

Inhaltsverzeichnis

Vorwort	1
1 Einführung	7
1.1 Schnelligkeit	8
1.1.1 Der Wert der Immutabilität	9
1.1.2 Deklarative Konfiguration	10
1.1.3 Selbstheilende Systeme	11
1.2 Ihren Service und Ihre Teams skalieren	11
1.2.1 Entkoppeln	11
1.2.2 Einfaches Skalieren für Anwendungen und Cluster	12
1.2.3 Entwicklungs-Teams mit Microservices skalieren	13
1.2.4 Konsistenz und Skalierung durch Separation of Concerns	14
1.3 Abstrahieren Sie Ihre Infrastruktur	16
1.4 Effizienz	17
1.5 Zusammenfassung	18
2 Container erstellen und ausführen	19
2.1 Container-Images	20
2.1.1 Das Docker-Image-Format	20
2.2 Anwendungs-Images mit Docker bauen	22
2.2.1 Dockerfiles	22
2.2.2 Sicherheit von Images	23
2.2.3 Die Image-Größe optimieren	23
2.3 Images in einer Remote-Registry ablegen	24
2.4 Die Docker Container Runtime	25
2.4.1 Container mit Docker ausführen	26
2.4.2 Die kuard-Anwendung erforschen	26
2.4.3 Den Ressourcen-Einsatz begrenzen	26
2.5 Aufräumen	27
2.6 Zusammenfassung	28

3	Ein Kubernetes-Cluster deployen	29
3.1	Kubernetes auf einem öffentlichen Cloud-Provider installieren	29
3.1.1	Google Container Service	30
3.1.2	Kubernetes auf dem Azure Container Service installieren . . .	30
3.1.3	Kubernetes auf den Amazon Web Services installieren	31
3.2	Kubernetes mit minikube lokal installieren	31
3.3	Kubernetes auf dem Raspberry Pi ausführen	32
3.4	Der Kubernetes-Client	32
3.4.1	Den Cluster-Status prüfen	32
3.4.2	Worker-Knoten in Kubernetes auflisten	33
3.5	Cluster-Komponenten	35
3.5.1	Kubernetes-Proxy	36
3.5.2	Kubernetes-DNS	36
3.5.3	Kubernetes-UI	36
3.6	Zusammenfassung	37
4	Häufige kubectl-Befehle	39
4.1	Namensräume	39
4.2	Kontexte	39
4.3	Objekte der Kubernetes-API anzeigen	40
4.4	Kubernetes-Objekte erstellen, aktualisieren und löschen	41
4.5	Objekte mit einem Label und Annotations versehen	41
4.6	Debugging-Befehle	42
4.7	Zusammenfassung	43
5	Pods	45
5.1	Pods in Kubernetes	46
5.2	In Pods denken	46
5.3	Das Pod-Manifest	47
5.3.1	Einen Pod erstellen	48
5.3.2	Ein Pod-Manifest schreiben	49
5.4	Pods starten	49
5.4.1	Pods auflisten	50
5.4.2	Pod-Details	50
5.4.3	Einen Pod löschen	51

5.5	Auf Ihren Pod zugreifen	52
5.5.1	Port-Forwarding einsetzen	52
5.5.2	Mehr Informationen aus Logs erhalten	52
5.5.3	Befehle in Ihrem Container mit exec ausführen	53
5.5.4	Dateien von und auf Container kopieren	53
5.6	Health-Checks	54
5.6.1	Liveness-Probe	54
5.6.2	Readiness-Probe	56
5.6.3	Arten von Health-Checks	56
5.7	Ressourcen-Management	56
5.7.1	Ressourcen-Anforderungen: Minimal notwendige Ressourcen	57
5.7.2	Den Ressourcen-Einsatz durch Grenzen beschränken	59
5.8	Daten mit Volumes persistieren	59
5.8.1	Volumes in Pods definieren	60
5.8.2	Volumes in Pods nutzen	60
5.8.3	Daten auf Remote-Speicher persistieren	61
5.9	Fügen Sie alles zusammen	62
5.10	Zusammenfassung	64
6	Labels und Annotations	65
6.1	Labels	65
6.1.1	Labels anwenden	66
6.1.2	Labels anpassen	68
6.1.3	Label-Selektoren	68
6.1.4	Label-Selektoren in API-Objekten	70
6.2	Annotations	71
6.2.1	Annotations definieren	72
6.3	Aufräumen	73
6.4	Zusammenfassung	73
7	Service-Discovery	75
7.1	Was ist Service-Discovery?	75
7.2	Das Service-Objekt	76
7.2.1	Service-DNS	77
7.2.2	Readiness-Checks	78
7.3	Über das Cluster hinausschauen	80
7.4	Cloud-Integration	81

7.5	Weitere Details	82
7.5.1	Endpunkte	82
7.5.2	Manuelle Service-Discovery	83
7.5.3	kube-proxy und Cluster-IPs	84
7.5.4	Umgebungsvariablen zur Cluster-IP	85
7.6	Aufräumen	86
7.7	Zusammenfassung	86
8	ReplicaSets	87
8.1	Reconciliation-Schleifen	88
8.2	Die Verbindung zwischen Pods und ReplicaSets	88
8.2.1	Bestehende Container übernehmen	89
8.2.2	Container in Quarantäne stecken	89
8.3	Mit ReplicaSets designen	90
8.4	Spezifikation eines ReplicaSets	90
8.4.1	Pod-Templates	90
8.4.2	Labels	91
8.5	Ein ReplicaSet erstellen	91
8.6	Ein ReplicaSet untersuchen	92
8.6.1	Ein ReplicaSet über einen Pod finden	92
8.6.2	Eine Gruppe von Pods für ein ReplicaSet finden	93
8.7	ReplicaSets skalieren	93
8.7.1	Imperatives Skalieren mit kubectl scale	93
8.7.2	Deklaratives Skalieren mit kubectl apply	94
8.7.3	Ein ReplicaSet automatisch skalieren	94
8.8	ReplicaSets löschen	96
8.9	Zusammenfassung	96
9	DaemonSets	97
9.1	Der DaemonSet-Scheduler	98
9.2	DaemonSets erstellen	98
9.3	DaemonSets auf bestimmte Knoten beschränken	100
9.3.1	Knoten mit Labels versehen	100
9.3.2	Knoten-Selektoren	101
9.4	Ein DaemonSet aktualisieren	102
9.4.1	Ein DaemonSet durch das Löschen der einzelnen Pods aktualisieren	102
9.4.2	Rollierendes Update eines DaemonSet	103

9.5	Ein DaemonSet löschen	104
9.6	Zusammenfassung	104
10	Jobs	105
10.1	Das Job-Objekt	105
10.2	Job-Muster	106
10.2.1	Einmalig	106
10.2.2	Parallelism	111
10.2.3	Work-Queues	113
10.3	Zusammenfassung	117
11	ConfigMaps und Secrets	119
11.1	ConfigMaps	119
11.1.1	ConfigMaps erstellen	119
11.1.2	Eine ConfigMap verwenden	121
11.2	Secrets	124
11.2.1	Secrets erstellen	124
11.2.2	Secrets konsumieren	125
11.2.3	Private Docker-Registries	127
11.3	Namensbeschränkungen	128
11.4	ConfigMaps und Secrets managen	128
11.4.1	Ausgabe	129
11.4.2	Erstellen	129
11.4.3	Aktualisieren	130
11.5	Zusammenfassung	132
12	Deployments	133
12.1	Ihr erstes Deployment	134
12.1.1	Deployment-Interna	134
12.2	Deployments erstellen	135
12.3	Deployments verwalten	137
12.4	Deployments aktualisieren	137
12.4.1	Ein Deployment skalieren	138
12.4.2	Ein Container-Image aktualisieren	138
12.4.3	Rollout-History	139
12.5	Deployment-Strategien	142
12.5.1	Recreate-Strategie	143
12.5.2	RollingUpdate-Strategie	143

12.5.3	Rollouts verlangsamen, um die Service-Qualität sicherzustellen	146
12.6	Ein Deployment löschen	148
12.7	Zusammenfassung	148
13	Storage-Lösungen in Kubernetes integrieren	149
13.1	Externe Services importieren	150
13.1.1	Services ohne Selektoren	151
13.1.2	Grenzen für externe Services: Health-Checking	153
13.2	Zuverlässige Singletons ausführen	153
13.2.1	Ein MySQL-Singleton ausführen	154
13.2.2	Dynamisches Volume-Provisioning	157
13.3	Kubernetes-eigenes Storage mit StatefulSets	159
13.3.1	Eigenschaften von StatefulSets	159
13.3.2	Manuell replizierte MongoDB mit StatefulSets	159
13.3.3	Das MongoDB-Cluster automatisch erstellen	162
13.3.4	Persistente Volumes und StatefulSets	165
13.3.5	Zum Abschluss: Readiness-Proben	166
13.4	Zusammenfassung	166
14	Reale Anwendungen deployen	167
14.1	Parse	167
14.1.1	Voraussetzungen	167
14.1.2	Den Parse-Server bauen	168
14.1.3	Den Parse-Server deployen	168
14.1.4	Parse testen	169
14.2	Ghost	169
14.2.1	Ghost konfigurieren	170
14.3	Redis	173
14.3.1	Redis konfigurieren	173
14.3.2	Einen Redis-Service erstellen	175
14.3.3	Redis deployen	175
14.3.4	Mit unserem Redis-Cluster experimentieren	177
14.4	Zusammenfassung	178
Anhang A	Ein Kubernetes-Cluster aus Raspberry Pis bauen	179
Index		187