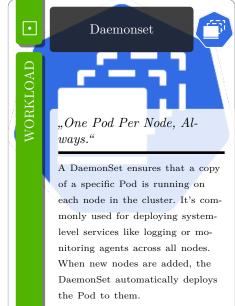
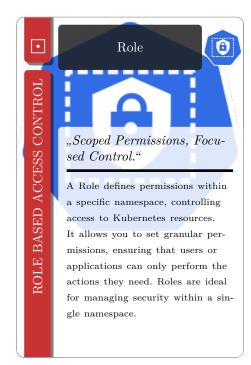
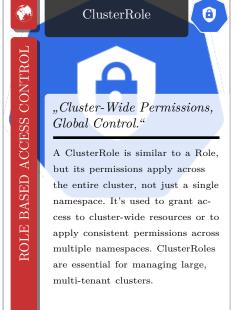


"Connecting Pods, Wherever They Are."

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.

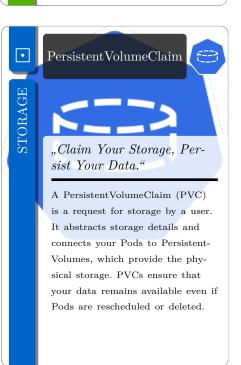


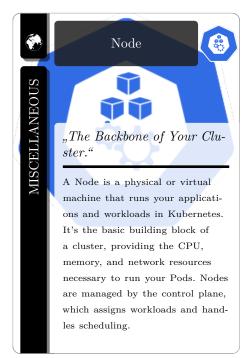


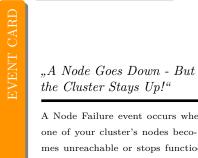


Node Failure





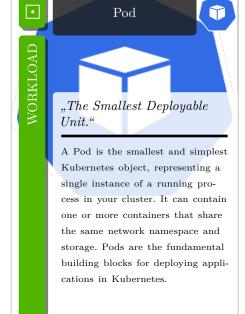


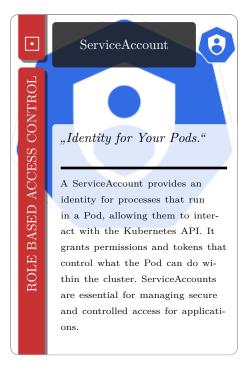


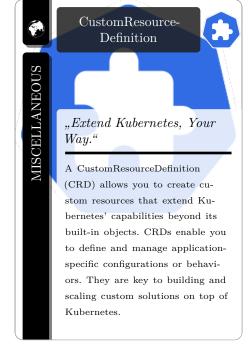
A Node Failure event occurs when one of your cluster's nodes becomes unreachable or stops functioning. Kubernetes automatically reschedules Pods from the failed node to other available nodes, ensuring application continuity. It tests the resilience of your cluster and highlights the importance of resource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



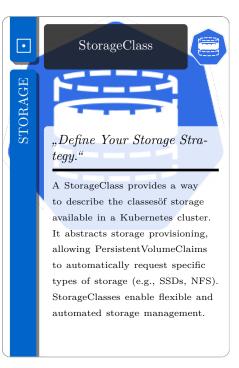
An Ingress is an API object that manages external access to services within a cluster, typically HTTP or HTTPS. It provides a way to route traffic based on rules, such as URL paths, to the appropriate services. Ingress helps you expose multiple services under a single IP address, simplifying external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



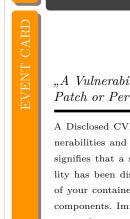












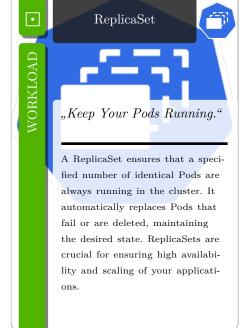
"A Vulnerability Is Found -Patch or Perish!"

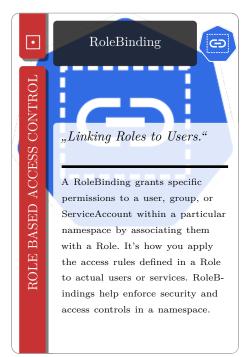
Disclosed CVE

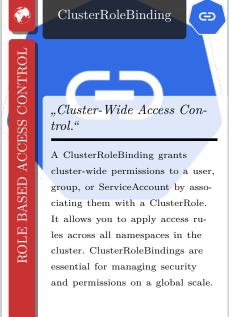
A Disclosed CVE (Common Vulnerabilities and Exposures) event signifies that a security vulnerability has been discovered in one of your containers or Kubernetes components. Immediate action is required to patch the affected systems or update vulnerable images. This event underscores the importance of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



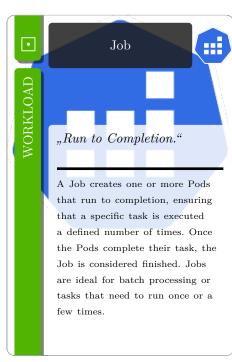
An IngressController is a component that implements the rules defined by an Ingress resource. It processes incoming HTTP/HTTPS requests and routes them to the appropriate Services in the cluster. IngressControllers are essential for managing external traffic and ensuring it reaches the correct endpoints.

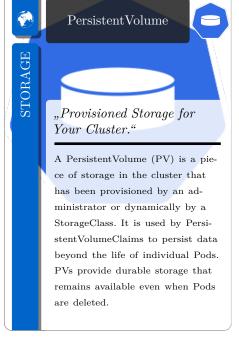


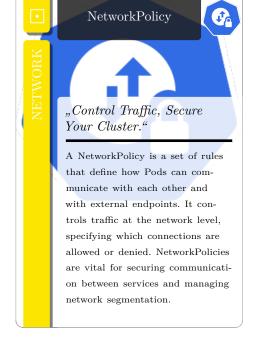


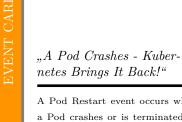


Pod Restart









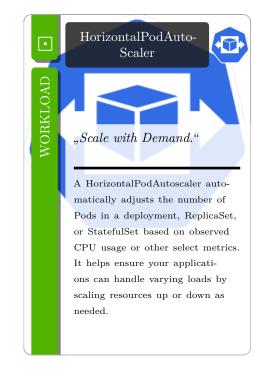
A Pod Restart event occurs when a Pod crashes or is terminated unexpectedly, triggering Kubernetes to automatically restart it. This self-healing mechanism helps maintain application uptime and ensures that services remain available despite occasional failures. It's a reminder of Kubernetes' robustness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



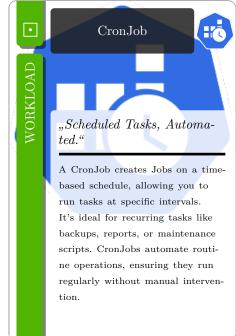
A Deployment automates the creation, updating, and scaling of Pods in a ReplicaSet. It ensures that your application runs consistently by rolling out updates and scaling up or down as needed. Deployments are essential for managing the lifecycle of stateless applications in Kubernetes.

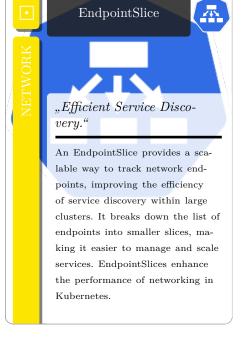


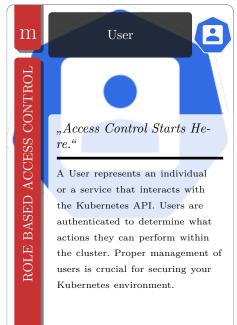
stable names.











"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. Custom-

Resources enable powerful customization and automation.

CustomResource