Introducción – conceptos asociados

Nodos o Nodes:

* Son las computadoras que serán usadas por kubernetes.

Cluster

* Conjunto de nodos o computadoras que serán utilizadas para correr una aplicación.

Pods o capsulas:

* Son un grupo de uno o más contenedores que comparten almacenamiento y configuración. Son creados por los deployments.

Deployments o despliegues:

* Son utilizados para realizar actualizaciones en los Pods. Nos permite crear nuevos pods, cambiar la versión de la imagen que se esté utilizando y de ser necesarios, recrear el estado anterior.

Services o servicios:

* Son un conjunto de políticas que definen el acceso a los contenedores. Permiten la integración de los contenedores con otros servicios.

Replication controller:

* Controla cuantas copias idénticas de un pod deben estar ejecutándose en un cluster.

Stateless:

* Las aplicaciones stateless o sin estado, no manejan información o mejor dicho, no almacenan información. Estos son intermediarios para el manejo de información. Por ejemplo, una aplicación de NODE que solo se dedique a pedir información de una BD, modificarla y enviarla hacia el cliente; sin que la aplicación de NODE no guarde nada de dicha información o la utilice para posteriores peticiones.
* Otra forma de verlo es que la aplicación no guarda ninguna información de las peticiones anteriores, como son las sesiones.

Statefull:

* Guarda la información de forma general. Puede ser de forma local como las BD o sistemas que almacenen datos en tiempo real. Un ejemplo sería un carro de compras de una aplicación, que guarda los datos del carro en tiempo real; o sea, cuando añades un nuevo producto, se mantienen los que anteriormente estaban agregados.
* El almacenamiento de los datos tiene que ser administrado por la aplicación.

\*\* Un ejemplo a mayor escala, serían las aplicaciones web modernas. Donde una aplicación “**backend**” (**stateless**) realiza todas las operaciones, pero necesita de una base de datos (**statefull**) para lograr la persistencia de datos.

StatefulSet:

* Provee mecanismos para el despliegue de aplicaciones que posean estado (necesitan de almacenamiento). Además, cada vez que el mismo contenedor es creado, se le asigna el mismo nombre de dominio (DNS) y el mismo almacenamiento. Estos no crean ReplicaSet, por lo que es imposible realizar un rollback.

ReplicaSet:

* Provee mecanismos para el despliegue de aplicaciones que no poseen estado.

Headless Service:

* Son servicios que se asocian al StatefulSet para permitir el acceso mediante IP y no a través del nombre (DNS).

Control plane o plano de control:

* Colección de procesos que controlan los nodos de Kubernetes.

PV (Persistent Volumes) o PVC (Persistent Volume Claim)

* Permiten la administración (manual/dinámica) del almacenamiento de los nodos de kubernetes.

Kubelet:

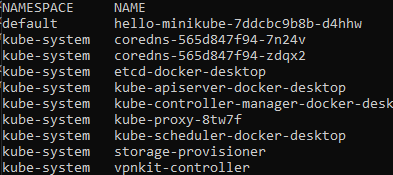
* Servicio que se ejecuta en cada nodo, permite la aplicación del manifiesto (configuración) del contenedor y se asegura que el o los contenedores estén inicializados y ejecutándose.

Kubectl:

* Es el CLI (línea de comandos) de configuración de Kubernetes.

Namespace

* Kubernetes agrupa los pods creados dentro del “**cluster**” mediante los nombres de dominio o “**namespace**”:



* + Esto nos permite aplicar reglas a dominios en específico.

ETCD

* Componente central de **kubernetes** que guarda los estados de todos los componentes creados dentro del **cluster**.

INGRESS

* Es una API que permite manejar el trafico desde internet hacia los **cluster** de **kubernentes**. Este permite exponer los **services** dentro de un **cluster** y decidir a quien va dirigida la petición.

INGRESS CONTROLLER

* Funciona como un proxy reverso y balanceador de carga. Este es el encargado de manejar todas las peticiones que vengan de internet y enviárselos al **INGRESS**, esto quiere decir que trabajan en conjunto. Para que posteriormente el **INGRESS** pueda enrutarlo al **POD** correspondiente. No confundir con el tipo de servicio **loadBalancer**.