RANCHER BY SUSE | RANCHER ACADEMY

01 FEBRUARY 2023

Kubernetes Basics

Services and Ingresses

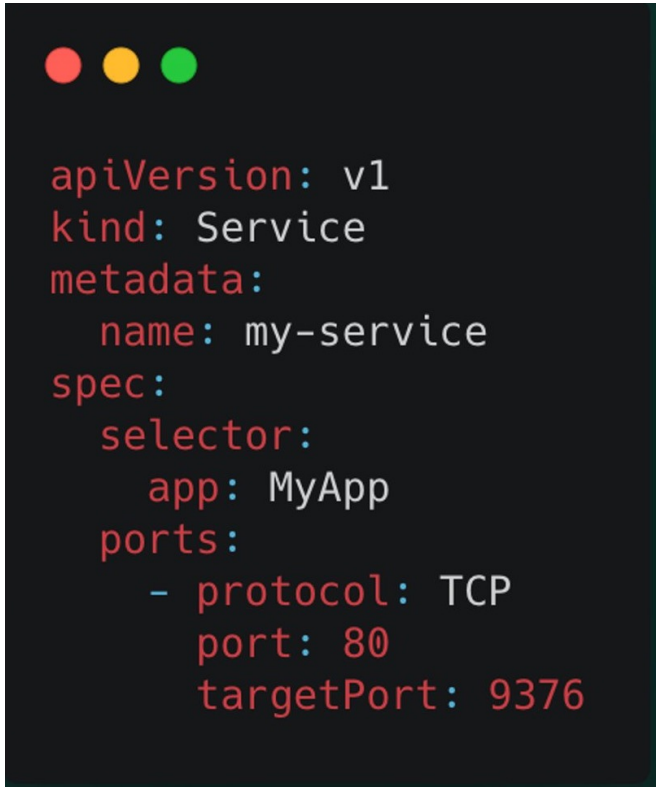
Agenda

1. What is a service?
2. Service Components
3. What is an Ingress?
4. Ingress Components
5. Ingresses and Services Working Together

What is a Service?

What is a Service?

- A method for exposing a network application that is running as one or more Pods in your cluster
- Defines a logical set of endpoints & policies for pod access over a network
- Three types of Services:
  - ClusterIP
  - NodePort
  - LoadBalancer



```yaml
apiVersion: v1
kind: Service
metadata:
  name: my-service
spec:
  selector:
    app: MyApp
  ports:
    - protocol: TCP
      port: 80
      targetPort: 9376
```

Service Components

Service Components

These are the common fields that are needed for a Service:

- **apiVersion** – will be “v1”
- **kind** – will be “Service”
- **metadata** – generally a ”name” is assigned
- **spec** - this will have the specifications of how the Service is exposed
  - **selector** – this is a label selector
  - **ports** – this will contain details about the port & protocol used to expose the service.
    - **protocol** – SCTP, TCP, and UDP are supported
    - **port** – port number that the service is exposed on
    - **targetPort** – port number that the Pod listens on

What is an Ingress?

What is an Ingress?

- API object that manages external access to the services in a cluster, typically HTTP.
- May provide load balancing, SSL termination and name-based virtual hosting.

```yaml
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: minimal-ingress
  annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /
spec:
  rules:
  - http:
      paths:
      - path: /testpath
        pathType: Prefix
        backend:
          service:
            name: test
            port:
              number: 80
```

Ingress Components

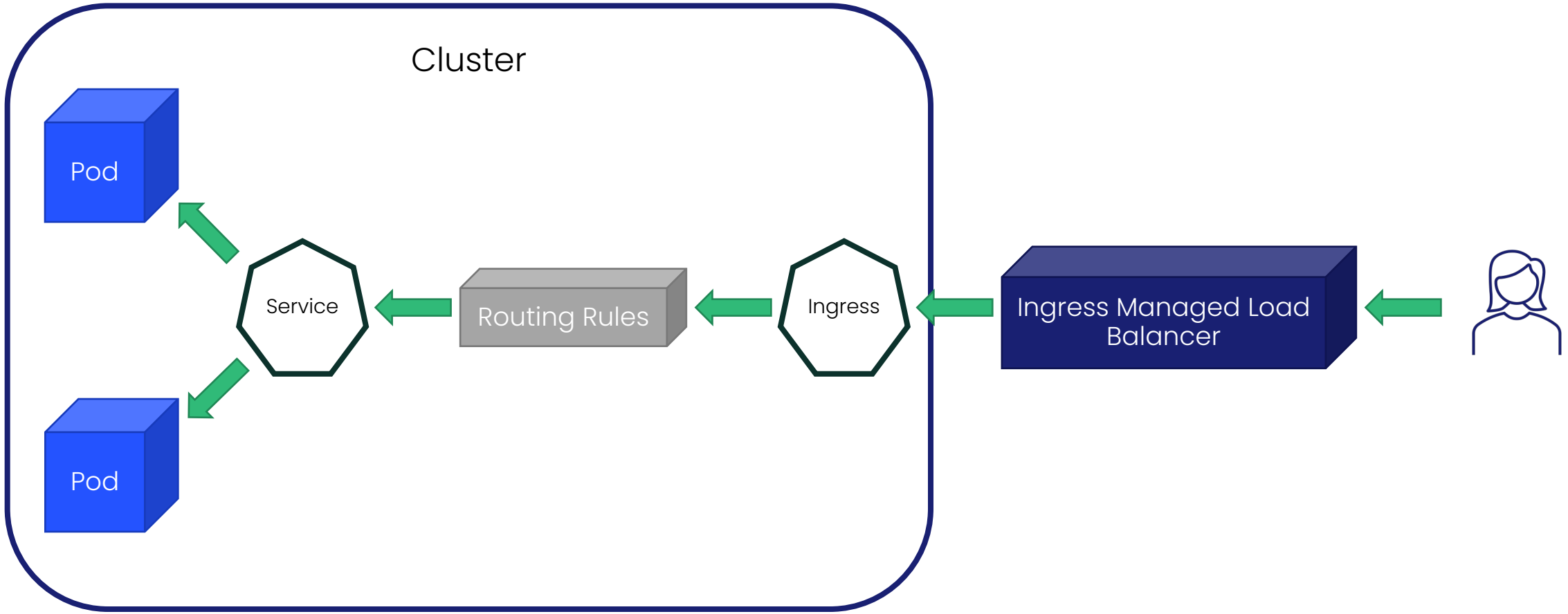
Ingress Components

These are the common fields that are needed for an Ingress:

- **apiVersion** – will be “v1”
- **kind** – will be “Ingress”
- **metadata** – generally a ”name” is assigned
  - **annotations**: configure options based on the Ingress controller. For example, rewrite-target annotation.
- **spec** - this will have the information to configure a load balancer/proxy server
- **rules** – specifies the routing rule used
  - **paths** – A list of paths, each of which has an associated backend defined with a **service.name** and a **service.port.name** or **service.port.number**
  - **backend** – Combination of **Service & port names** or a **custom resource backend** defined by a CRD.

Ingresses and Services Working Together

Ingresses and Services Working Together



RANCHER BY SUSE | RANCHER ACADEMY

Thank you

For more information, contact SUSE at:

+1 800 796 3700 (U.S./Canada)

Frankenstrasse 146

90461 Nürnberg

www.suse.com