

# Configuring and managing application access with Ingress



THAO LUONG  
09/2020



# Content

---

- ❑ Ingress overview and architecture
  - ❑ Ingress
  - ❑ Ingress controller
- ❑ Common use cases



# Ingress Architecture



Ingress Object



Ingress Controller



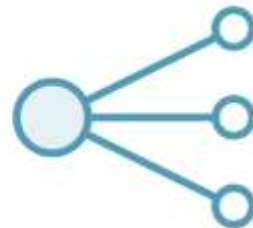
# Ingress Overview



API Object



Defines rules for external access to Services



Load balancing to Endpoints



Name-based virtual hosts



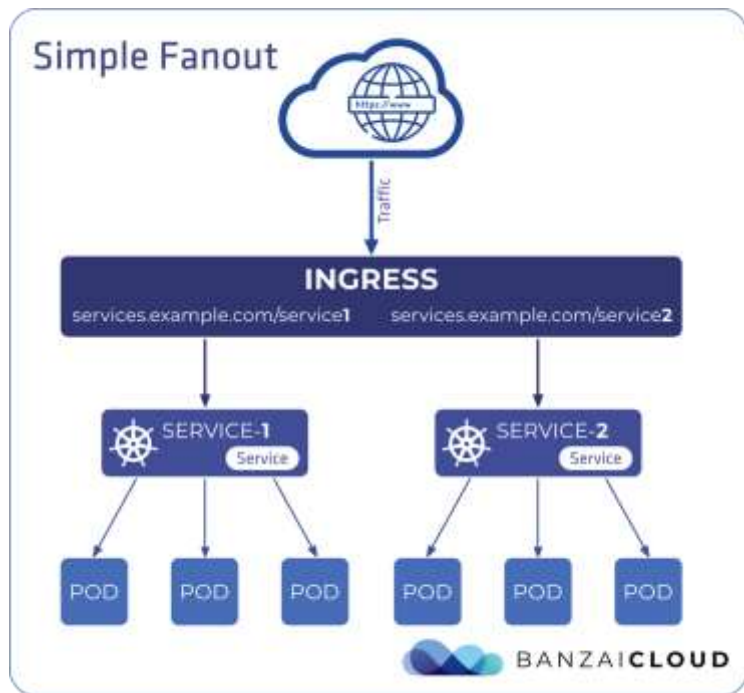
Path-based routing



TLS termination



# Fanout vs name-based virtual hosts





# Ingress Controller



Implements the rules defined in the Ingress Resource

Many types of Ingress Controllers

Pods in a cluster - nginx

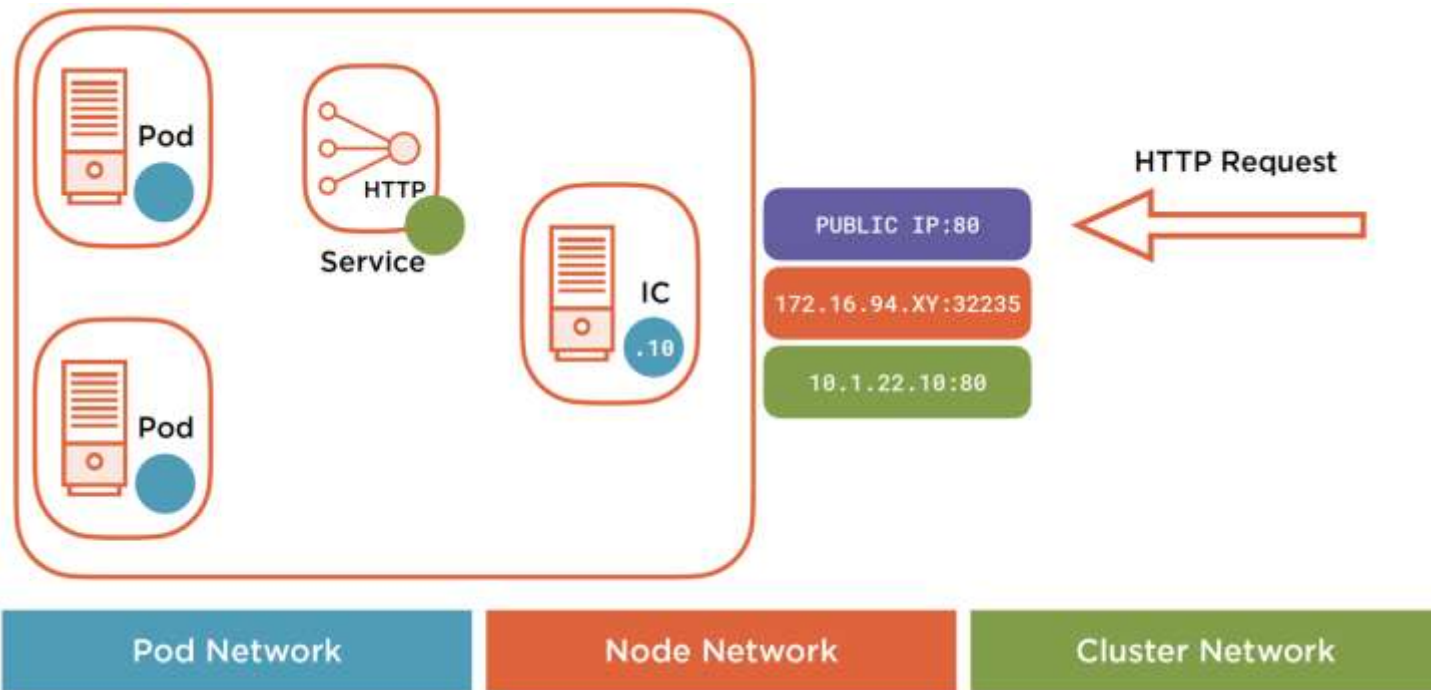
Hardware external to the cluster - Citrix and F5

Cloud Controllers - AppGW, Google Load Balancer and AWS ALB Ingress

Ingress controllers have a defined spec



# Exposing a single service with Ingress





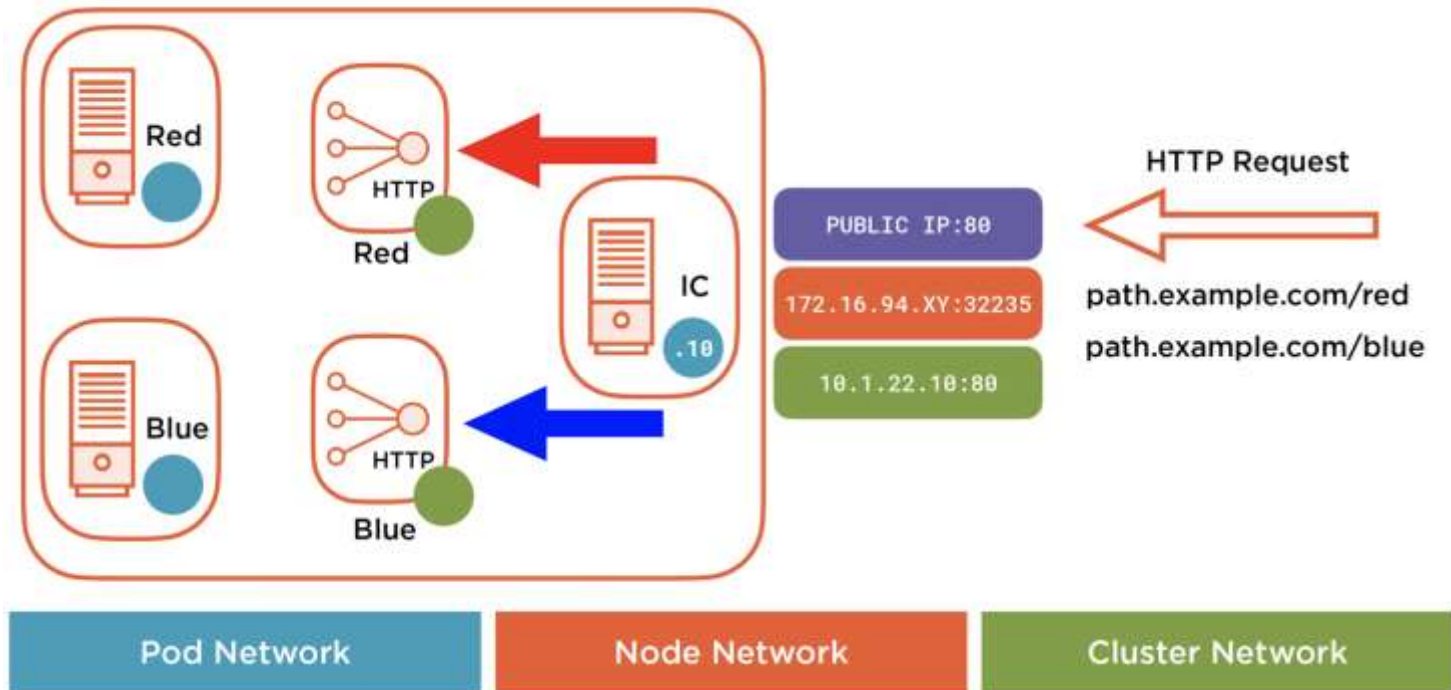
## Exposing a single service with Ingress

```
apiVersion: networking.k8s.io/v1beta1
kind: Ingress
metadata:
  name: ingress-single
spec:
  backend:
    serviceName: hello-world-service-single
    servicePort: 80
```





# Exposing multiple services with Ingress



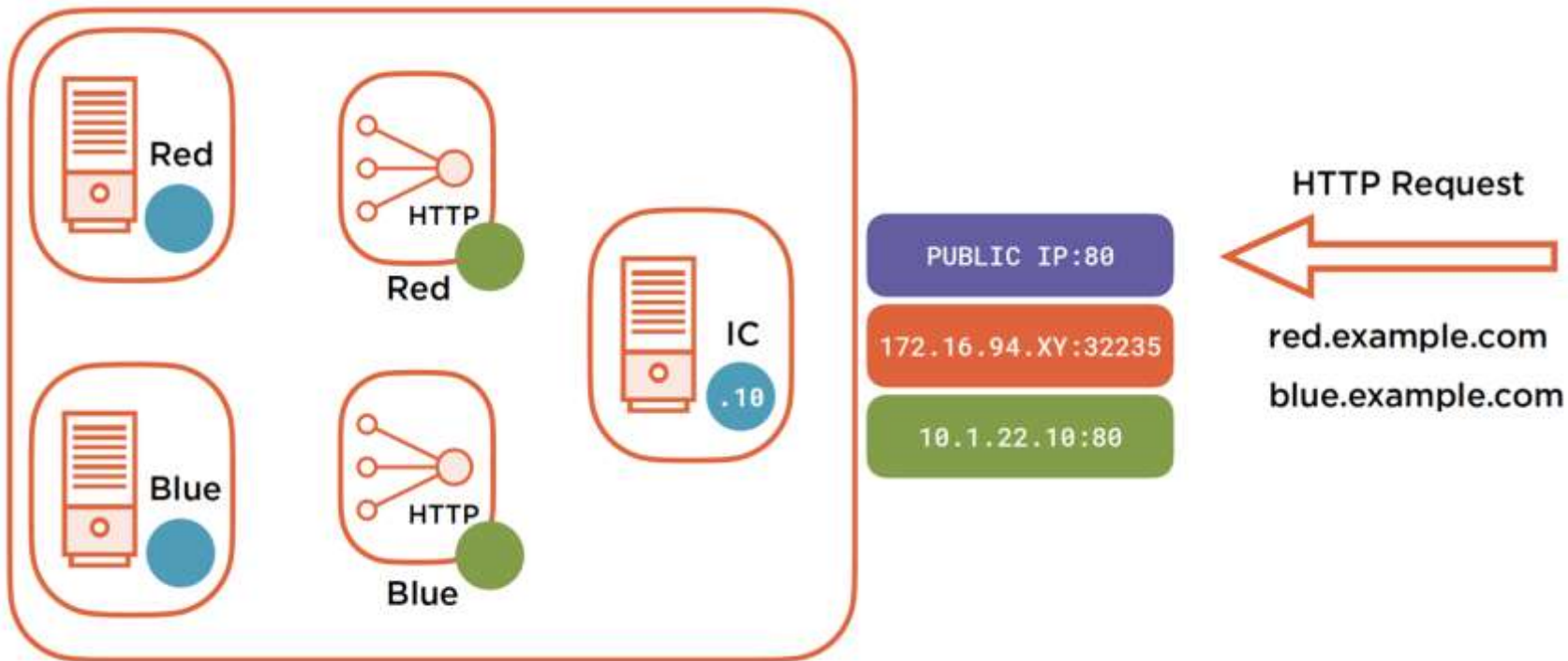


# Exposing multiple services with Ingress

```
spec:
  rules:
    - host: path.example.com
      http:
        paths:
          - path: /red
            backend:
              serviceName: hello-world-service-red
              servicePort: 4242
          - path: /blue
            backend:
              serviceName: hello-world-service-blue
              servicePort: 4343
  backend:
    serviceName: hello-world-service-single
    servicePort: 80
```



# Name-based virtual hosts with ingress





# Name-based virtual hosts with ingress

```
spec:
  rules:
    - host: red.example.com
      http:
        paths:
          - backend:
              serviceName: hello-world-service-red
              servicePort: 4242
    - host: blue.example.com
      http:
        paths:
          - backend:
              serviceName: hello-world-service-blue
              servicePort: 4343
```



# Using TLS certificates for HTTPS Ingress

```
spec:
  tls:
    - hosts:
      - tls.example.com
      secretName: tls-secret
  rules:
    - host: tls.example.com
      http:
        paths:
          backend:
            serviceName: hello-world-service-single
            servicePort: 80
```

