

Spis treści

O autorach	9
O korektorze merytorycznym	11
Podziękowania	13
Wstęp	17
Rozdział 1. Architektura x86 i x64	21
Rejestry i typy danych	22
Zbiór instrukcji	23
Składnia	24
Przenoszenie danych	25
Ćwiczenie	30
Operacje arytmetyczne	31
Operacje stosu i wywołanie funkcji	32
Ćwiczenia	36
Sterowanie wykonywanym programem	37
Mechanizm systemowy	44
Translacja adresów	45
Przerwania i wyjątki	47
Analiza krok po kroku	47
Ćwiczenia	54
x64	55
Rejestry i typy danych	55
Przenoszenie danych	56
Adresowanie kanoniczne	56
Wywołanie funkcji	57
Ćwiczenia	57

Rozdział 2.	Architektura ARM	59
	Podstawowe funkcje	60
	Typy danych i rejestry	63
	Opcje systemu i ustawienia	65
	Instrukcje — wprowadzenie	66
	Ładowanie i zapisywanie danych	67
	Instrukcje LDR i STR	67
	Inne zastosowania instrukcji LDR	71
	Instrukcje LDM i STM	72
	Instrukcje PUSH i POP	76
	Funkcje i wywoływanie funkcji	77
	Operacje arytmetyczne	80
	Rozgałęzianie i wykonywanie warunkowe	81
	Tryb Thumb	85
	Polecenia switch-case	85
	Rozmaitości	87
	Kompilacja just-in-time i samomodyfikujący się kod	87
	Podstawy synchronizacji	88
	Mechanizmy i usługi systemowe	89
	Instrukcje	91
	Analiza krok po kroku	91
	Co dalej?	98
	Ćwiczenia	98
Rozdział 3.	Jądro systemu Windows	107
	Podstawy systemu Windows	108
	Rozkład pamięci	108
	Inicjalizacja procesora	109
	Wywołania systemowe	111
	Poziom żądań przerwania urządzenia	123
	Pule pamięci	125
	Listy deskryptorów pamięci	126
	Procesy i wątki	126
	Kontekst wykonywania	128
	Podstawy synchronizacji jądra	129
	Listy	130
	Szczegóły implementacji	131
	Analiza krok po kroku	138
	Ćwiczenia	142

Wykonywanie asynchroniczne i ad hoc	146
Wątki systemowe	147
Elementy robocze	148
Asynchroniczne wywoływanie procedur	150
Opóźnione wywoływanie procedur	154
Timery	158
Wywołania zwrotne procesów i wątków	159
Procedury zakończenia	161
Pakiety żądań wejścia-wyjścia	162
Struktura sterownika	164
Punkty rozpoczęcia	165
Obiekty sterownika i urządzenia	166
Obsługa pakietów IRP	167
Popularne mechanizmy zapewniające komunikację pomiędzy kodem użytkownika a kodem jądra	168
Inne mechanizmy systemowe	170
Analiza krok po kroku	173
Rootkit w architekturze x86	173
Rootkit w architekturze x64	188
Dalszy rozwój	195
Ćwiczenia	196
Rozwijanie pewności siebie i utrwalanie wiadomości	197
Poszerzanie horyzontów	198
Analiza prawdziwych sterowników	201
Rozdział 4. Debugowanie i automatyzacja	203
Narzędzia i podstawowe polecenia służące do debugowania	204
Określanie ścieżki plików symboli	205
Okna debugera	205
Obliczanie wartości wyrażenia	206
Zarządzanie procesami i debugowanie zdarzeń	210
Rejestry, pamięć i symbole	214
Punkty wstrzymania	223
Kontrolowanie procesów i modułów	226
Inne polecenia	229
Skrypty i debugowanie	230
Pseudorejestry	231
Alias	233
Język	240
Pliki skryptów	251
Skrypty jako funkcje	255
Przykładowe skrypty przydatne podczas debugowania	260

	Korzystanie z narzędzi SDK	267
	Pojęcia	268
	Tworzenie rozszerzeń narzędzi debugujących	272
	Praktyczne rozszerzenia, narzędzia i źródła wiedzy	274
Rozdział 5.	Zaciemnianie kodu	277
	Techniki zaciemniania kodu	279
	Po co zaciemniać kod?	279
	Zaciemnianie oparte na danych	282
	Zaciemnianie oparte na sterowaniu	287
	Jednoczesne zaciemnianie przepływu sterowania i przepływu danych	293
	Zabezpieczanie przez zaciemnianie	297
	Techniki rozjaśniania kodu	297
	Natura rozjaśniania kodu: odwracanie przekształceń	298
	Narzędzia przeznaczone do rozjaśniania kodu	303
	Rozjaśnianie kodu w praktyce	319
	Studium przypadku	335
	Pierwsze wrażenie	335
	Analiza semantyki procedury	337
	Obliczanie symboliczne	339
	Rozwiązanie zadania	340
	Podsumowanie	342
	Ćwiczenia	343
	Bibliografia	343
Dodatek A	Sumy kontrolne SHA1	347
	Skorowidz	349