanie i opcje	307
Konfiguracja Nikto	309
Praktyka	311
Rozdział 18. Łaty poprawiające poziom bezpieczeństwa	315
Grsecurity	317
Łatamy i konfigurujemy nowe jądro	318
Kompilacja jądra z łatą grsecurity	323
Stack-Smashing Protector	326
LibSafe	336
Rozdział 19. Systemy wykrywania włamań	341
Czym jest IDS?	341

Snort	342
Instalacja Snorta	343
Konfiguracja Snorta	343
Użytkowanie	344
Zasady Snorta	347
Portsentry	354
Instalujemy Portsentry	354
Użytkowanie Portsentry	355
Rozdział 20. Atak z wykorzystaniem serwera stron WWW	357
Cel	357
Serwer Apache i jego możliwości	358
PHP	359
Niebezpieczne funkcje uruchamiające	359
Listowanie zawartości katalogu i odczyt plików	360
Moduły PHP	362
Kompilacja modułów i problemy z tym związane	369
Wszystko zablokowane, co teraz?	370
Nie da się – czy to już koniec?	371
CGI	371
Moduły mod_python, mod_perl, mod_*	375
Rozdział 21. Spim, czyli zmora dla komunikatora	377
Czym jest spim?	377
Narzędzia	378
Biblioteka libgadu	379
Wysyłanie wiadomości za pomocą libgadu	379
Obsługa katalogu publicznego z poziomu libgadu	384
SpimBomber	389
Jak się obronić	396
Rozdział 22. Tworzenie shellcodu w środowisku Win32	397
Czym jest shellcode	397
Typy shellcodów	398
Odnalezienie adresu kernela	398
Wykorzystanie adresów stałych	398
Odnajdywanie adresów funkcji API za pomocą Export Section kernela	408
Czym są funkcje API	408
Do czego shellcode potrzebuje funkcje API	408
Czym jest sekcja export	409
Odnajdywanie adresów funkcji API za pomocą Import Address Table	414
Shellcode, który ściąga i uruchamia trojana z wykorzystaniem Win32-IF	419
Czym są Win32-Internet Functions?	419
Zakończenie	429