

Install the nginx ingress controller

Ref - <https://docs.nginx.com/nginx-ingress-controller/installation/installing-nic/installation-with-manifests/>

Using helm

- Install helm

Ref - <https://helm.sh/>

Installation - <https://helm.sh/docs/intro/install/>

- Add the **ingress-nginx** chart

```
helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx  
helm repo update  
helm install nginx-ingress ingress-nginx/ingress-nginx --namespace ingress-
```

- Check if you have pods running in the

```
kubectl get pods -n ingress-nginx
```

Default loadbalancer service

You will notice that if you use **helm** to install the nginx-ingress-controller, it creates a **Loadbalancer** service for you

```
kubectl get services --all-namespaces
```

NAMESPACE	NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
default	kubernetes	ClusterIP	10.96.0.1	<none>	443/TCP	82m
ingress-nginx	nginx-ingress-nginx-controller	LoadBalancer	10.104.239.164	65.20.84.86	80:31481/TCP,443:31751/TCP	9m41s
ingress-nginx	nginx-ingress-nginx-controller-admission	ClusterIP	10.109.41.115	<none>	443/TCP	9m41s
kube-system	kube-dns	ClusterIP	10.96.0.10	<none>	53/UDP,53/TCP,9153/TCP	82m

This routes all the traffic to an external



```
kubectl get pods -n ingress-nginx
```



This means the first part of our **ingress deployment** is already created

What is Ingress?

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.

Here is a simple example where an Ingress sends all its traffic to one Service:

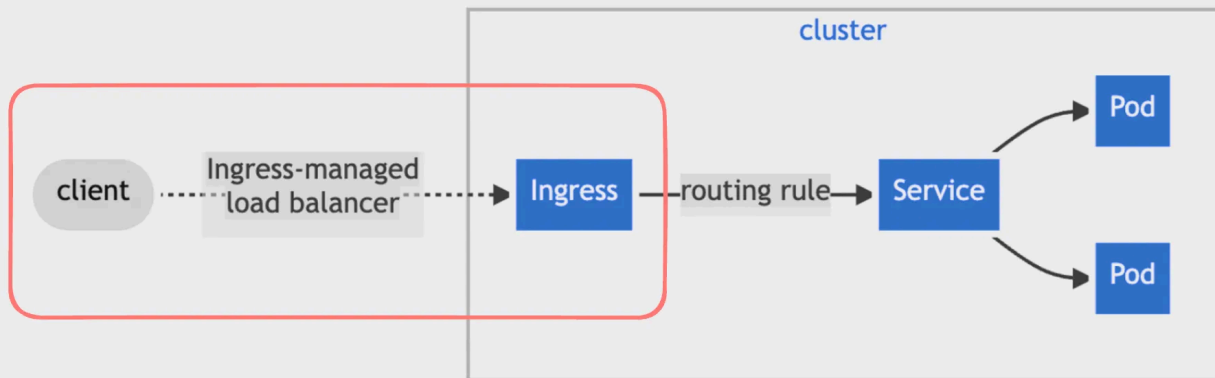


Figure. Ingress