## WHAT IS KUBERNETES?

Kubernetes (also known as k8s or "kube") is an open source container orchestration platform that automates many of the manual processes involved in deploying, managing, and scaling containerized applications.

What are Kubernetes clusters?

You can cluster together groups of hosts running Linux® containers, and Kubernetes helps you easily and efficiently manage those clusters.

Kubernetes clusters can span hosts across on-premise, public, private, or hybrid clouds. For this reason, Kubernetes is an ideal platform for hosting cloud-native applications that require rapid scaling
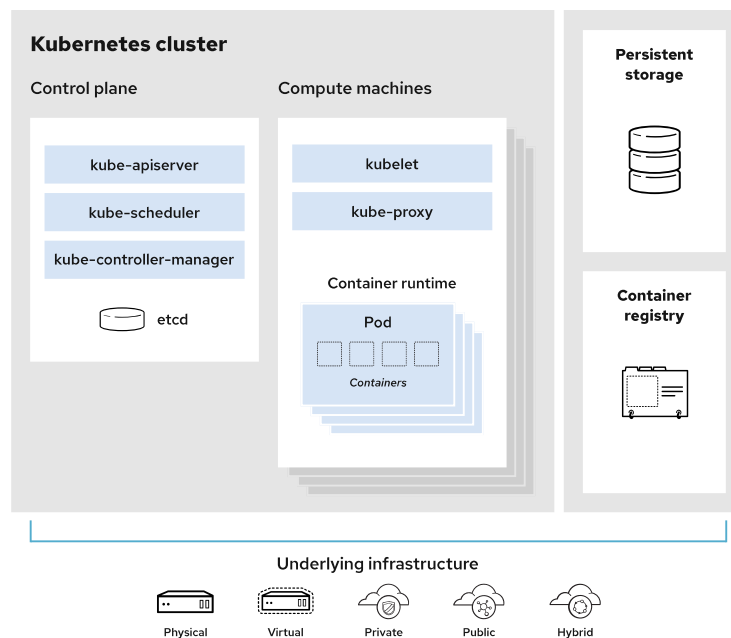
Self-Managed Kubernetes Cluster

Self Managed K8s custer means that the admin setup the overall connection and the layout of the cluster as per the need the main benefit of using a self-managed cluster is that ,You have control over the management layer on a self-managed Kubernetes Cluster . Moreover, you get complete access to the cluster master and all other related management components when you deploy your own cluster using kubeadm, kubespray, Kops or even the hard way.

This also gives configuration flexibility, allowing you to setup the cluster and nodes however you wish, rather than having to deal with the configuration options allowed by a managed service.

By using a Self-Managed Kubernetes Cluster you have many pros and cons.

As a pro, on a self-managed Kubernetes Cluster, you have control over the management layer. Fully managed Kubernetes services in the cloud don't allow you to configure the cluster master, because that component is handled by the managed service. When you deploy your own cluster using kubeadm, kubespray or even doing it the hard way, you have full access to the cluster master all the other related management components.

This also adds flexibility of configuration where you can configure the cluster and the nodes the way you want, instead of wrestling with the configuration options supported by a managed service.

You also have more control over the deployment and management of your cluster. For example, you can deploy multiple node pools or choose to have different instance types for different nodes. These options aren't available with many managed Kubernetes services.

On the other hand, you have the fact that deploying and maintain a self-managed cluster is time-consuming and requires deeper knowledge from the maintainer. Cloud providers have specific teams to take care of these solutions and in general, this adds more reliability to the solutions provided.

What does it mean to have "Managed Kubernetes"?

Despite the fact that Kubernetes is free source, many firms who want to use it don't have the skills or resources to set up and manage the cluster. Managed Kubernetes providers assist those interested in using Kubernetes by providing the necessary support and management for Kubernetes clusters. A managed Kubernetes deployment should offer users a painless control plane, simple deployment options, and ongoing Kubernetes maintenance, allowing them to focus on their company and getting their apps to market.

Many cloud providers offer managed Kubernetes services, however the configuration options and degree of support they provide to their customers vary slightly.

A managed Kubernetes cluster is preferable to a self-managed one if we take the following into consideration:

- A single cloud
- You do not require the most recent Kubernetes releases.
- You're ready to delegate vulnerability patching to your cloud provider

Here are three main contenders from AWS, Azure, and GCP:

- **Azure Kubernetes Service (AKS)** — Originally, Azure had a service called Azure Container Service. This supported not only Kubernetes but also Apache Mesos and Docker Swarm. With Kubernetes seeing its popularity outgrow the competition by a significant margin, Azure replaced its Azure Container Service with Azure Kubernetes Service in June 2018.

- **Amazon Elastic Kubernetes Service (EKS)** — Similar to Microsoft, Amazon first released its Elastic Container Service, and then in June 2018, they released the Elastic Kubernetes Service.

- **Google Kubernetes Engine (GKE)** — It should come as no surprise that Google was the first cloud provider to release its Google Kubernetes Engine (GKE). After all, Kubernetes was originally developed at Google for their internal applications. GKE was released in 2015.
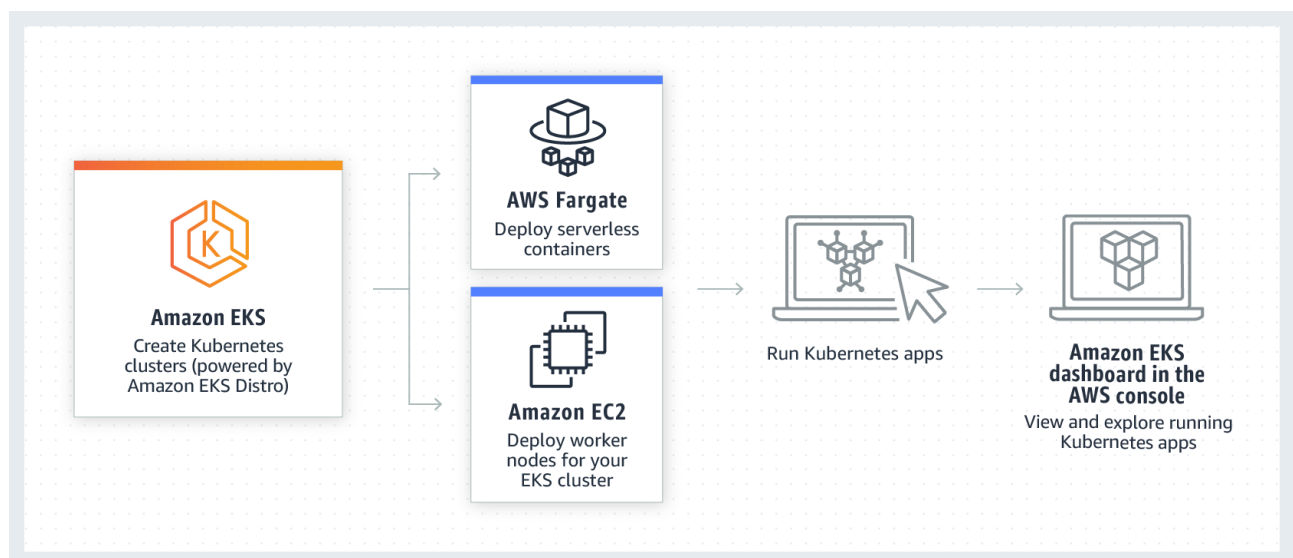
Kubernetes on AWS

With EKS, Amazon fully manages the control plane -- including components like etcd and Kubernetes API server with integration to other AWS services, such as Identity and Access Management (IAM).

The entire management infrastructure runs behind the scenes, across multiple availability zones. AWS automatically replaces any unhealthy nodes in order to maintain high availability. It also handles all the upgrades and patching. For users, the EKS control plane is essentially a black box.

For the data plane, there are three options for EKS users: self-managed nodes, EKS-managed node groups and AWS Fargate.
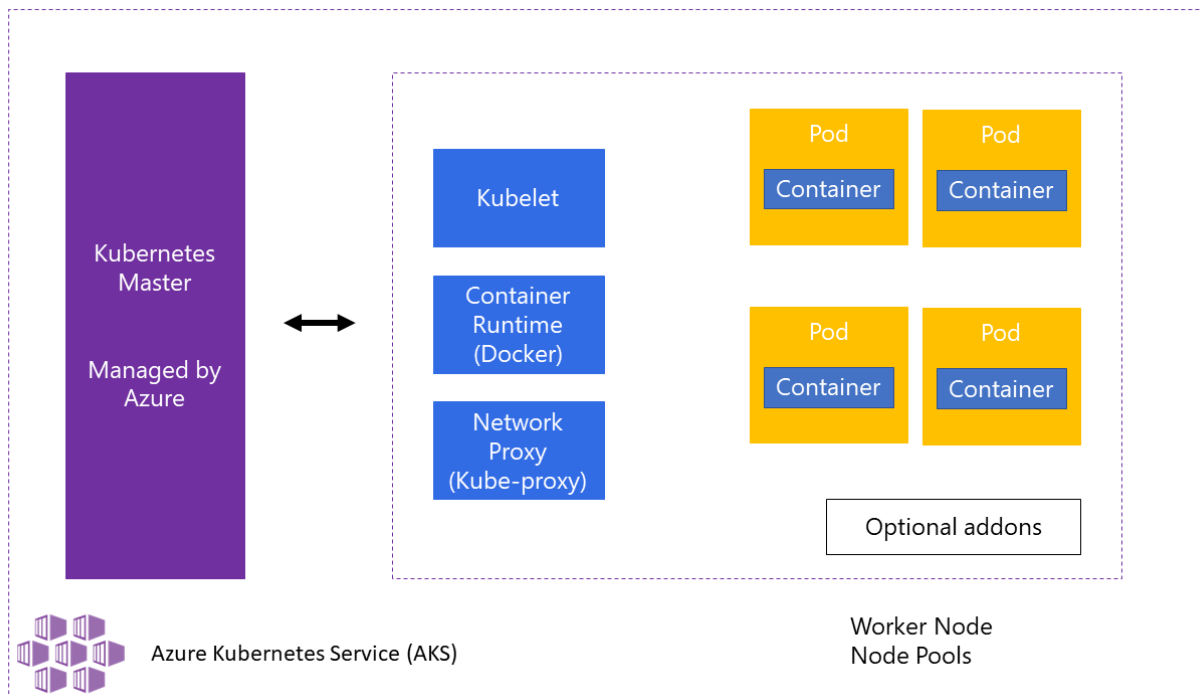
Kubernetes on AZURE

Azure Kubernetes Service (AKS)

Azure Kubernetes Service is a robust and cost-effective container orchestration service that helps you to deploy and manage containerized applications in seconds where additional resources are assigned automatically without the headache of managing additional servers.

AKS nodes are scaled-out automatically as the demand increases. It has numerous benefits such as security with role-based access, easy integration with other development tools, and running any workload in the Kubernetes cluster environment. It also offers efficient utilization of resources, removes complexities, easily scaled-out, and migrates any existing workload to a containerized environment and all containerized resources can be accessed via the AKS management portal or AKS CLI

Kubernetes on GCP

Kubernetes was created by Google for its own container orchestration purposes. This history of Kubernetes in Google labs is why GKE is considered the most advanced managed Kubernetes service. GKE includes health checks and automatic repair of microservices, logging, and monitoring with operations suite (formerly Stackdriver). In addition, it comes with four-way auto-scaling and multi-cluster support.

Some of the major benefits of GKE are:

- Single-click clusters
- A high-availability control plane including multi-zonal and regional clusters
- Auto-repair, auto-upgrade, and release channels
- Vulnerability scanning of container images and data encryption
- Integrated cloud monitoring with infrastructure, application, and Kubernetes-specific views

GKE provides two main modes of cluster configuration:

- **Autopilot:** In this mode, GKE provisions and manages the cluster's underlying infrastructure, including nodes and node pools, giving users an optimized cluster with a hands-off experience.
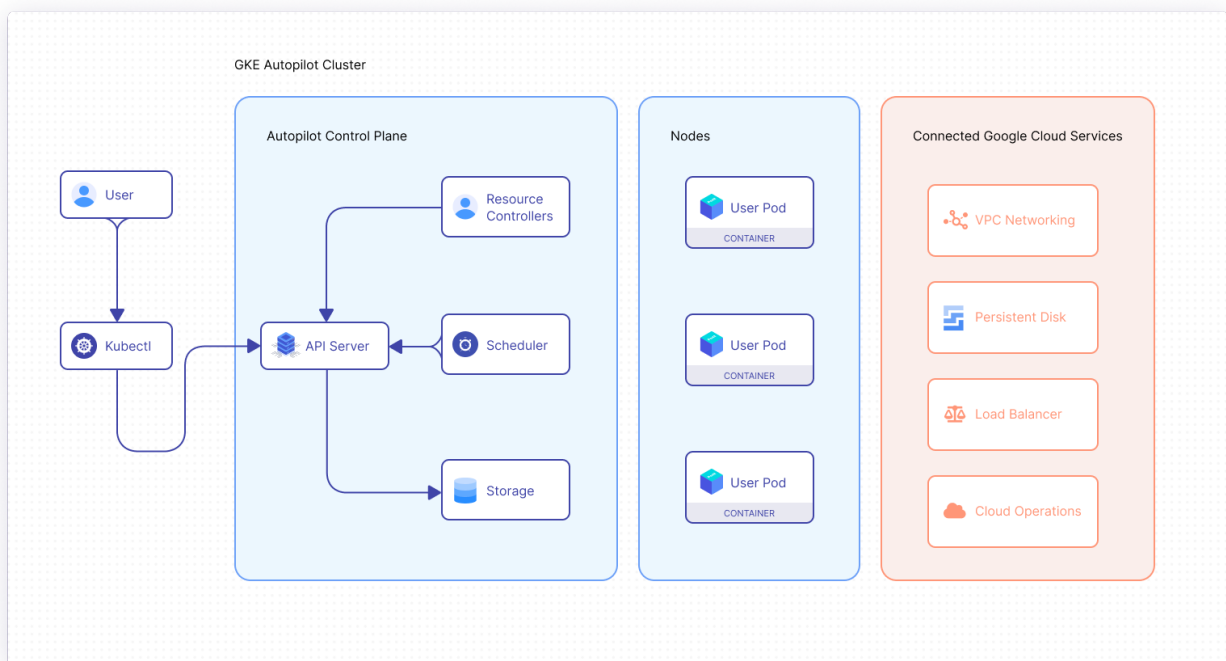
Standard: In this mode, users manage their cluster's underlying infrastructure, allowing for more node configuration flexibility.