# Ingress in Kubernetes

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

## How Ingress Works

Ingress exposes HTTP and HTTPS routes from outside the cluster to services within the cluster. Traffic routing is controlled by rules defined on the Ingress resource.

## Components of Ingress

1. Ingress Resource: A set of rules and configurations applied to the cluster to manage external access to the services.

2. Ingress Controller: An application that watches the Ingress resources, applies the rules, and manages the actual traffic routing. Popular controllers include NGINX, HAProxy, and Traefik.

## Configuration of Ingress

An example of an Ingress resource configuration is shown below.

apiVersion: networking.k8s.io/v1  
kind: Ingress  
metadata:  
 name: example-ingress  
 annotations:  
 nginx.ingress.kubernetes.io/rewrite-target: /  
spec:  
 rules:  
 - host: www.example.com  
 http:  
 paths:  
 - path: /service1  
 pathType: Prefix  
 backend:  
 service:  
 name: service1  
 port:  
 number: 80  
 - path: /service2  
 pathType: Prefix  
 backend:  
 service:  
 name: service2  
 port:  
 number: 80

## Benefits of Using Ingress

1. Centralized Management: Manage all external traffic rules from a single resource.

2. Cost Efficiency: Reduces the number of load balancers needed, as multiple services can be exposed through a single ingress point.

3. Enhanced Security: Supports SSL/TLS termination, protecting internal communications.