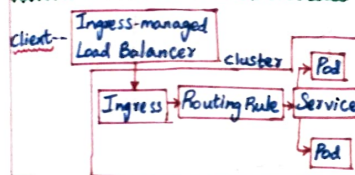


## 1) Ingress & Egress:-

\* Ingress: It is incoming traffic to the Pod.

\* Exposes HTTP & HTTPS routes from outside the cluster to Services within the cluster.

Ingress sends all its traffic to one:-



\* Egress: It is outgoing traffic from the pod.

(i) Restricting egress traffic:

\* It's a common security requirement and best practice to restrict outgoing connect from the cluster.

(ii) Outgoing (NAT) Behaviour:

\* Network Address Translation is the process of mapping an IP address in a packet to different IP address as the packet passes through the device performing the (NAT).

(iii) Egress Gateways:

\* Another approach to Kubernetes egress is to route all outbound connections via one or more gateway. The gateways SNAT (Source Network Address Translation) the connections so the external service being connected to sees the connection coming from egress gateway.

## 2) Kubernetes Deployment Yaml:-

\* It specifies the configuration for a Deployment object - this is a Kubernetes objects that can create and update a set of identical pods. Each pod runs specific containers, which are defined in the spec.template field of the Yaml configuration.

\* A Kubernetes Deployment Yaml specifies the configuration for a Deployment object - this is a Kubernetes object that can create and update a set of identical pods.

### 3) DaemonSet:

\* It deploys a pod on all cluster nodes or a certain subset of nodes.

### 4) StatefulSet:

\* Used for stateful applications. Similar to a Deployment, but each pod is unique and has persistent identifier.

### 5) Expose Router:

\* It navigates to your router's configuration page by typing the router's IP address into your browser. Find a settings tab for ports, or port forwarding. Where indicated, input the number of port you want to open. Where indicated input the static IP address of the device you want the port to forward to router.

### 6) Services in Kubernetes:

(i) Cluster IP: Expose the service on a cluster-internal IP.

\* Makes the service only reachable from within the cluster. This is default service type.

(ii) Node Port: Expose the service on each node's IP at a static port.

\* A cluster IP service to which node port service will route, is automatically created.

(iii) Load Balancer: Expose the service externally using a cloud provider's load balancer.

\* Services, to which the external load balancer will route, are automatically created.

(iv) External Name: Maps the service to the contents of the External Name field by returning a (CNAME) record with its value.

\* No proxying of any kind is set-up.

## 7) Types of Load Balancer:

### (i) Internal Load Balancer:

It automatically balances load and allocates the pods with required configuration.

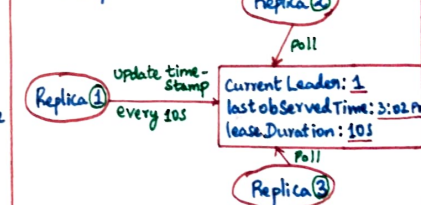
(ii) External Load Balancer: It directs the traffic from the external load to the backend pods.

### (8) Polling:

\* It is a library for computing the status of Kubernetes resources based on polling of resource state from a cluster. It can keep polling until either some condition is met, or until it is cancelled through the provided context.

### 9) Election in Kubernetes:

\* It begins with the creation of a lock object, where the leader updates the current timestamp at regular intervals as a way of informing other replicas regarding its leadership.



### (10) Taint & Tolerance in Kubernetes:

(i) Taint: It is opposite - they allow a node to repel a set of pods.

(ii) Tolerance: It is applied pods & it allows the scheduler to schedule pods with matching taints.

### (ii) Node affinity:

\* It is a property of pods that attracts them to a set of nodes (either as a preference or a hard requirement).

## (12) Node Scheduler:

\* It is a control plane process which assigns pods to nodes. It determines which nodes are valid placements for each pod in the scheduling queue according to the constraints and available resources. Scheduler then ranks each valid node and binds the pod to a suitable node.

### (13) Node Selector:

\* The simplest recommended form of node selection constraint. You can add the node selector field to your pod specification and specify the node labels you want the target node to have. Kubernetes only schedules the pod into nodes that have each of the labels you specify.

### (14) Node Anti-Affinity:

\* It also called inter-pod affinity allows you to constrain pods against labels on other pods.

### (15) Router in Kubernetes:

\* It is a turnkey solution for K8S networking with aim to provide operational simplicity. Networking is hard as such. In typical K8S cluster you would need to install multiple network components for various functionality.

### (16) Create Secret in K8S:

\* Secret is an object that contains a small amount of sensitive data such as a password, a token, or a key.

- Use kubectl.
- Use a configuration file.
- Use the kustomize tool.

### (17) Workloads in K8S:

\* It is application running in K8S. It is a single component or several that work together, on K8S you run it inside a set of pods.