

NETWORK

Service

„Verbindet Pods, wo immer sie sind.“

Ein Service ist eine Abstraktion, die eine Gruppe von Pods zusammenfasst und eine stabile IP-Adresse sowie DNS-Namen bereitstellt. Er ermöglicht den Zugriff auf diese Pods von anderen Anwendungen oder Nutzern, selbst wenn sich die Pods auf verschiedenen Nodes befinden.

WORKLOAD

Daemonset

„Ein Pod pro Node, immer.“

Ein DaemonSet stellt sicher, dass eine Kopie eines bestimmten Pods auf jeder Node im Cluster ausgeführt wird. Es wird häufig für den Einsatz systemnaher Dienste wie Logging- oder Monitoring-Agenten auf allen Nodes verwendet. Wenn neue Nodes hinzugefügt werden, wird der Pod automatisch darauf bereitgestellt.

ROLE BASED ACCESS CONTROL

Role

„Berechtigungen im Fokus, Kontrolle auf den Punkt.“

Eine Rolle definiert Berechtigungen innerhalb eines Namespaces und regelt den Zugriff auf Kubernetes-Ressourcen. Sie sorgt dafür, dass Benutzer oder Anwendungen nur die benötigten Aktionen ausführen können.

ROLE BASED ACCESS CONTROL

ClusterRole

„Clusterweite Berechtigungen, globale Kontrolle.“

Eine ClusterRole ist ähnlich einer Rolle, jedoch gelten ihre Berechtigungen für das gesamte Cluster und nicht nur für einen einzelnen Namespace. Sie wird verwendet, um den Zugriff auf clusterweite Ressourcen zu gewähren oder konsistente Berechtigungen über mehrere Namespaces hinweg anzuwenden.

CONFIGURATION

ConfigMap

„Konfigurationen auf einen Blick.“

Eine ConfigMap speichert Schlüssel-Wert-Paare, die Pods für Konfigurationsdaten nutzen können. Sie entkoppelt umgebungs-spezifische Konfigurationen von Container-Images, wodurch die Verwaltung und Aktualisierung von Einstellungen ohne Neubau der Images vereinfacht wird.

STORAGE

PersistentVolumeClaim

„Beantreue deinen Speicher, sichere deine Daten.“

Ein PersistentVolumeClaim (PVC) ist eine Anfrage nach Speicher durch einen Benutzer. Es abstrahiert die Details der Speicherbereitstellung und verbindet deine Pods mit PersistentVolumes, die den physischen Speicher bereitstellen. PVCs stellen sicher, dass deine Daten auch dann verfügbar bleiben, wenn Pods verschoben oder gelöscht werden.

MISCELLANEOUS

Node

„Das Rückgrat deines Clusters.“

Ein Node ist eine physische oder virtuelle Maschine, die deine Anwendungen und Workloads in Kubernetes ausführt. Es ist das grundlegende Bauteil eines Clusters und stellt die CPU, den Arbeitsspeicher und die Netzwerkressourcen zur Verfügung, die zum Ausführen deiner Pods notwendig sind.

EVENT CARD

Node-Ausfall

„Ein Node fällt aus – das Cluster bleibt am Leben!“

Ein Node-Ausfall-Ereignis tritt auf, wenn eine der Nodes im Cluster nicht mehr erreichbar ist oder ihre Funktion einstellt. Kubernetes verschiebt die Pods der ausgefallenen Node automatisch auf andere verfügbare Nodes, um die Anwendungskontinuität sicherzustellen. Dies testet die Resilienz deines Clusters und hebt die Bedeutung der Ressourcenverteilung hervor.

NETWORK

Ingress

„Dein Gateway zum Cluster.“

Ein Ingress ist ein k8s Objekt, das externen Zugriff auf Cluster-Dienste, meist über HTTP oder HTTPS, ermöglicht. Er leitet den Datenverkehr anhand von Regeln wie URL-Pfaden zu den passenden Diensten und erlaubt es, mehrere Dienste über eine einzige IP-Adresse bereitzustellen.

WORKLOAD

Pod

„Die kleinstmögliche Ausführungseinheit.“

Ein Pod ist das kleinste und einfachste Kubernetes-Objekt, das eine einzelne Instanz eines laufenden Prozesses in deinem Cluster darstellt. Er kann ein oder mehrere Container enthalten, die denselben Netzwerk-Namespace und Speicher teilen. Pods sind die grundlegenden Bausteine, um Anwendungen in Kubernetes bereitzustellen.

ROLE BASED ACCESS CONTROL

ServiceAccount

„Identität für deine Pods.“

Ein ServiceAccount stellt eine Identität für Prozesse zur Verfügung, die in einem Pod laufen, und ermöglicht diesen, mit der Kubernetes-API zu interagieren. Er gewährt Berechtigungen und Tokens, die steuern, welche Aktionen der Pod innerhalb des Clusters durchführen kann.

MISCELLANEOUS

CustomResourceDefinition

„Erweitere Kubernetes nach deinen Wünschen.“

Eine CustomResourceDefinition (CRD) erlaubt es dir, eigene Ressourcen in Kubernetes zu erstellen, die über die Standardobjekte hinausgehen. CRDs erweitern Kubernetes, sodass du anwendungsspezifische Konfigurationen oder Funktionen definieren und verwalten kannst.

CONFIGURATION

Secret

„Halte es sicher, halte es geheim.“

Ein Secret speichert sensible Daten wie Passwörter, Tokens oder Schlüssel sicher im Cluster. Es trennt diese vom Anwendungscode und schützt sie vor Klartext-Offenlegung, was die Sicherheit in Kubernetes verbessert.

STORAGE

StorageClass

„Definiere deine Speicherstrategie.“

Eine StorageClass bietet eine Möglichkeit, die verschiedenen Klassen von Speicher in einem Kubernetes-Cluster zu beschreiben. Sie abstrahiert die Speicherbereitstellung und ermöglicht es PVCs, automatisch bestimmte Speicherarten anzufordern (z.B. SSDs, NFS). StorageClasses ermöglichen eine flexible und automatisierte Speicherverwaltung.

MISCELLANEOUS

Namespace

„Abgrenzung innerhalb des Clusters.“

Ein Namespace ist eine logische Partition eines Kubernetes-Clusters, die dazu dient, Ressourcen voneinander zu isolieren und zu organisieren. Er ermöglicht mehrere Projekte oder Teams innerhalb desselben Clusters, ohne sich gegenseitig zu stören. Namespaces helfen, Konflikte zu vermeiden und vereinfachen das Ressourcenmanagement.

EVENT CARD

Aufgedeckte CVE

„Sicherheitslücke entdeckt – Patchen oder verlieren!“

Ein offengelegtes CVE Ereignis bedeutet, dass eine Sicherheitslücke in einem deiner Container oder Kubernetes-Komponenten entdeckt wurde. Sofortige Maßnahmen sind erforderlich, um die betroffenen Systeme zu patchen oder anfällige Images zu aktualisieren.



IngressController

„Verkehrskontrolle an der Grenze.“

Ein IngressController implementiert die Regeln einer Ingress-Ressource. Er verarbeitet eingehende HTTP/S-Anfragen und leitet sie an die entsprechenden Services im Cluster weiter. IngressController sind entscheidend für das Management des externen Verkehrs und sorgen dafür, dass dieser die richtigen Endpunkte erreicht.



ReplicaSet

„Halte Deine Pods am Laufen.“

Ein ReplicaSet stellt sicher, dass eine festgelegte Anzahl identischer Pods immer im Cluster läuft. Es ersetzt automatisch Pods, die fehlgeschlagen oder gelöscht werden, und stellt so den gewünschten Zustand sicher. ReplicaSets sind entscheidend für die Gewährleistung der hohen Verfügbarkeit und Skalierung deiner Anwendungen.



RoleBinding

„Verknüpfung von Rollen mit Benutzern.“

Ein RoleBinding gewährt einem Benutzer, einer Gruppe oder einem ServiceAccount innerhalb eines bestimmten Namespaces spezifische Berechtigungen, indem es sie mit einer Rolle verknüpft. Es ist die Art und Weise, wie die Zugriffskontrollregeln, die in einer Rolle definiert sind, auf tatsächliche Benutzer oder Dienste angewendet werden.



ClusterRoleBinding

„Clusterweite Zugriffskontrolle.“

Ein ClusterRoleBinding gewährt einem Benutzer, einer Gruppe oder einem ServiceAccount clusterweite Berechtigungen, indem es sie mit einer ClusterRole verknüpft. Es ermöglicht, Zugriffskontrollregeln auf alle Namensräume anzuwenden. ClusterRoleBindings sind entscheidend für das Management von Sicherheit und Berechtigungen auf globaler Ebene.



Job

„Führe Aufgaben bis zum Abschluss aus.“

Ein Job erstellt einen oder mehrere Pods, die bis zum Abschluss laufen und sicherstellen, dass eine bestimmte Aufgabe eine definierte Anzahl von Malen ausgeführt wird. Sobald die Pods ihre Aufgabe abgeschlossen haben, wird der Job als abgeschlossen betrachtet. Jobs eignen sich ideal für einmalige oder seltene Batch-Verarbeitungen.



PersistentVolume

„Bereitgestellter Speicher für Deinen Cluster.“

Ein PersistentVolume (PV) ist ein Speicherstück im Cluster, das von einem Administrator oder dynamisch durch eine StorageClass bereitgestellt wurde. Es wird von PersistentVolumeClaims verwendet, um Daten über die Lebensdauer einzelner Pods hinaus zu speichern. PVs bieten dauerhaften Speicher, der auch dann verfügbar bleibt, wenn Pods gelöscht werden.



NetworkPolicy

„Steuere den Datenverkehr, sichere deinen Cluster.“

Eine NetworkPolicy ist eine Reihe von Regeln, die definieren, wie Pods miteinander und mit externen Endpunkten kommunizieren können. Sie steuert den Datenverkehr auf Netzwerkebene und gibt an, welche Verbindungen erlaubt oder abgelehnt werden.



Pod-Neustart

„Ein Pod stürzt ab – Kubernetes startet ihn neu!“

Ein Pod-Neustart-Ereignis tritt auf, wenn ein Pod abstürzt oder unerwartet beendet wird, was Kubernetes dazu veranlasst, ihn automatisch neu zu starten. Dieser Selbstheilungsmechanismus trägt dazu bei, die Betriebszeit der Anwendung aufrechtzuerhalten und sicherzustellen, dass die Dienste trotz gelegentlicher Fehler verfügbar bleiben.

WORKLOAD

Deployment

„Verwalte und skaliere Deine Pods.“

Ein Deployment automatisiert die Erstellung, das Update und die Skalierung von Pods in einem ReplicaSet. Es stellt sicher, dass deine Anwendung konsistent läuft, indem Updates ausgerollt und nach Bedarf skaliert werden. Deployments sind entscheidend für das Management des Lebenszyklus von zustandslosen Anwendungen in Kubernetes.

WORKLOAD

StatefulSet

„Persistente Identität, stabiler Speicher.“

Ein StatefulSet verwaltet die Bereitstellung und Skalierung von Pods mit einzigartigen Identitäten und stabilem Speicher. Es wird für zustandsbehaftete Anwendungen verwendet, bei denen jeder Pod seinen Zustand über Neustarts bewahren muss. StatefulSets garantieren, dass Pods in einer konsistenten Reihenfolge laufen und stabile Namen behalten.

WORKLOAD

HorizontalPodAutoScaler

„Skaliere nach Bedarf.“

Ein HorizontalPodAutoscaler passt automatisch die Anzahl der Pods in einem Deployment, ReplicaSet oder StatefulSet basierend auf beobachteter CPU-Nutzung oder anderen ausgewählten Metriken an. Er sorgt dafür, dass deine Anwendungen mit wechselnden Lasten umgehen können, indem Ressourcen je nach Bedarf skaliert werden.

ROLE BASED ACCESS CONTROL

Gruppe

„Organisiere Benutzer, vereinfache den Zugriff.“

Eine Gruppe ist eine Sammlung von Benutzern in Kubernetes, die gemeinsam Rollen und Berechtigungen zugewiesen bekommen können. Gruppen vereinfachen das Management der Zugriffskontrollen, indem Administratoren Richtlinien für mehrere Benutzer gleichzeitig anwenden können.

WORKLOAD

CronJob

„Geplante Aufgaben, automatisiert.“

Ein CronJob erstellt Jobs basierend auf einem Zeitplan und ermöglicht es, Aufgaben in regelmäßigen Abständen auszuführen. Er eignet sich ideal für wiederkehrende Aufgaben wie Backups, Berichte oder Wartungsskripte. CronJobs automatisieren routinemäßige Operationen, sodass diese regelmäßig ohne manuelle Eingriffe ausgeführt werden.

NETWORK

EndpointSlice

„Effiziente Dienstentdeckung.“

EndpointSlices gruppieren Netzwerkendpunkte (z.B. Pods) für einen Kubernetes-Service basierend auf dessen Selector und verwalten sie nach Protokoll, Port und Servicename. Sie werden automatisch vom Control Plane erstellt und verbessern die Skalierbarkeit und Effizienz des Netzwerkverkehrs. Jeder EndpointSlice hat einen DNS-konformen Namen.

ROLE BASED ACCESS CONTROL

Benutzer

„Zugriffskontrolle beginnt hier.“

Ein Benutzer repräsentiert eine Einzelperson oder einen Dienst, der mit der Kubernetes-API interagiert. Benutzer werden authentifiziert, um festzustellen, welche Aktionen sie im Cluster ausführen dürfen. Eine ordnungsgemäße Verwaltung der Benutzer ist entscheidend für die Sicherheit deiner Kubernetes-Umgebung.

MISCELLANEOUS

CustomResource

„Maßgeschneidert auf deine Bedürfnisse.“

Eine CustomResource erweitert Kubernetes um eigene Objekttypen, die du für anwendungsspezifische Konfigurationen nutzen kannst. Sie ermöglicht es, Ressourcen zu erstellen und zu verwalten, die nicht Teil der Standard-Kubernetes-API sind.