

الـ Kubernetes نفسه يعمل في داخل الـ Containers

The cluster meaning

master node and worker nodes

Controller manager → trace the run time of cluster
↳ and trace the state of the cluster

Scheduler → يتبع على الاتصال أو الأوامر ... من الـ Master
↳ hardware Resources؟ من صغيف؟ management.

The pod meaning

Container

ETCD → Data store

↳ take a backup of the cluster, you can restore the data by it.

* مثلاً ان يقدر يتواصل مع الـ worker node على Virtual network غير الـ Physical network

The virtual Network meaning

communication by or on software

↳ Like Kubelet with scheduler and controller manager in the cluster.
↳ made on top of physical network to work as one virtual machine

The control plane don't need high Requirement

but you should have a backup version of it, cause if you loose it, you loose the program. (cluster)

while if you loose a worker machine, it o'kay!

The main component of Kubernetes

Node → is the virtual machine or physical machine
as the worker or the master.

POD → is the container... Abstraction layer above the container

هو الي بيحل الcontainers ، هاد الحجي مشان تقدر الKubernetes

تقدر تعامل مع containers ثابته من بزاج افرواخ...

one Application → one pod

Two Application → Two Pods

pod → container → APPS

كل وحدة من هود ال pod لها ip address

مشان يعجل Communication بين بعضهم البعض...

pod's → سرعات اتوال

لما تنقلها من Node ل Node فانت عليها انشاءت وحدة جديدة وهيك ما تاكل...

المشكلة انه لما تيجي الجديدة فيج بيمر لها ip address جديد... الخ في ال services

Service → it has permanent ip address, and it not Related to the pod
we designed it for the pod.

ازامات ال pod وعلنا pod جديد ، بيوصلنا في نفس ال service

service is always here, don't die.

External service

داخلي by default

External service

الخارجي

فانك ال nodes على cluster وداخلي ال cluster

Engress → The main goal of it, to reach the cluster from the outside

cluster ال باب على

ال service وال ingress مش بي بسجل بالموصل ، كان بملو

load balancer → The load are distributed on the pods.

اذا تغير ال container ، أو على ال image تبع ال docker بيك تهل Rebuild للتبقي تبعك...

ليش

افرض نجربنا البيولد والبال مع ال data بي ، بنا نوع تغير container تبعنا **سوال الحل ؟**

الحل انما تكون هاي التعلقات أو ال Variables ما تكون مخزنة داخل ال image ...

ConfigMap → The main goal of it is to store some configuration, like the Address of database (URI), username, Password of DB

Secret → as the configmap
the main different is that the secret can be encrypted
while the configmap is clear text.

So, username and password of DB

you can store them into secret...

The secret file by default is unencrypted file.

you have to encrypt it manually.

and you can call it like a variables.

it not easy to manage the database by kubernetes cluster

Persistent Volume → حفظ البيانات بشكل دائم باستخدام الـ volume

The kubernetes don't support a synchronization (persistent)

So manage your Apps on it, but not the database.

→ for stateless
APP'S

Deployment → define the blueprint of the pod's

منظومة، يحدد عدد الـ pod's أو عدد النسخ من التطبيق تبعي

ويجعل deployment ليعطي الـ deployment ويتبع نسخ بالعدد التي كتابته.

So you create deployments not pod's

So you write your desire state in the deployment file (number of replicant maybe)

and deploy it (Run it).

والحل في الواقع انه الـ system يفرض عليك على المتطلبات التي انت مطلوبا في الـ deployment

وفي حال وريدت طيفت وكذا هو يقوم باشاء وريدت جوية بدلية عنها...

don't create a replication for database by deployment file, be careful!

Stateful Application → use stateful sets
هذا الـ deployment