

Api server:  It is the central touch point that is accessed by all users, automation, and components in the **Kubernetes** cluster. The **API server** implements a RESTful **API** over HTTP, performs all **API** operations, and is responsible for storing **API** objects into a persistent storage backend.

Controller Manager: daemon that embeds the core **control** loops shipped with **Kubernetes**. ... In **Kubernetes**, a **controller** is a **control** loop that watches the shared state of the cluster through the apiserver and makes changes attempting to move the current state towards the desired state

Scheduler: open source **Kubernetes** container orchestration platform that controls performance, capacity and availability through policies and topology awareness. ... The **Kubernetes scheduler** attempts to match each Pod created by **Kubernetes** to a suitable set of IT resources on a Node.

CAdvusor: **cAdvisor** is an open source container resource usage collector. It is purpose built for containers and supports Docker containers natively. Unlike most elements within **Kubernetes** that operate at the Pod level, **cAdvisor** operates per node

Etcd:  its configuration data, its state, and its metadata. **Kubernetes** is a distributed system, so it needs a distributed data store like **etcd**. **etcd** lets any of the nodes in the **Kubernetes** cluster read and write data.

Kubelet: The unit of execution that **Kubernetes** works with is the pod. A pod is a collection of containers that share some resources: they have a single IP, and can share volumes

CAdvusor: An open source container resource usage collector. It is purpose built for containers and supports Docker containers natively. Unlike most elements within **Kubernetes** that operate at the Pod level, **cAdvisor** operates per node

Kube-pxoxy: he **Kubernetes** network **proxy** runs on each node. This reflects services as defined in the **Kubernetes** API on each node and can do simple TCP, UDP, and SCTP stream forwarding or round robin TCP, UDP, and SCTP forwarding across a set of backends. ... The user must create a service with the apiserver API to configure the **proxy**

Pod: group of containers that are deployed together on the same host. If you frequently deploy single containers, you can generally replace the word "**pod**" with "container" and accurately understand the concept.

All the command in Kubernetes

1000 mile