NETWORK

Service

„Connecting Pods, Wherever They Are.“

A Service is an abstraction that defines a logical set of Pods and a policy by which to access them. It provides stable IP addresses and DNS names, allowing you to expose an application running on a set of Pods to other applications or external users. Services can route traffic to Pods across multiple nodes.

WORKLOAD

Daemonset

„One Pod Per Node, Always.“

A DaemonSet ensures that a copy of a specific Pod is running on each node in the cluster. It's commonly used for deploying system-level services like logging or monitoring agents across all nodes. When new nodes are added, the DaemonSet automatically deploys the Pod to them.

ROLE BASED ACCESS CONTROL

Role

„Scoped Permissions, Focused Control.“

A Role defines permissions within a specific namespace, controlling access to Kubernetes resources. It allows you to set granular permissions, ensuring that users or applications can only perform the actions they need. Roles are ideal for managing security within a single namespace.

ROLE BASED ACCESS CONTROL

ClusterRole

„Cluster-Wide Permissions, Global Control.“

A ClusterRole is similar to a Role, but its permissions apply across the entire cluster, not just a single namespace. It's used to grant access to cluster-wide resources or to apply consistent permissions across multiple namespaces. ClusterRoles are essential for managing large, multi-tenant clusters.

CONFIGURATION

ConfigMap

„Configuration at Your Fingertips.“

A ConfigMap stores key-value pairs that Pods can use for configuration data. It decouples environment-specific configurations from container images, making it easier to manage and update settings without rebuilding images. ConfigMaps are ideal for keeping configuration data separate from application code.

STORAGE

PersistentVolumeClaim

„Claim Your Storage, Persist Your Data.“

A PersistentVolumeClaim (PVC) is a request for storage by a user. It abstracts storage details and connects your Pods to PersistentVolumes, which provide the physical storage. PVCs ensure that your data remains available even if Pods are rescheduled or deleted.

MISCELLANEOUS

Node

„The Backbone of Your Cluster.“

A Node is a physical or virtual machine that runs your applications and workloads in Kubernetes. It's the basic building block of a cluster, providing the CPU, memory, and network resources necessary to run your Pods. Nodes are managed by the control plane, which assigns workloads and handles scheduling.

EVENT CARD

Node Failure

„A Node Goes Down - But the Cluster Stays Up!“

A Node Failure event occurs when one of your cluster's nodes becomes unreachable or stops functioning. Kubernetes automatically reschedules Pods from the failed node to other available nodes, ensuring application continuity. It tests the resilience of your cluster and highlights the importance of resource distribution.

NETWORK

Ingress

„Your Gateway to the Cluster.“

An Ingress is an API object that manages external access to services within a cluster, typically HTTP or HTTPS. It provides a way to route traffic based on rules, such as URL paths, to the appropriate services. Ingress helps you expose multiple services under a single IP address, simplifying external access management.

WORKLOAD

Pod

„The Smallest Deployable Unit.“

A Pod is the smallest and simplest Kubernetes object, representing a single instance of a running process in your cluster. It can contain one or more containers that share the same network namespace and storage. Pods are the fundamental building blocks for deploying applications in Kubernetes.

ROLE BASED ACCESS CONTROL

ServiceAccount

„Identity for Your Pods.“

A ServiceAccount provides an identity for processes that run in a Pod, allowing them to interact with the Kubernetes API. It grants permissions and tokens that control what the Pod can do within the cluster. ServiceAccounts are essential for managing secure and controlled access for applications.

MISCELLANEOUS

CustomResource-Definition

„Extend Kubernetes, Your Way.“

A CustomResourceDefinition (CRD) allows you to create custom resources that extend Kubernetes' capabilities beyond its built-in objects. CRDs enable you to define and manage application-specific configurations or behaviors. They are key to building and scaling custom solutions on top of Kubernetes.

CONFIGURATION

Secret

„Keep It Safe, Keep It Secret.“

A Secret is used to store and manage sensitive information, such as passwords, tokens, or keys, securely in your cluster. It decouples sensitive data from application code, ensuring that credentials and other secrets are not exposed in plain text. Secrets help maintain security best practices in Kubernetes.

STORAGE

StorageClass

„Define Your Storage Strategy.“

A StorageClass provides a way to describe the classesöf storage available in a Kubernetes cluster. It abstracts storage provisioning, allowing PersistentVolumeClaims to automatically request specific types of storage (e.g., SSDs, NFS). StorageClasses enable flexible and automated storage management.

MISCELLANEOUS

Namespace

„Organize and Isolate Your Resources.“

A Namespace is a way to divide cluster resources between multiple users or teams, providing isolation and resource scoping. It helps manage environments (like dev, test, prod) within the same cluster and allows for better organization and access control. Namespaces are essential for multi-tenant or complex deployments.

EVENT CARD

Disclosed CVE

„A Vulnerability Is Found - Patch or Perish!“

A Disclosed CVE (Common Vulnerabilities and Exposures) event signifies that a security vulnerability has been discovered in one of your containers or Kubernetes components. Immediate action is required to patch the affected systems or update vulnerable images. This event underscores the importance of maintaining security hygiene in your cluster.

NETWORK

IngressController

„Traffic Control at the Edge.“

An IngressController is a component that implements the rules defined by an Ingress resource. It processes incoming HTTP/HTTPS requests and routes them to the appropriate Services in the cluster. IngressControllers are essential for managing external traffic and ensuring it reaches the correct endpoints.

WORKLOAD

ReplicaSet

„Keep Your Pods Running.“

A ReplicaSet ensures that a specified number of identical Pods are always running in the cluster. It automatically replaces Pods that fail or are deleted, maintaining the desired state. ReplicaSets are crucial for ensuring high availability and scaling of your applications.

ROLE BASED ACCESS CONTROL

RoleBinding

„Linking Roles to Users.“

A RoleBinding grants specific permissions to a user, group, or ServiceAccount within a particular namespace by associating them with a Role. It's how you apply the access rules defined in a Role to actual users or services. RoleBindings help enforce security and access controls in a namespace.

ROLE BASED ACCESS CONTROL

ClusterRoleBinding

„Cluster-Wide Access Control.“

A ClusterRoleBinding grants cluster-wide permissions to a user, group, or ServiceAccount by associating them with a ClusterRole. It allows you to apply access rules across all namespaces in the cluster. ClusterRoleBindings are essential for managing security and permissions on a global scale.

WORKLOAD

Job

„Run to Completion.“

A Job creates one or more Pods that run to completion, ensuring that a specific task is executed a defined number of times. Once the Pods complete their task, the Job is considered finished. Jobs are ideal for batch processing or tasks that need to run once or a few times.

STORAGE

PersistentVolume

„Provisioned Storage for Your Cluster.“

A PersistentVolume (PV) is a piece of storage in the cluster that has been provisioned by an administrator or dynamically by a StorageClass. It is used by PersistentVolumeClaims to persist data beyond the life of individual Pods. PVs provide durable storage that remains available even when Pods are deleted.

NETWORK

NetworkPolicy

„Control Traffic, Secure Your Cluster.“

A NetworkPolicy is a set of rules that define how Pods can communicate with each other and with external endpoints. It controls traffic at the network level, specifying which connections are allowed or denied. NetworkPolicies are vital for securing communication between services and managing network segmentation.

EVENT CARD

Pod Restart

„A Pod Crashes - Kubernetes Brings It Back!“

A Pod Restart event occurs when a Pod crashes or is terminated unexpectedly, triggering Kubernetes to automatically restart it. This self-healing mechanism helps maintain application uptime and ensures that services remain available despite occasional failures. It's a reminder of Kubernetes' robustness in handling disruptions.

Deployment

WORKLOAD

„Manage and Scale Your Pods.“

A Deployment automates the creation, updating, and scaling of Pods in a ReplicaSet. It ensures that your application runs consistently by rolling out updates and scaling up or down as needed. Deployments are essential for managing the lifecycle of stateless applications in Kubernetes.

StatefulSet

WORKLOAD

„Persistent Identity, Stable Storage.“

A StatefulSet manages the deployment and scaling of Pods with unique identities and stable network storage. It's used for stateful applications, where each Pod must retain its state across restarts. StatefulSets ensure that Pods are consistently ordered and maintain stable names.

HorizontalPodAuto-Scaler

WORKLOAD

„Scale with Demand.“

A HorizontalPodAutoscaler automatically adjusts the number of Pods in a deployment, ReplicaSet, or StatefulSet based on observed CPU usage or other select metrics. It helps ensure your applications can handle varying loads by scaling resources up or down as needed.

Group

m

ROLE BASED ACCESS CONTROL

„Organize Users, Simplify Access.“

A Group is a collection of users in Kubernetes that can be assigned roles and permissions collectively. Groups simplify access control management by allowing administrators to apply policies to multiple users at once. They are essential for managing permissions in larger teams or organizations.

CronJob

WORKLOAD

„Scheduled Tasks, Automated.“

A CronJob creates Jobs on a time-based schedule, allowing you to run tasks at specific intervals. It's ideal for recurring tasks like backups, reports, or maintenance scripts. CronJobs automate routine operations, ensuring they run regularly without manual intervention.

EndpointSlice

NETWORK

„Efficient Service Discovery.“

An EndpointSlice provides a scalable way to track network endpoints, improving the efficiency of service discovery within large clusters. It breaks down the list of endpoints into smaller slices, making it easier to manage and scale services. EndpointSlices enhance the performance of networking in Kubernetes.

User

m

ROLE BASED ACCESS CONTROL

„Access Control Starts Here.“

A User represents an individual or a service that interacts with the Kubernetes API. Users are authenticated to determine what actions they can perform within the cluster. Proper management of users is crucial for securing your Kubernetes environment.

CustomResource

MISCELLANEOUS

„Tailored to Your Needs.“

A CustomResource extends Kubernetes by adding your own object types to the API, allowing you to manage application-specific configurations or processes. It works alongside CustomResourceDefinitions to let you create and control resources that Kubernetes doesn't provide out of the box. CustomResources enable powerful customization and automation.