

Spis treści |

O autorze	13
O recenzencie	14
Przedmowa.....	15
Wstęp	17
ROZDZIAŁ 1	
Prezentacja zestawu protokołów TCP/IP i Pythona	23
Przedstawienie internetu	25
Serwery, hosty i komponenty sieci	26
Powstanie centrów danych	27
Model OSI	31
Model klient-serwer	34
Zestawy protokołów sieciowych	34
Protokół sterowania transmisją — TCP	35
Protokół pakietów użytkownika — UDP	36
Protokół internetowy — IP	37
Przegląd języka Python	39
System operacyjny	41
Uruchamianie programów w Pythonie	42
Wbudowane typy danych Pythona	43
Operatory w Pythonie	49
Narzędzia do sterowania przepływem	50
Funkcje w Pythonie	52
Klasy w Pythonie	52
Moduły i pakiety	53
Podsumowanie	55

ROZDZIAŁ 2

Niskopoziomowe interakcje z urządzeniami sieciowymi	56
Wyzwania stosowania CLI	57
Konstruowanie wirtualnego laboratorium	59
Urządzenia fizyczne	59
Urządzenia wirtualne	59
Laboratoria modelowania Cisco	61
Cisco DevNet	64
GNS3 i inne laboratoria	64
Wirtualne środowisko Pythona	66
Biblioteka Pexpect	67
Instalacja modułu Pexpect	67
Przegląd biblioteki Pexpect	68
Pierwszy program używający Pexpect	73
Więcej możliwości Pexpect	74
Pexpect i SSH	75
Kompletny przykład użycia biblioteki Pexpect	76
Biblioteka Paramiko	77
Instalacja biblioteki Paramiko	77
Prezentacja biblioteki Paramiko	77
Pierwszy program z użyciem Paramiko	80
Więcej możliwości biblioteki Paramiko	81
Więcej przykładów użycia biblioteki Paramiko	83
Biblioteka Netmiko	84
Framework Nornir	86
Wady Pexpect i Paramiko w porównaniu z innymi narzędziami	88
Podsumowanie	89

ROZDZIAŁ 3

API i sieci intuicyjne	90
Infrastruktura jako kod (IaC)	91
Sieci intuicyjne	92
Analiza wyświetlanych wyników a strukturalne wyniki API	93
Modelowanie danych na potrzeby IaC	96
YANG i NETCONF	98
Przykłady wykorzystania API urządzeń Cisco	98
Cisco NX-API	99
Model YANG firmy Cisco	104
Przykłady Cisco ACI	105
Kontroler Cisco Meraki	109
API Pythona dla urządzeń Juniper Networks	110
Juniper i NETCONF	111
Biblioteka Juniper PyEZ dla programistów	115

API Pythona dla urządzeń Arista	119
Zarządzanie przy użyciu Arista eAPI	120
Biblioteka Pyeapi firmy Arista	124
Przykład VyOS	128
Inne biblioteki	129
Podsumowanie	130

ROZDZIAŁ 4

Framework automatyzacyjny Pythona — Ansible	131
Ansible — framework bardziej deklaracyjny	132
Wersje Ansible	134
Pierwszy sieciowy przykład użycia Ansible	135
Instalacja Ansible na węźle kontrolnym	136
Topologia laboratorium	137
Zalety Ansible	141
Brak agentów	141
Idempotentność	142
Prostota i rozszerzalność	143
Ansible Content Collection	143
Kolejne sieciowe przykłady użycia Ansible	144
Zagnieżdżanie ewidencji	145
Wyrażenia warunkowe w Ansible	147
Zmiany konfiguracji	149
Fakty sieciowe Ansible	150
Pętle Ansible	152
Szablony	156
Podsumowanie	162

ROZDZIAŁ 5

Kontenery Dockera dla inżynierów sieciowych	163
Prezentacja Dockera	164
Zalety Dockera	165
Tworzenie aplikacji Pythona w Dockerze	166
Instalowanie Dockera	166
Przydatne polecenia Dockera	167
Budowanie aplikacji hello-world	168
Budowanie własnej aplikacji	169
Udostępnianie obrazów Dockera	172
Orkiestracja kontenerów przy użyciu docker-compose	175
Sieci w kontenerach	177
Sieć hosta kontenerów	178
Własne sieci mostkowe	179
Inne opcje sieciowe kontenerów	180

Kontenery w zastosowaniach inżynierii sieciowej	181
Containerlab	181
Docker i Kubernetes	185
Podsumowanie	186

ROZDZIAŁ 6

Bezpieczeństwo sieci w Pythonie	187
Przygotowanie laboratorium	188
Program Scapy	194
Instalowanie Scapy	194
Interaktywne przykłady korzystania ze Scapy	196
Przechwytywanie pakietów z użyciem Scapy	198
Skanowanie portów TCP	199
Zestaw narzędzi ping	203
Popularne ataki	204
Zasoby dotyczące Scapy	205
Listy dostępu	205
Implementacja list dostępu przy użyciu Ansible	206
Listy dostępu adresów MAC	210
Przeszukiwanie dzienników Syslog	211
Przeszukiwanie z użyciem modułu obsługi wyrażeń regularnych	212
Inne narzędzia	214
Prywatne wirtualne sieci lokalne	214
Obsługa UFW w Pythonie	215
Dalsza lektura	216
Podsumowanie	216

ROZDZIAŁ 7

Monitorowanie sieci przy użyciu Pythona — część 1.	218
Konfiguracja laboratorium	219
SNMP	221
Konfiguracja	222
PySNMP	225
Stosowanie Pythona do wizualizacji danych	230
Matplotlib	230
Pygal	238
Stosowanie Cacti	242
Instalacja	243
Skrypt Pythona jako źródło danych wejściowych	245
Podsumowanie	247

ROZDZIAŁ 8

Monitorowanie sieci przy użyciu Pythona — część 2.	249
Graphviz	250
Konfiguracja laboratorium	251
Instalacja	252
Przykłady Graphviz	253
Przykłady użycia Graphviza w Pythonie	255
Użycie LLDP do prezentowania sąsiadujących węzłów sieci	256
Monitorowanie oparte na przepływach	265
Parsowanie danych NetFlow z użyciem Pythona	266
Monitorowanie ruchu sieciowego przy użyciu ntop	270
Rozszerzanie ntop przy użyciu Pythona	273
sFlow	277
Podsumowanie	282

ROZDZIAŁ 9

Tworzenie sieciowych usług webowych przy użyciu Pythona	283
Porównanie frameworków webowych Pythona	285
Konfiguracja Flaska i laboratorium sieciowego	288
Wprowadzenie do Flaska	289
Wersje Flaska	289
Przykłady stosowania Flaska	290
Klient HTTPie	291
Obsługa adresów URL	293
Zmienne URL	294
Generowanie adresów URL	295
Zwracanie wyników w formacie JSON	296
API zasobu sieciowego	297
Flask-SQLAlchemy	297
API zawartości sieci	299
API urządzeń	302
API konkretnego urządzenia	304
Dynamiczne operacje sieciowe	305
Operacje asynchroniczne	308
Uwierzelnianie i autoryzacja	310
Uruchamianie Flaska w kontenerach	313
Podsumowanie	317

ROZDZIAŁ 10

Wprowadzenie do asynchronicznych operacji wejścia-wyjścia	318
Przegląd operacji asynchronicznych	319
Wieloprocusowość w Pythonie	320
Wielowątkowość w Pythonie	321

Moduł asyncio Pythona	323
Projekt Scrapli	327
Przykład zastosowania Scrapli	327
Asynchroniczny przykład korzystający ze Scrapli	329
Podsumowanie	332

ROZDZIAŁ 11

Sieci w chmurze AWS	334
Konfiguracja AWS	335
AWS CLI i Python SDK	337
Przegląd zagadnień sieciowych w chmurze AWS	340
Wirtualna chmura prywatna	346
Tabele tras i cele tras	351
Automatyzacja przy użyciu CloudFormation	353
Grupy bezpieczeństwa i sieciowe listy ACL	356
Elastic IP	358
Bramy NAT	360
Direct Connect i VPN	361
Bramy VPN	361
Direct Connect	362
Usługi skalowania sieci	364
Elastic Load Balancing	364
Usługa DNS Route 53	365
Usługi CDN CloudFront	365
Inne sieciowe usługi AWS	366
Podsumowanie	366

ROZDZIAŁ 12

Sieci w chmurze Azure	368
Porównanie usług sieciowych Azure i AWS	369
Konfiguracja Azure	371
Administracja i API Azure	372
Jednostki usług Azure	376
Python a PowerShell	379
Globalna infrastruktura Azure	380
Sieci wirtualne na platformie Azure	381
Dostęp do internetu	384
Tworzenie zasobów sieciowych	387
Punkty końcowe usługi VNet	389
Peering sieci VNet	390
Routing w sieciach wirtualnych	392
Grupy bezpieczeństwa sieci	397
Wirtualne sieci prywatne na platformie Azure	399

ExpressRoute	402
Równoważenie obciążenia w Azure	404
Inne usługi sieciowe Azure	405
Podsumowanie	406

ROZDZIAŁ 13

Analiza danych sieciowych z użyciem Elastic Stack	407
Czym jest Elastic Stack?	408
Topologia laboratorium	409
Elastic Stack jako usługa	415
Pierwszy kompleksowy przykład	417
Obsługa Elasticsearch przy użyciu klienta napisanego w Pythonie	422
Pozyskiwanie danych za pomocą Logstash	424
Pozyskiwanie danych za pomocą Beats	426
Wyszukiwanie przy użyciu Elasticsearch	432
Wizualizacja danych za pomocą Kibany	436
Podsumowanie	442

ROZDZIAŁ 14

Korzystanie z systemu Git	443
Rozważania dotyczące zarządzania treścią i system Git	444
Wprowadzenie do systemu Git	445
Zalety Gita	446
Terminologia Gita	447
Git i GitHub	448
Konfiguracja Gita	449
Plik gitignore	450
Przykłady stosowania systemu Git	451
Gałęzie	455
Przykład GitHub	457
Stosowanie Gita z Pythonem	465
Pakiet GitPython	465
Biblioteka PyGithub	466
Automatyzacja tworzenia kopii zapasowych konfiguracji	467
Współdziałanie przy użyciu Gita	470
Podsumowanie	471

ROZDZIAŁ 15

Ciągła integracja z użyciem GitLab	472
Tradycyjny proces zarządzania zmianą	473
Wprowadzenie do ciągłej integracji	474
Instalowanie GitLab	476

Runnery GitLaba	480
Pierwszy przykład stosowania GitLaba	481
Sieciowy przykład użycia GitLaba	489
Podsumowanie	491

ROZDZIAŁ 16

Programowanie w oparciu o testy

na potrzeby programowania sieciowego	493
Prezentacja programowania w oparciu o testy	494
Definicje testów	495
Topologia jako kod	496
Przykład parsowania kodu XML	499
Moduł unittest Pythona	501
Więcej o testowaniu w Pythonie	504
Przykłady stosowania modułu pytest	505
Pisanie testów na potrzeby sieci	507
Testowanie dostępności	508
Testowanie opóźnień sieci	509
Testowanie bezpieczeństwa	510
Testowanie transakcji	510
Testowanie konfiguracji sieciowych	511
Testowanie Ansible	511
pyATS i Genie	512
Podsumowanie	518

Skorowidz	520
------------------------	------------