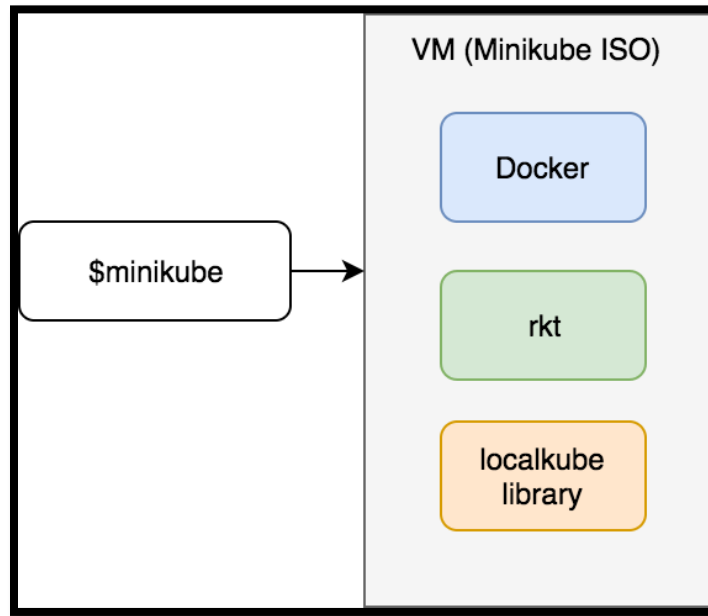


Chapter 2: Running Kubernetes Cluster Locally



The screenshot shows the Kubernetes dashboard interface. The top navigation bar includes the Kubernetes logo, a search bar, and a '+ CREATE' button. The left sidebar contains a menu with categories: Cluster, Namespaces, Nodes, Persistent Volumes, Roles, Storage Classes, Namespace (set to 'default'), Overview (selected), Workloads (Daemon Sets, Deployments, Jobs, Pods, Replica Sets, Replication Controllers, Stateful Sets), and Discovery and Load Balancing.

The main content area displays two sections:

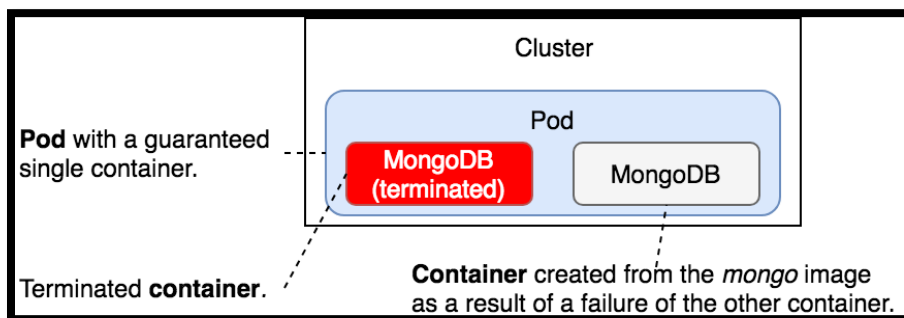
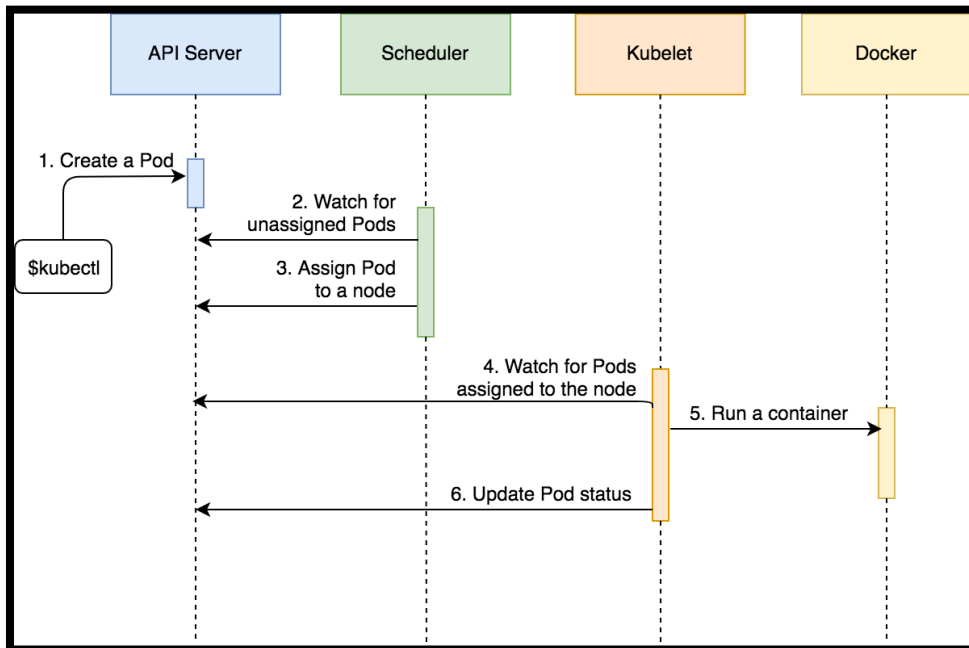
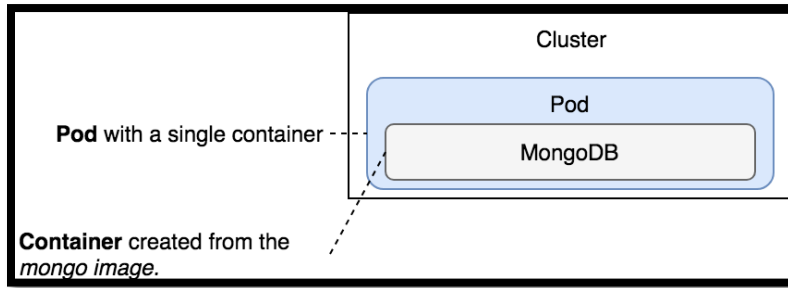
Services

Name	Labels	Cluster IP	Internal endpoints	External endpoints	Age
✓ kubernet	component: apiserver provider: kubernetes	10.0.0.1	kubernetes:443 TCP kubernetes:0 TCP	-	23 minutes

Secrets

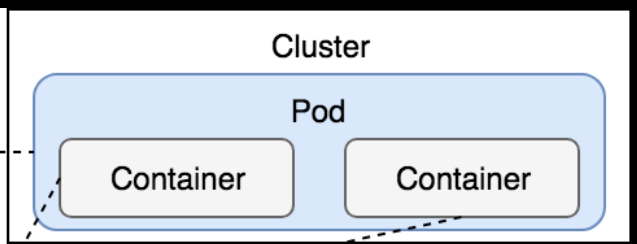
Name	Type	Age
default-token-dkkzg	kubernetes.io/service-account-token	23 minutes

Chapter 3: Creating Pods



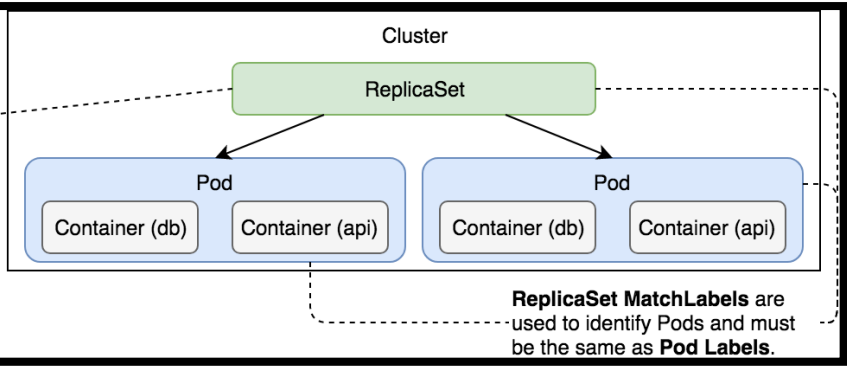
Pod: a group of one or more containers, with shared storage/network, and a specification for how to run the containers.

Container: created from an image with the purpose to host an application.

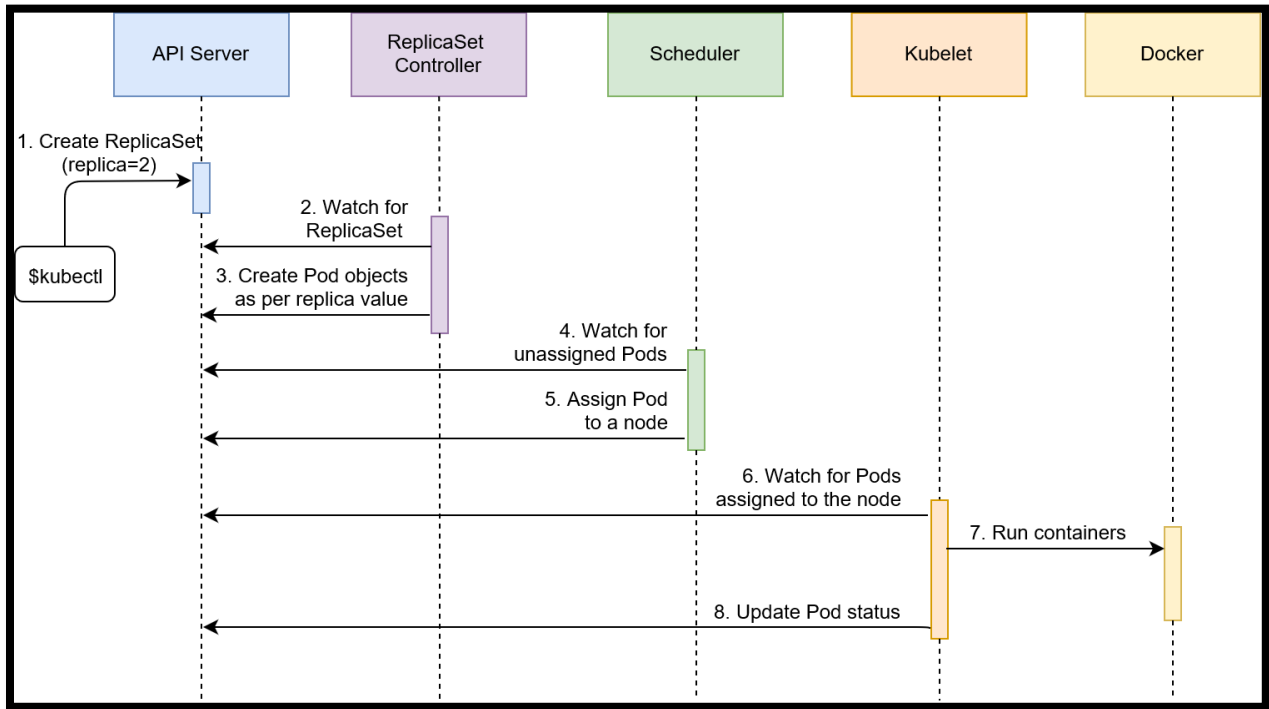


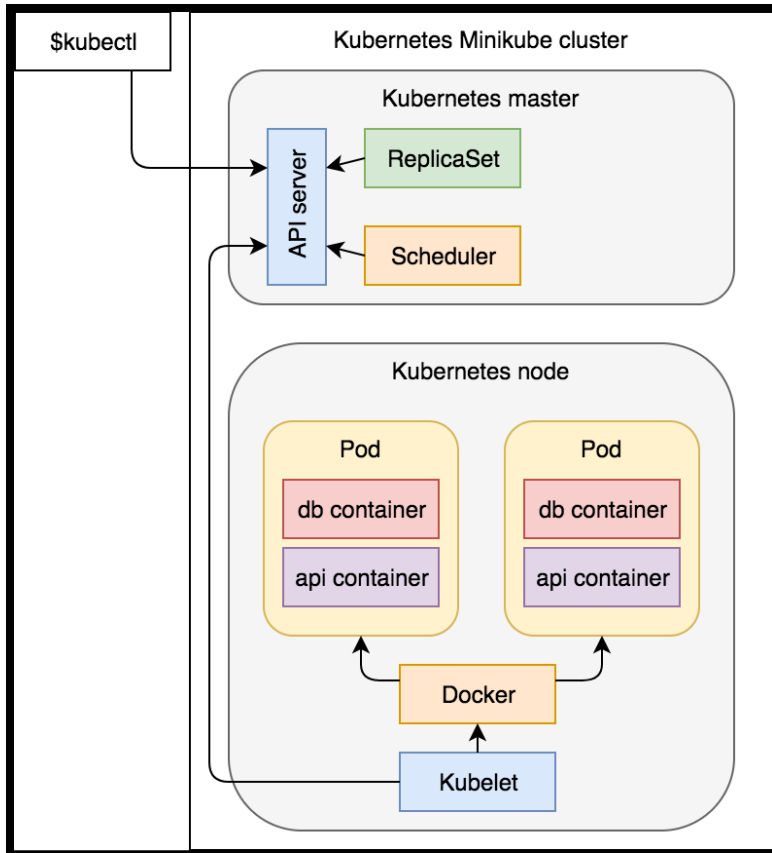
Chapter 4: Scaling Pods With ReplicaSets

ReplicaSet: ensures that a specified number of Pod replicas are running at any one time.



ReplicaSet MatchLabels are used to identify Pods and must be the same as **Pod Labels**.

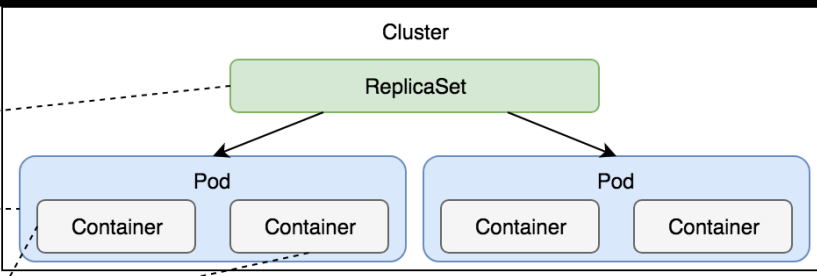




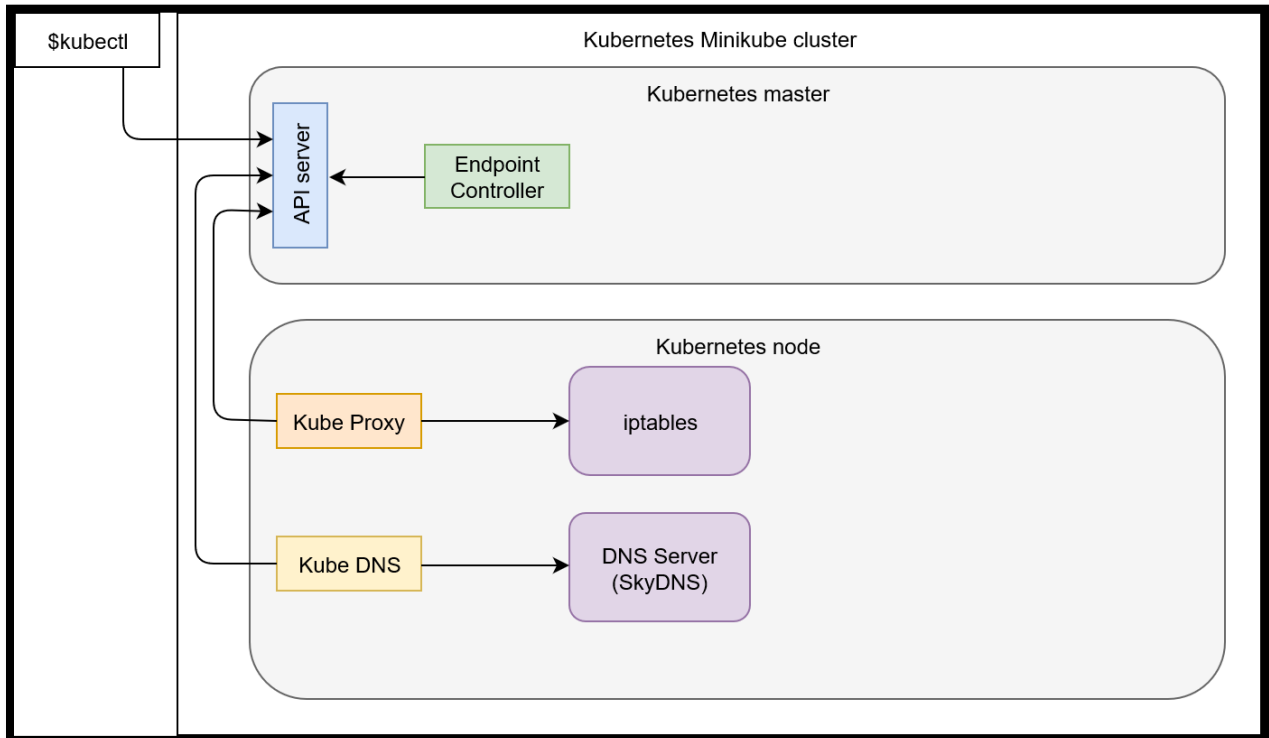
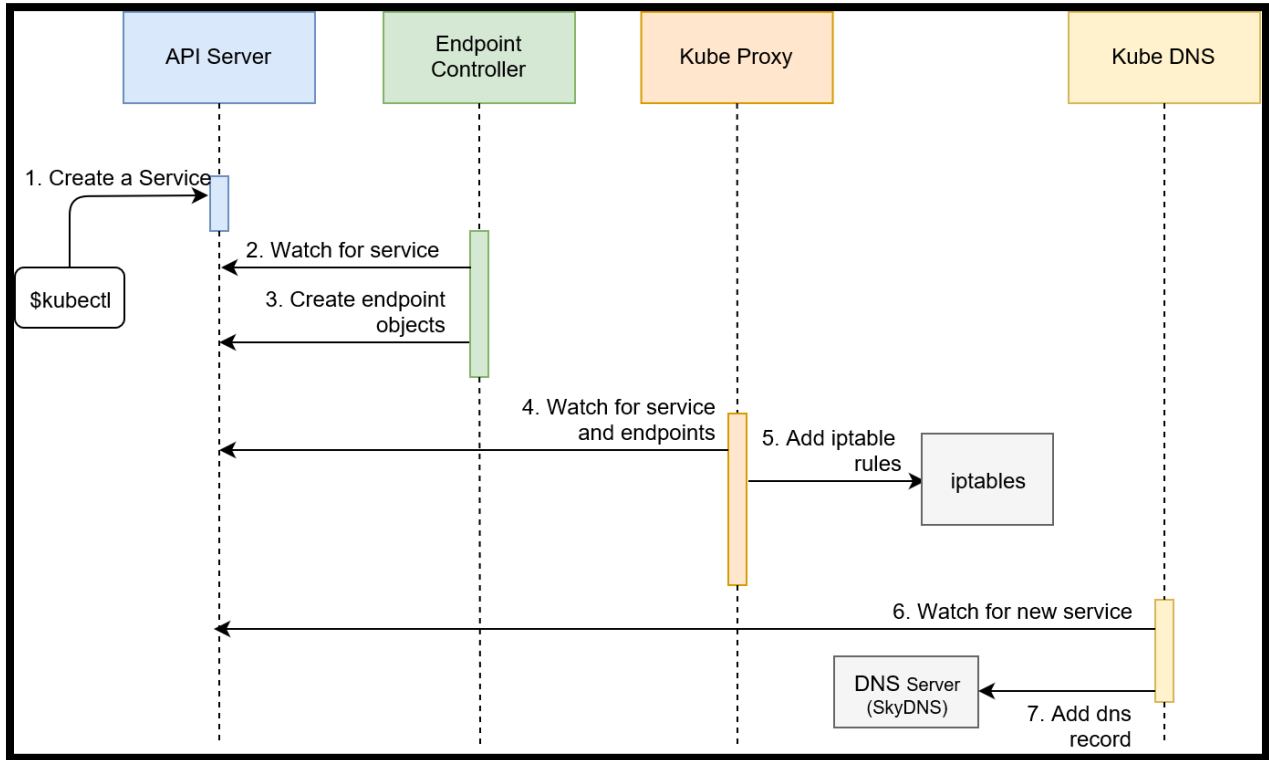
ReplicaSet: ensures that a specified number of pod replicas are running at any one time.

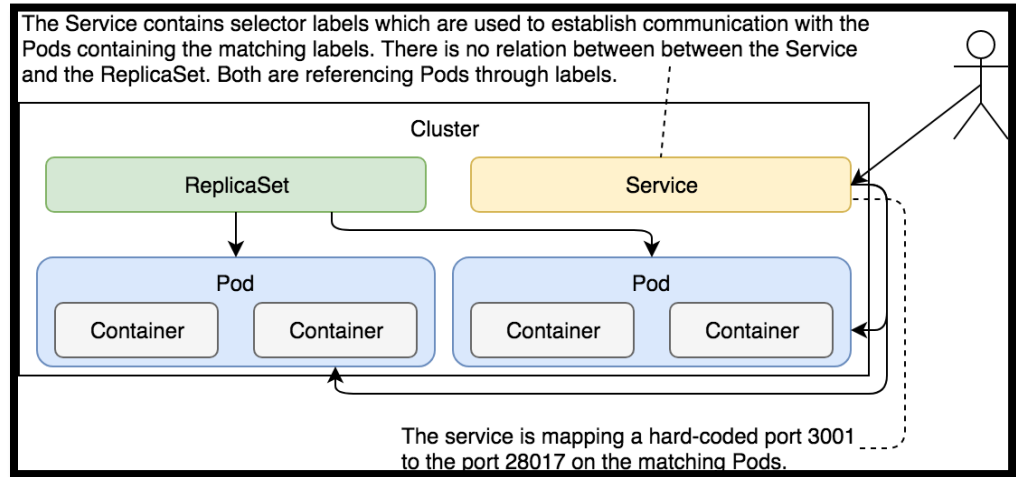
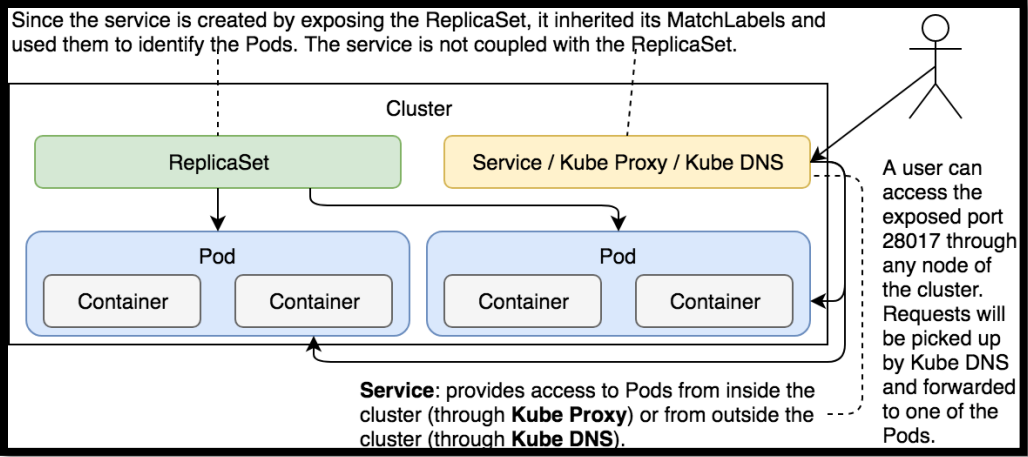
Pod: a group of one or more containers, with shared storage/network, and a specification for how to run the containers.

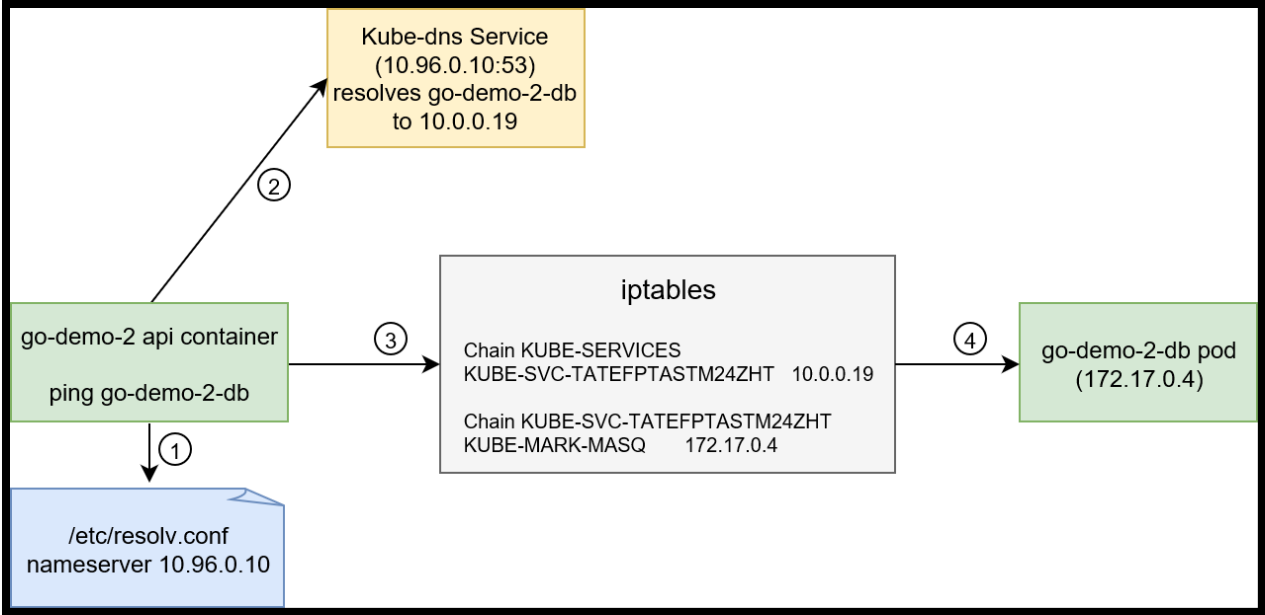
Container: created from an image with the purpose to host an application.



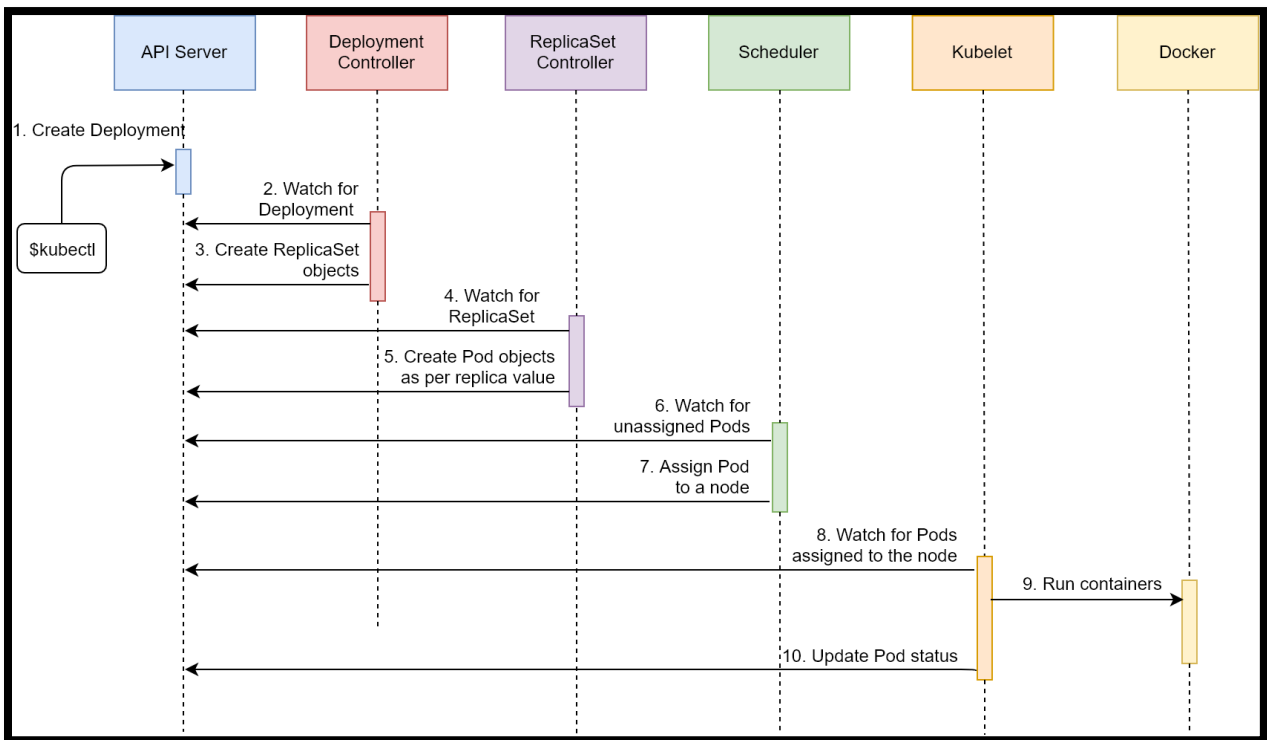
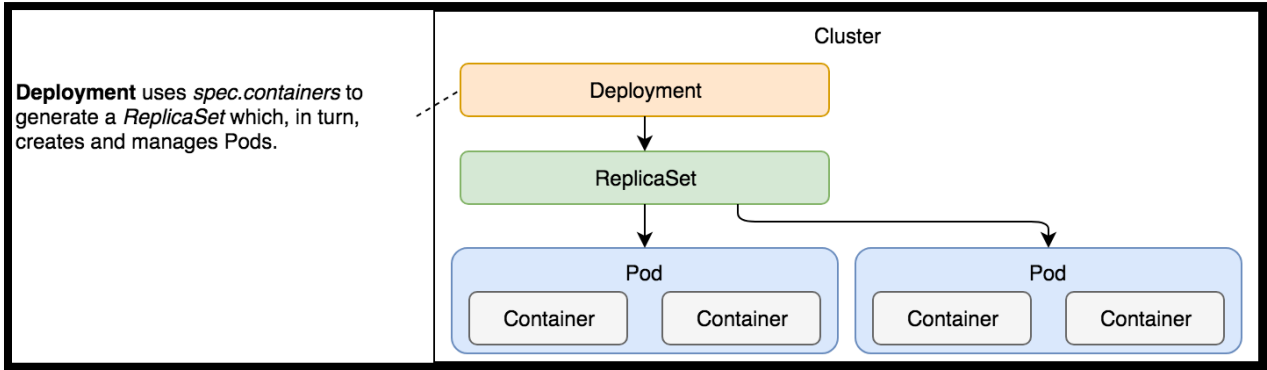
Chapter 5: Using Services to Enable Communication between Pods

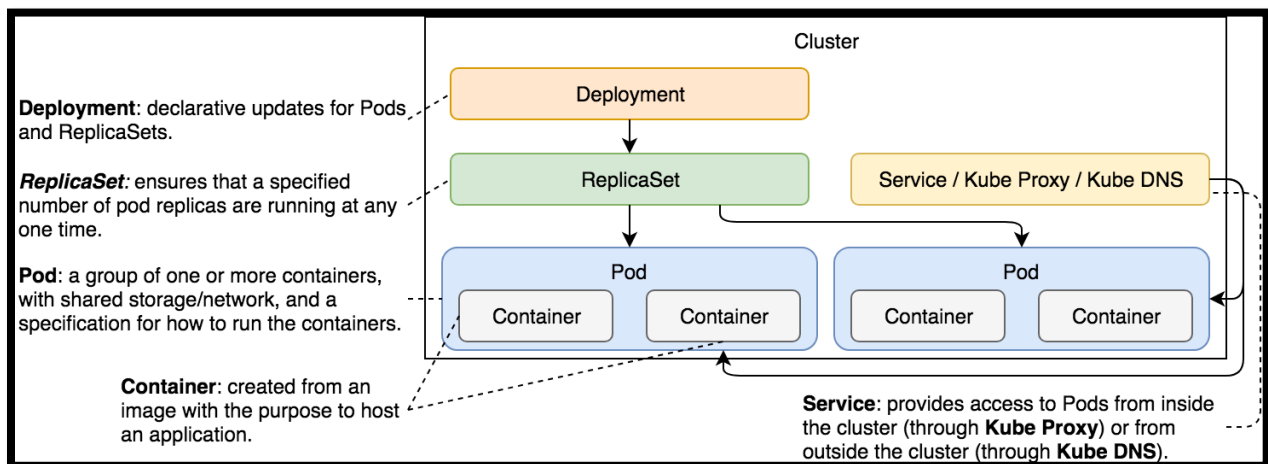




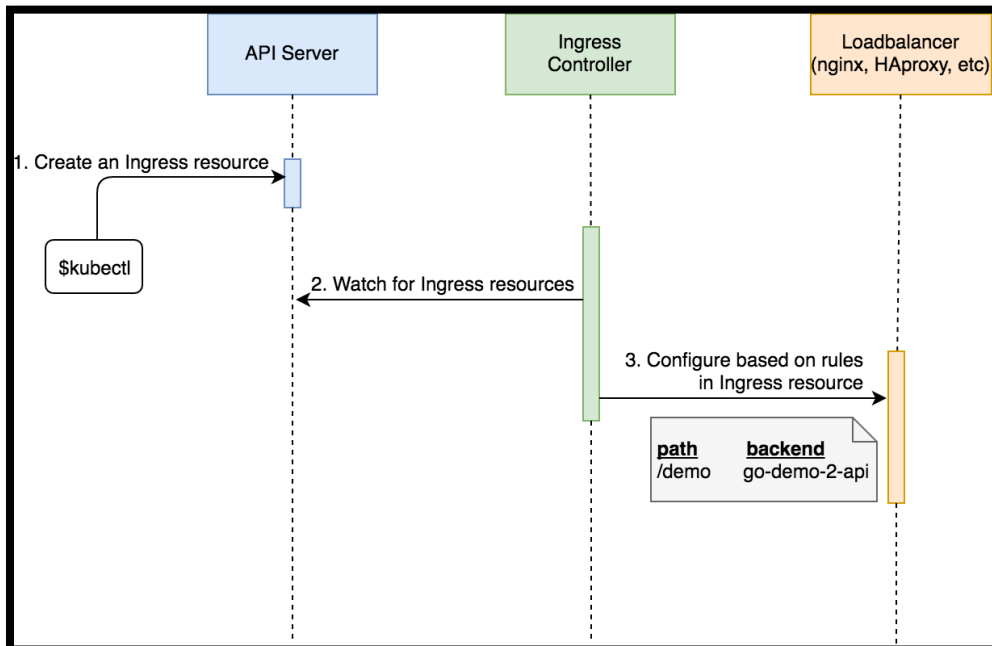
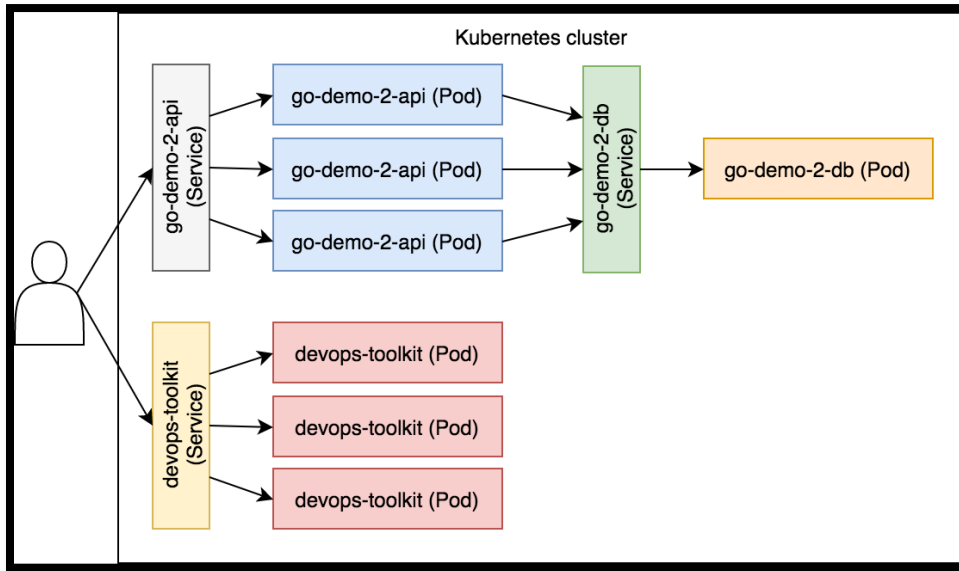


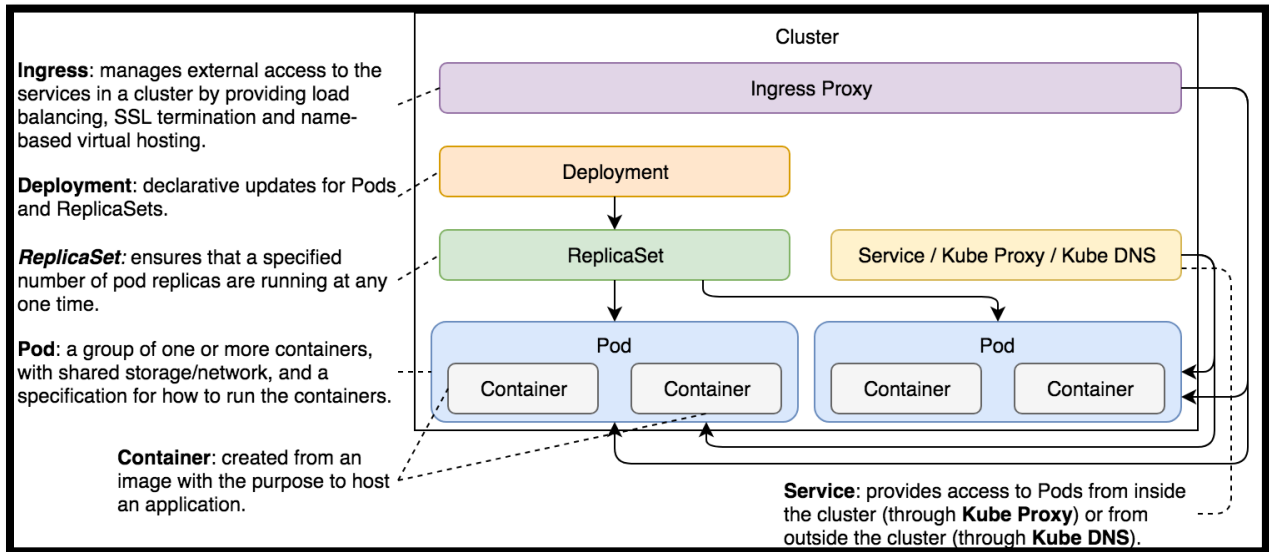
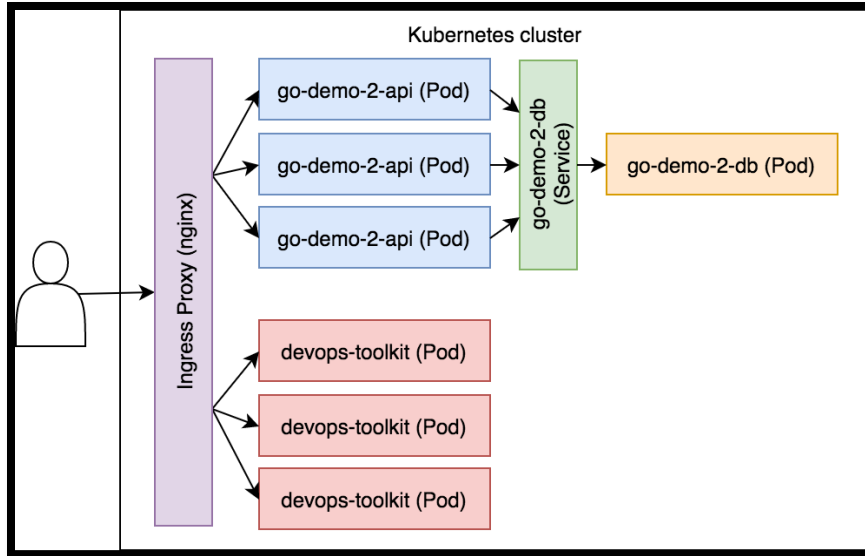
Chapter 6: Deploying Releases with Zero-Downtime



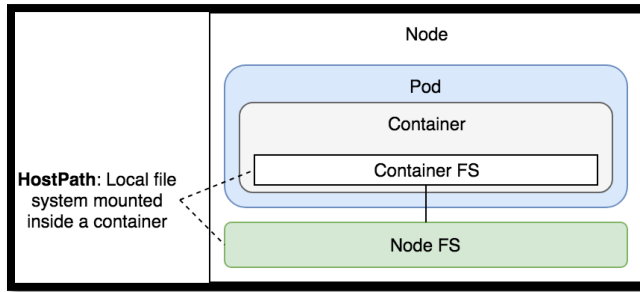


Chapter 7: Using Ingress to Forward Traffic





Chapter 8: Using Volumes to Access Host's File System

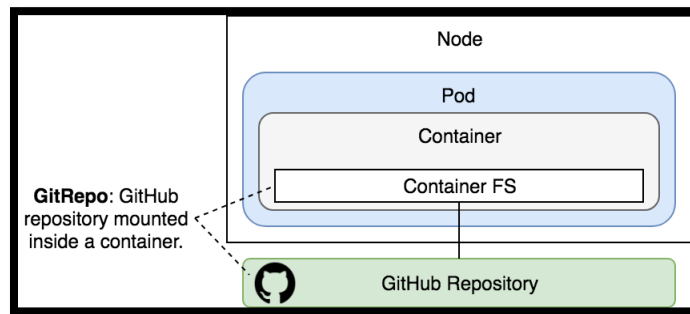


Prometheus Alerts Graph Status - Help

Targets

[prometheus \(1/1 up\)](#)

Endpoint	State	Labels	Last Scrape	Error
http://localhost:9090/prometheus/metrics	UP	Instance="localhost:9090"	4.026s ago	



Jenkins 5 search

ENABLE AUTO REFRESH

add description

All +

S	W	Name ↓	Last Success	Last Failure	Last Duration
🌑	☀️	test	N/A	N/A	N/A

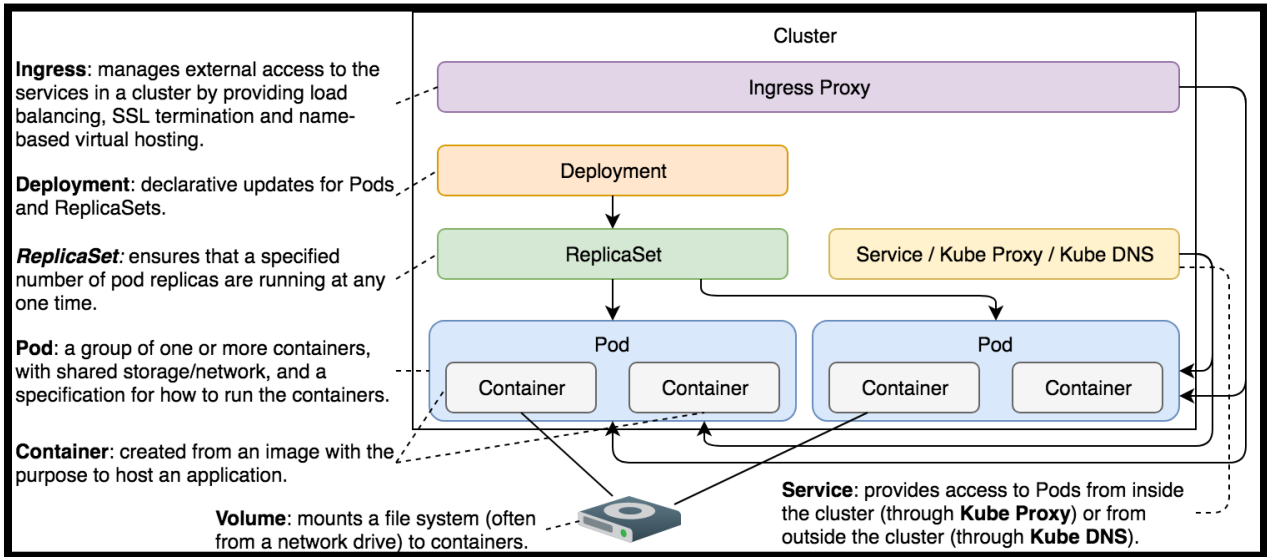
Icon: [S](#) [M](#) [L](#)

Legend [RSS for all](#) [RSS for failures](#) [RSS for just latest builds](#)

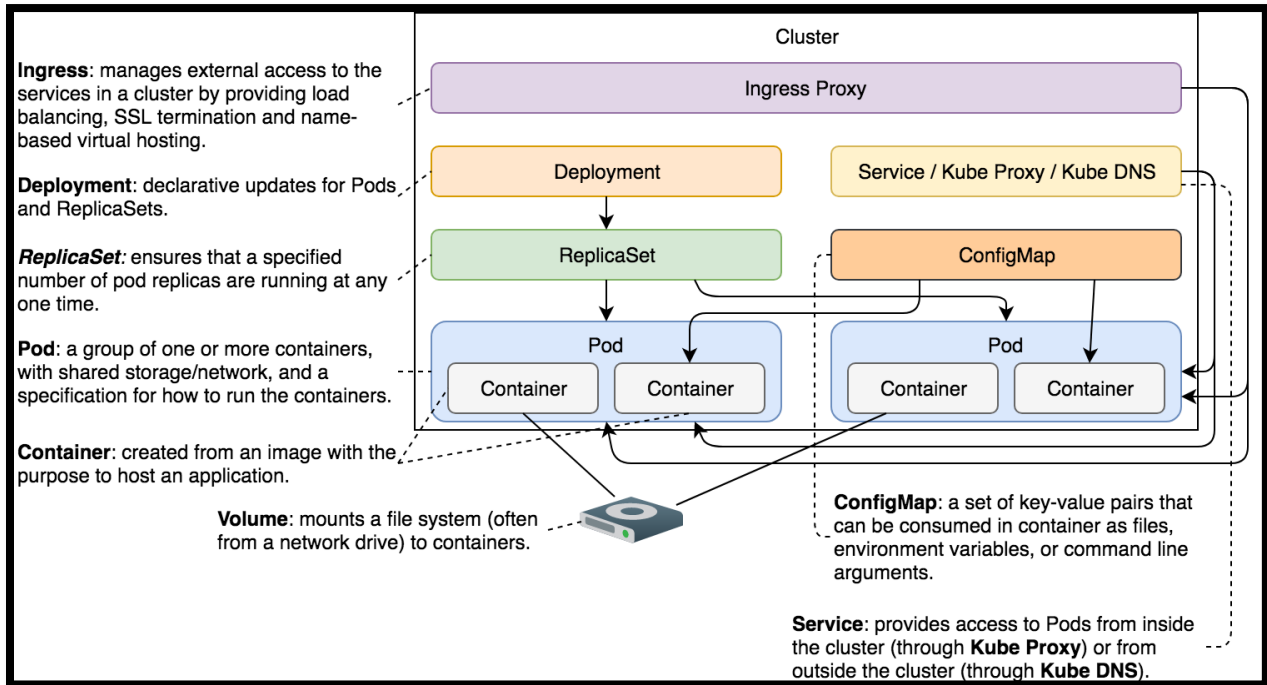
Build Queue -
No builds in the queue.

Build Executor Status -

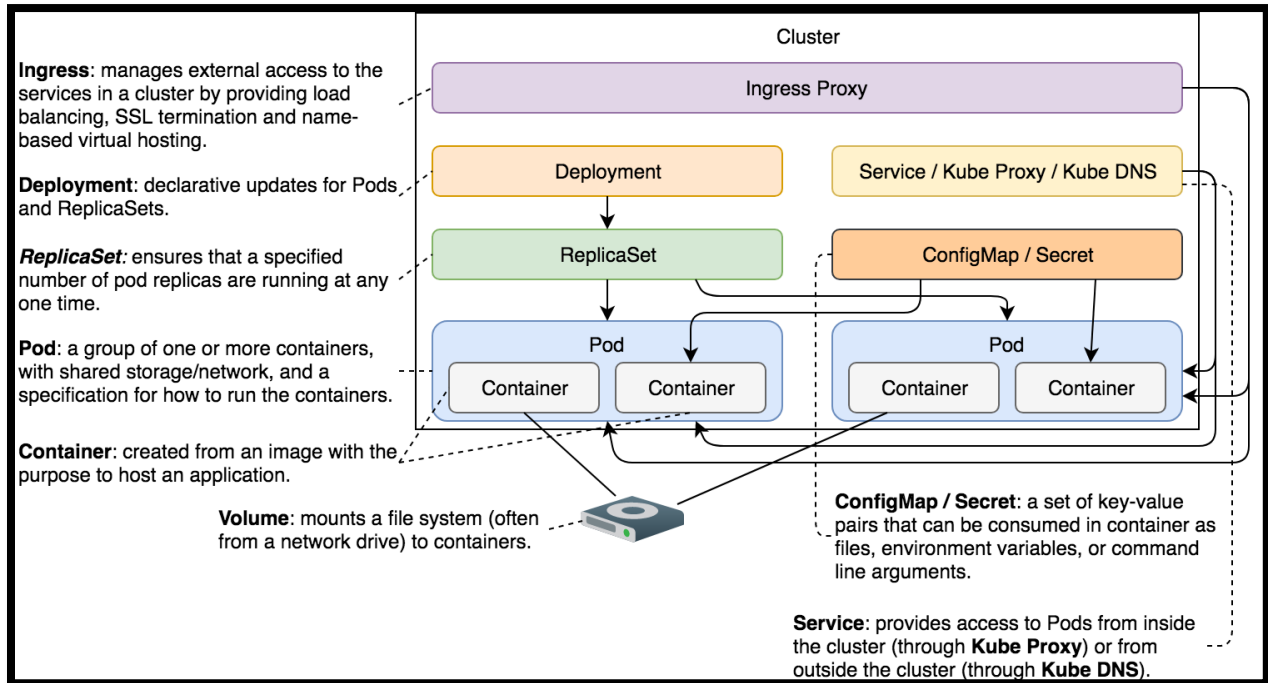
- 1 Idle
- 2 Idle



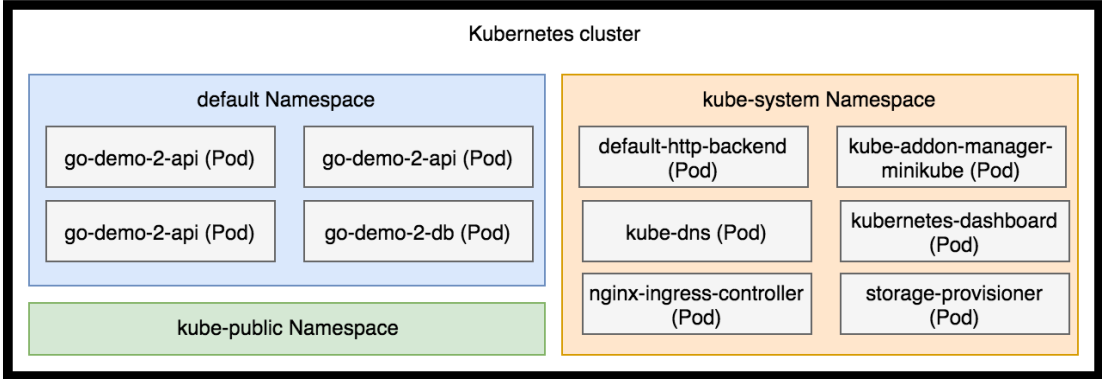
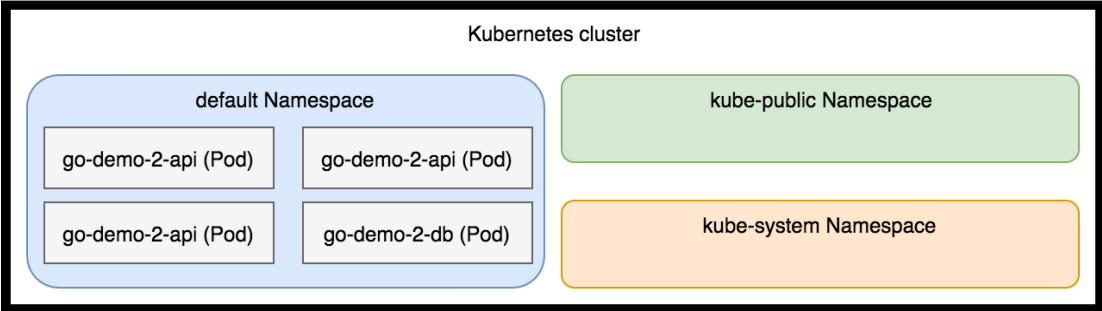
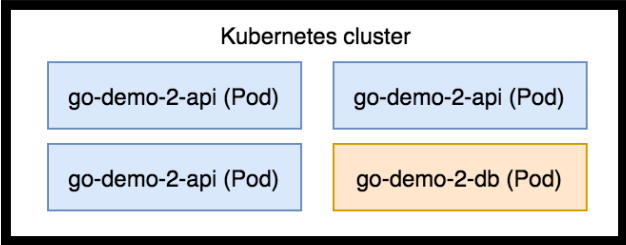
Chapter 9: Using ConfigMaps to Inject Configuration Files



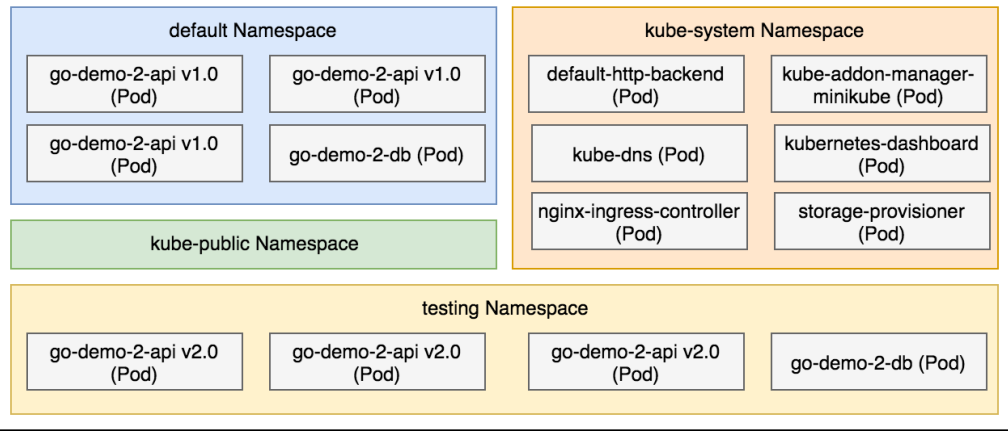
Chapter 10: Using Secrets to Hide Confidential Information



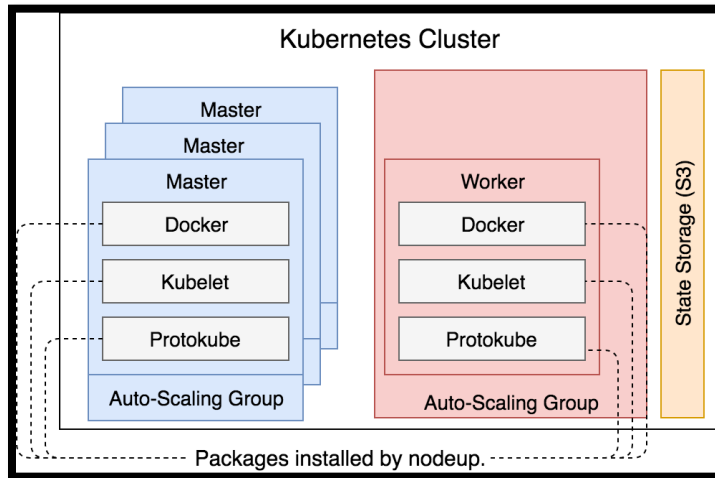
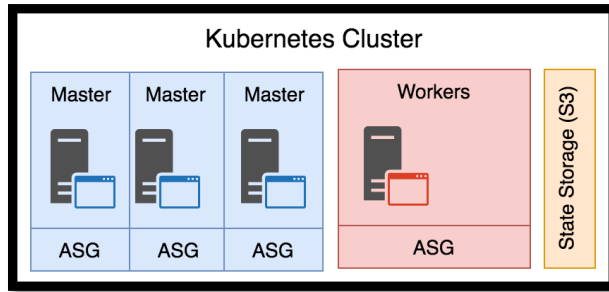
Chapter 11: Dividing a Cluster into Namespaces

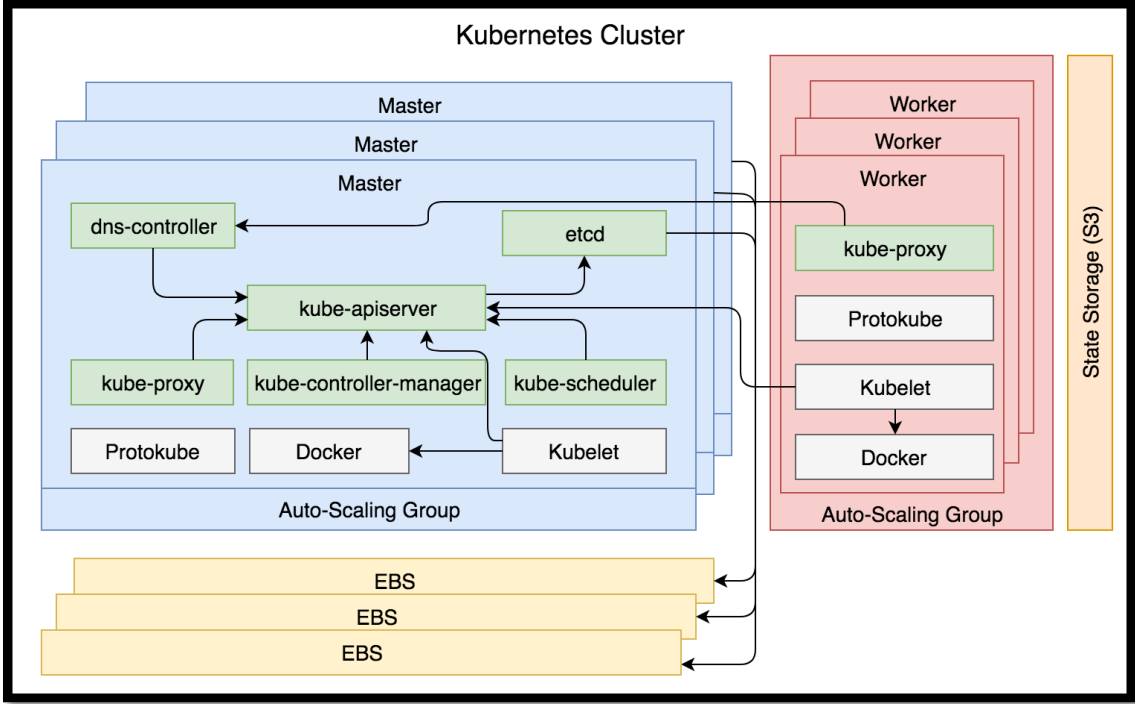


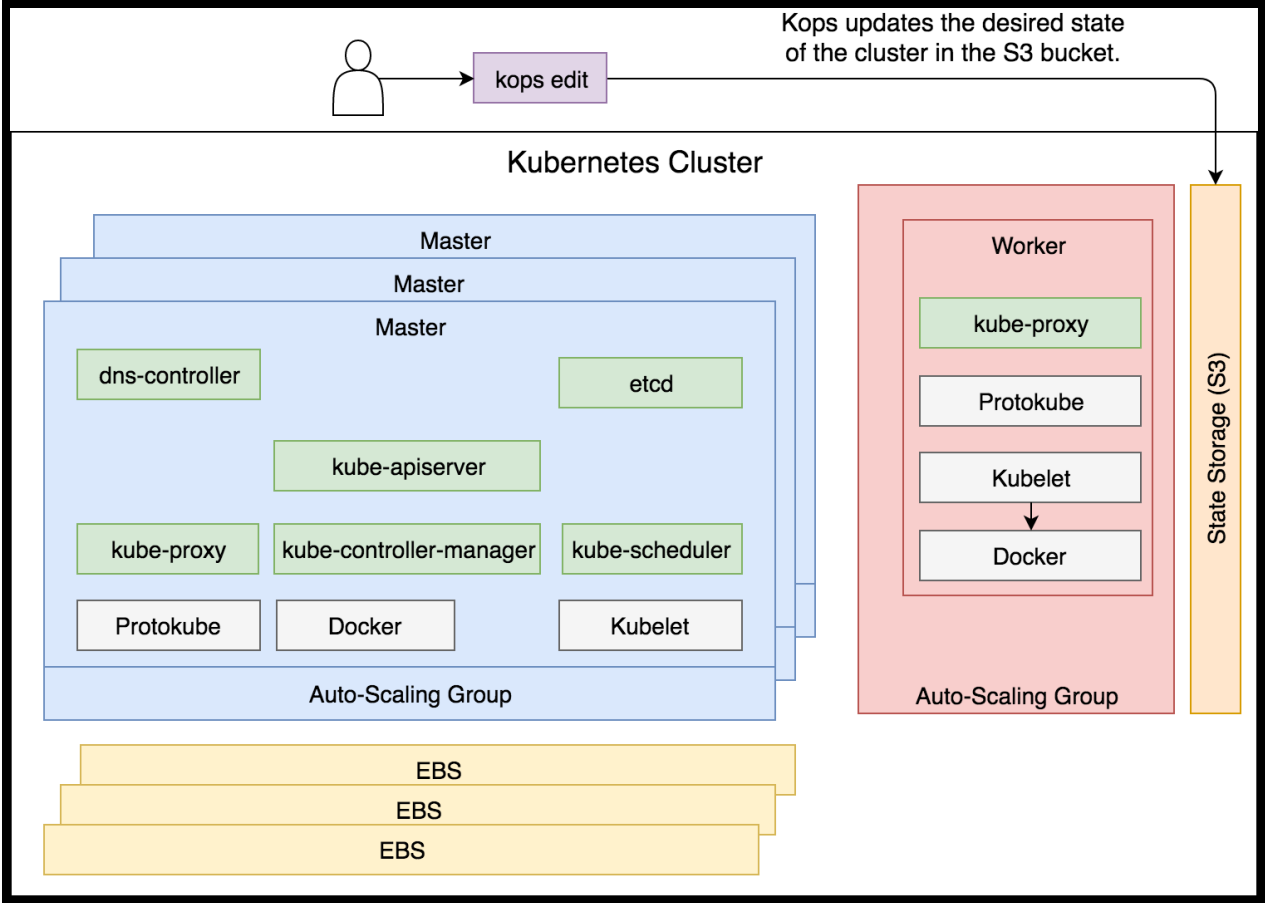
Kubernetes cluster

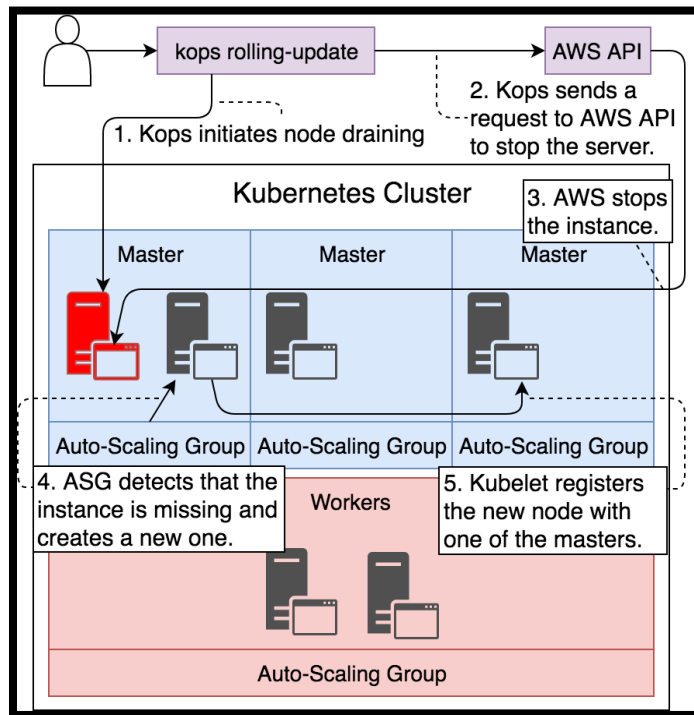
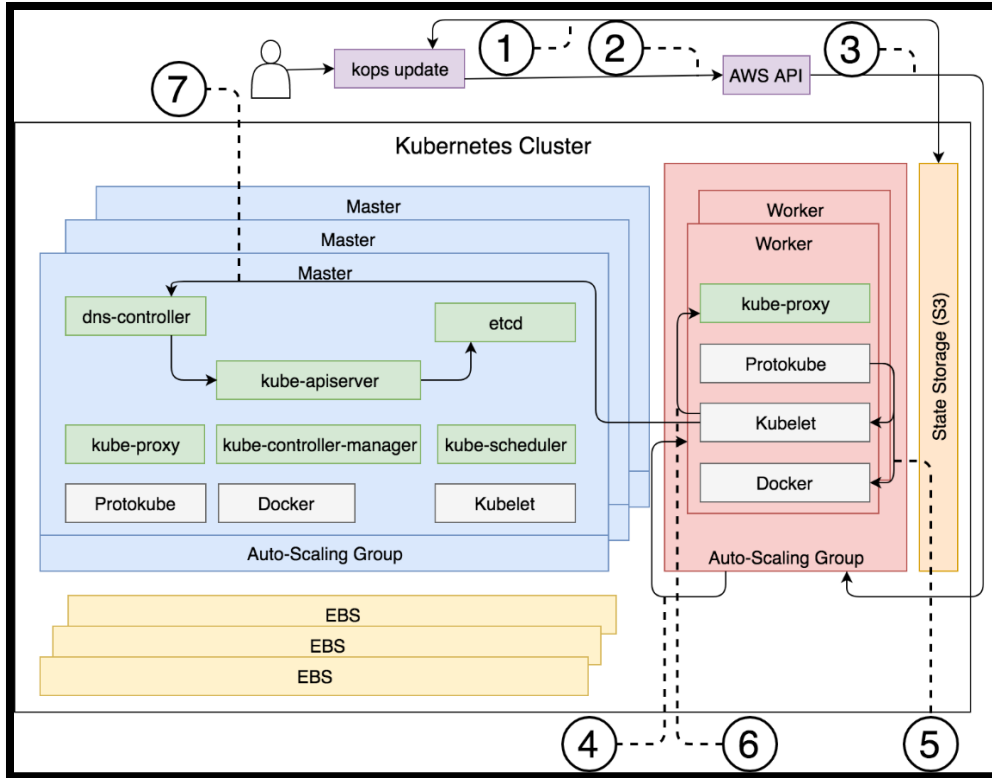


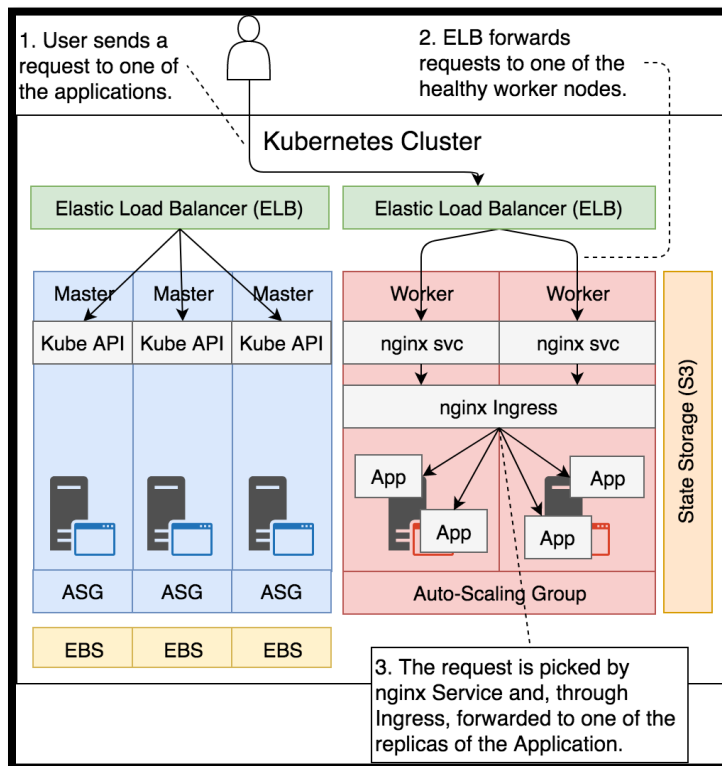
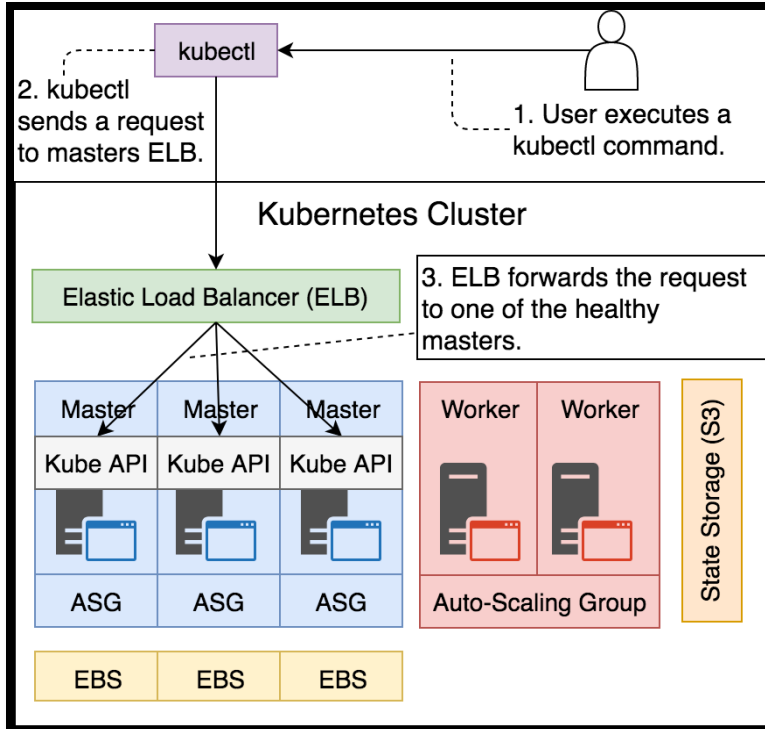
Chapter 14: Creating a Production-Ready Kubernetes Cluster











Update Docker stack

[Select Template](#)

Specify Details

[Options](#)

[Review](#)

Specify Details

Specify parameter values. You can use or change the default parameter values, which are defined in the AWS CloudFormation template. [Learn more.](#)

Stack name

Parameters

Swarm Size

Number of Swarm managers? Number of Swarm manager nodes (1, 3, 5)

Number of Swarm worker nodes? Number of worker nodes in the Swarm (0-1000).

Swarm Properties

Which SSH key to use? Name of an existing EC2 KeyPair to enable SSH access to the instances

Enable daily resource cleanup? Cleans up unused images, containers, networks and volumes

Use Cloudwatch for container logging? Send all Container logs to CloudWatch

Chapter 15: Persisting State

