

Spis treści

O autorach	13
<u>O recenzencie technicznym</u>	<u>14</u>
<u>Wprowadzenie</u>	<u>15</u>
Część I. Wprowadzenie do Dockera i kontenerów	21
<u>Rozdział 1. Podstawy Dockera i kontenerów</u>	<u>23</u>
Wymagania techniczne	24
Zrozumienie potrzeby stosowania kontenerów	24
Pojawienie się Dockera	25
Poznajemy Dockera	26
Kontenery są tymczasowe	26
Obrazy Dockera	27
Warstwy obrazu	28
Trwałe przechowywanie danych	29
Uzyskiwanie dostępu do usług działających w kontenerach	29
Instalacja Dockera	30
Przygotowania do instalacji Dockera	31
Instalowanie Dockera w Ubuntu	31
Nadanie Dockerowi niezbędnych uprawnień	32
Używanie Dockera w powłoce	34
docker help	34
docker run	34
docker ps	35
docker start i docker stop	36

Spis treści	
docker attach	37
docker exec	38
docker logs	39
docker rm	40
Podsumowanie	41
Pytania	41
Rozdział 2. Praca z danymi Dockera	43
Wymagania techniczne	43
Dlaczego w ogóle potrzebujesz mechanizmu trwałego przechowywania danych?	44
Woluminy Dockera	45
Tworzenie woluminu z poziomu powłoki	46
Montowanie woluminu w kontenerze	48
Montowanie istniejącego woluminu	49
Montowanie woluminu w wielu kontenerach	50
Wyświetlanie woluminów Dockera	51
Usuwanie woluminów	51
Dołączane punkty montowania w Dockerze	53
Tymczasowy system plików w Dockerze	55
Używanie systemu tmpfs w kontenerze	56
Podsumowanie	58
Pytania	58
Rozdział 3. Sieć w Dockerze	60
Wymagania techniczne	60
Obsługa sieci w Dockerze	61
Krótkie wprowadzenie do używania portów przez TCP/IP	61
Dołączanie portu do usługi	63
Sterowniki sieciowe Dockera	63
Domyślna sieć mostu	65
Wyświetlanie dostępnych sieci	66
Pobieranie informacji szczegółowych dotyczących sieci	67
Samodzielne tworzenie sieci typu most	68
Połączenie kontenera z siecią zdefiniowaną przez użytkownika	69
Zmiana sieci w uruchomionym kontenerze	69
Usunięcie sieci	70
Uruchomienie kontenera bez obsługi sieci	71
Udostępnianie usług kontenera	72
Udostępnianie portów za pomocą sieci hosta	72
Udostępnianie portów za pomocą sieci typu most	74
Podsumowanie	77
Pytania	77

Część II. Tworzenie klastra programistycznego Kubernetes, poznawanie obiektów i udostępnianie usług	79
<u>Rozdział 4. Wdrażanie Kubernetes za pomocą KinD</u>	81
Wymagania techniczne	82
Wprowadzenie do obiektów i komponentów Kubernetes	83
Praca z klastrem	84
Używanie klastrów programistycznych	84
Dlaczego zdecydowaliśmy się na KinD?	85
Praca z podstawowym klastrem KinD Kubernetes	86
Poznajemy obraz węzła	89
KinD i sieć Dockera	90
Instalacja KinD	93
Przygotowanie do instalacji KinD	93
Instalacja pliku binarnego KinD	94
Tworzenie klastra KinD	95
Tworzenie prostego klastra	95
Usunięcie klastra	96
Tworzenie pliku konfiguracyjnego klastra	96
Konfiguracja klastra składającego się z wielu węzłów	97
Dostosowanie do własnych potrzeb warstwy sterowania i opcji kubelet	100
Tworzenie własnego klastra KinD	101
Instalacja Calico	102
Instalacja kontrolera Ingress	103
Analiza utworzonego klastra KinD	104
Obiekty pamięci masowej KinD	105
Sterowniki pamięci masowej	106
Klasy pamięci masowej w KinD	106
Używanie komponentu KinD przygotowującego pamięć masową	107
Dodawanie niestandardowego mechanizmu równoważenia obciążenia dla kontrolera Ingress	109
Przygotowanie do instalacji	110
Tworzenie konfiguracji klastra KinD	110
Wdrażanie niestandardowego kontenera HAProxy	111
Przepływ ruchu sieciowego HAProxy	113
Symulowanie awarii kubeletu	115
Usunięcie kontenera HAProxy	116
Podsumowanie	117
Pytania	117
Rozdział 5. Krótkie wprowadzenie do Kubernetes	119
Wymagania techniczne	120
Ogólne omówienie komponentów Kubernetes	120
Poznajemy warstwę sterowania	121
Serwer API w Kubernetes	121
Baza danych Etcd	122
kube-scheduler	124

Spis treści

kube-controller-manager	124
cloud-controller-manager	125
Poznajemy sposób działania komponentów węzła roboczego	125
kubelet	125
kube-proxy	125
Środowisko uruchomieniowe kontenera	125
Współpraca z serwerem API	126
Praca z narzędziem Kubernetes kubectl	126
Poznajemy opcję verbose	127
Ogólne polecenia kubectl	128
Poznajemy obiekty Kubernetes	129
Manifest w Kubernetes	130
Czym jest obiekt Kubernetes?	130
Przegląd obiektów Kubernetes	132
Podsumowanie	148
Pytania	149
Rozdział 6. Usługi, mechanizm równoważenia obciążenia i zewnętrzny serwer DNS	151
Wymagania techniczne	152
Zapewnienie żądaniom dostępu do zadań	152
Jak działa usługa?	152
Poznajemy poszczególne typy usług	157
Wprowadzenie do mechanizmu równoważenia obciążenia	163
Poznajemy model OSI	163
Mechanizmy równoważenia obciążenia działające na warstwie siódmej	164
Określanie nazw i mechanizmy równoważenia obciążenia działające na warstwie siódmej	165
Używanie usługi nip.io do określania nazw	166
Tworzenie reguł Ingress	168
Mechanizmy równoważenia obciążenia działające na warstwie czwartej	172
Opcje w zakresie mechanizmu równoważenia obciążenia na warstwie czwartej	172
Używanie MetalLB jako działającego na warstwie czwartej mechanizmu równoważenia obciążenia	172
Tworzenie usługi typu LoadBalancer	176
Dodawanie wielu pul adresów IP do MetalLB	178
Problemy związane z używaniem wielu protokołów	180
Używanie wielu protokołów z MetalLB	181
Używanie współdzielonych adresów IP	182
Udostępnianie nazw usług na zewnątrz	184
Konfiguracja projektu ExternalDNS	185
Integracja projektu ExternalDNS z CoreDNS	185
Dodawanie strefy Etcd do CoreDNS	186
Tworzenie usługi typu LoadBalancer zintegrowanej z ExternalDNS	189
Podsumowanie	195
Pytania	195

Część III. Kubernetes w korporacjach **197**

Rozdział 7. Integracja z klastrem mechanizmu uwierzytelniania **199**

Wymagania techniczne	200
Jak Kubernetes rozpoznaje użytkownika?	200
Użytkownik zewnętrzny	200
Grupy w Kubernetes	201
Konta usług	201
Poznajemy protokół OpenID Connect	202
Protokół OpenID Connect	203
Współdziałanie OIDC i API	205
Inne opcje w zakresie uwierzytelniania	209
Konfiguracja klastra KinD dla OpenID Connect	213
Spełnienie wymagań	214
Wdrażanie OIDC	217
Wprowadzenie do funkcjonalności „wcielania się w rolę” w celu integracji systemu uwierzytelniania z klastrami zarządzanymi w chmurze	230
Czym jest funkcjonalność wcielania się w rolę?	231
Kwestie związane z zapewnieniem bezpieczeństwa	232
Konfiguracja klastra do użycia funkcjonalności wcielania się w rolę	233
Testowanie rozwiązania	235
Konfiguracja funkcjonalności wcielania się w rolę bez użycia OpenUnison	236
Polityki modelu RBAC dotyczące funkcjonalności wcielania się w rolę	236
Grupy domyślne	237
Podsumowanie	237
Pytania	237

Rozdział 8. Polityki modelu RBAC i audyt **239**

Wymagania techniczne	240
Wprowadzenie do modelu RBAC	240
Czym jest rola?	241
Identyfikowanie roli	242
Role kontra obiekt typu ClusterRole	243
Odwrotność roli	244
Agregowane obiekty typu ClusterRole	245
Obiekty typu RoleBinding i ClusterRoleBinding	246
Mapowanie tożsamości użytkowników organizacji na polityki Kubernetes w celu autoryzacji dostępu do zasobów	248
Implementacja wielodostępności za pomocą przestrzeni nazw	250
Audyt w Kubernetes	252
Zdefiniowanie polityki audytu	252
Włączanie audytu w klastrze	254
Używanie audit2rbac do debugowania polityk	256
Podsumowanie	261
Pytania	261

Spis treści

Rozdział 9. Wdrażanie bezpiecznego panelu Kubernetes	263
Wymagania techniczne	264
Jak panel rozpoznaje użytkownika?	264
Architektura panelu	264
Metody uwierzytelniania	265
Niebezpieczeństwa związane z panelem Kubernetes	266
Wdrażanie niewystarczająco zabezpieczonego panelu	267
Używanie tokenu do logowania	273
Wdrożenie panelu z użyciem odwrotnego proxy	273
Panel lokalny	274
Inne aplikacje na poziomie klastra	275
Integracja panelu z OpenUnison	276
Podsumowanie	278
Pytania	278
Rozdział 10. Definiowanie polityki bezpieczeństwa poda	280
Wymagania techniczne	281
Czym jest PSP?	281
Różnice między kontenerem i maszyną wirtualną	281
Atak typu container breakout	282
Prawidłowe projektowanie kontenera	284
Czy coś się zmienia?	291
Włączenie PSP	292
Alternatywy dla PSP	297
Podsumowanie	298
Pytania	298
Rozdział 11. Poprawianie bezpieczeństwa za pomocą Open Policy Agent	300
Wymagania techniczne	301
Wprowadzenie do dynamicznych kontrolerów sterowania dopuszczeniem	301
Co to jest program typu OPA i na czym polega jego działanie?	303
Architektura OPA	303
Rego, czyli język polityki w OPA	304
GateKeeper	305
Zautomatyzowany framework testowania	306
Używanie Rego do definiowania polityki	306
Opracowanie polityki OPA	307
Testowanie polityki OPA	308
Wdrażanie polityki do GateKeeper	310
Tworzenie polityki dynamicznej	312
Debugowanie kodu w języku Rego	316
Używanie istniejącej polityki	317
Wymuszanie ograniczeń dotyczących pamięci	317
Włączanie bufora GateKeeper	318
Imitacja danych testowych	320
Budowanie i wdrażanie polityki	320

Spis treści

Wymuszanie PSP za pomocą OPA	322
Podsumowanie	323
Pytania	323
Rozdział 12. Audyt za pomocą Falco i EFK	325
Wymagania techniczne	326
Poznajemy audyt	326
Wprowadzenie do Falco	328
Poznajemy pliki konfiguracyjne Falco	328
Plik konfiguracyjny falco.yaml	329
Pliki konfiguracyjne reguł Falco	333
Definiowanie i dołączanie reguł niestandardowych	339
Wdrożenie Falco	340
Moduł jądra Falco	341
Tworzenie modułu jądra na podstawie zainstalowanych nagłóweków jądra	342
Używanie nagłóweków jądra do utworzenia modułu Falco	343
Tworzenie modułu jądra za pomocą narzędzia driverkit	344
Używanie modułu w klastrze	347
Używanie modułu w KinD	347
Wdrożenie z użyciem manifestu DaemonSet	348
Wdrażanie stosu EFK	351
Podsumowanie	369
Pytania	369
Rozdział 13. Tworzenie kopii zapasowej	371
Wymagania techniczne	372
Kopie zapasowe w Kubernetes	372
Tworzenie kopii zapasowej Etcd	373
Tworzenie kopii zapasowej wymaganych certyfikatów	373
Tworzenie kopii zapasowej bazy danych Etcd	373
Poznajemy narzędzie Velero Heptio i jego konfigurację	375
Wymagania Velero	375
Instalacja działającego w powłoce narzędzia Velero	376
Instalacja Velero	376
Używanie Velero do tworzenia kopii zapasowej	382
Jednorazowe utworzenie kopii zapasowej klastra	382
Harmonogram tworzenia kopii zapasowej klastra	386
Tworzenie niestandardowej kopii zapasowej	387
Zarządzanie Velero za pomocą narzędzia działającego w powłoce	388
Najczęściej używane polecenia Velero	390
Przywracanie z kopii zapasowej	392
Przywracanie w akcji	392
Przywrócenie przestrzeni nazw	395
Używanie kopii zapasowej do przywrócenia danych w nowym klastrze	396
Przywrócenie kopii zapasowej w nowym klastrze	398
Podsumowanie	400
Pytania	401

Spis treści

Rozdział 14. Przygotowywanie platformy	402
Wymagania techniczne	403
Opracowanie potoku	403
Najważniejsze istniejące platformy	405
Zabezpieczanie potoku	406
Określenie wymagań platformy	406
Wybór stosu technologii	409
Przygotowanie klastra	410
Wdrażanie cert-manager	411
Wdrażanie rejestru kontenerów Dockera	413
Wdrażanie OpenUnison	414
Wdrażanie GitLab	417
Tworzenie przykładowych projektów	420
Wdrażanie Tekton	421
Tworzenie aplikacji typu Witaj, świecie!	422
Kompilacja automatyczna	427
Wdrażanie ArgoCD	428
Automatyzacja tworzenia projektu z użyciem OpenUnison	431
Integracja z GitLab	434
Integracja z ArgoCD	435
Uaktualnienie OpenUnison	436
Podsumowanie	437
Pytania	438
Odpowiedzi na pytania	439