

Sprint 2 Research Summary

Here, I will give a summary of:

1. Kubernetes
2. Amazon EKS
3. OpenStack
4. Magnum
5. OpenShift
6. OpenShift VS OpenStack
7. OpenShift VS Kubernetes
8. Resources

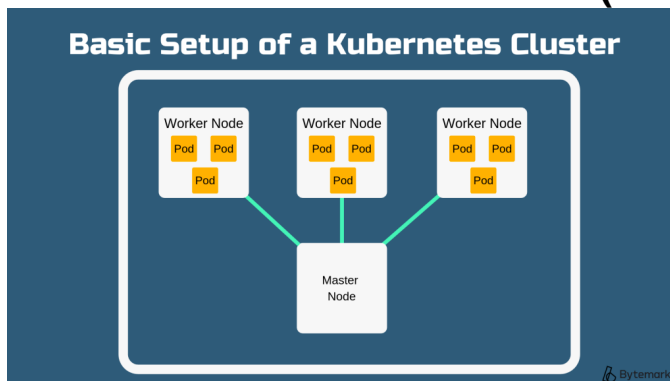
1. Kubernetes

1.1. Overview

Kubernetes is an open-source system that allows organizations to **deploy and manage containerized applications** like platforms as a service (PaaS), batch processing workers, and microservices in the cloud at scale.

Through an abstraction layer created on top of a group of hosts, development teams can let Kubernetes manage a host of functions--including load balancing, monitoring and controlling resource consumption by team or application, limiting resource consumption and leveraging additional resources from new hosts added to a cluster, and other workflows.

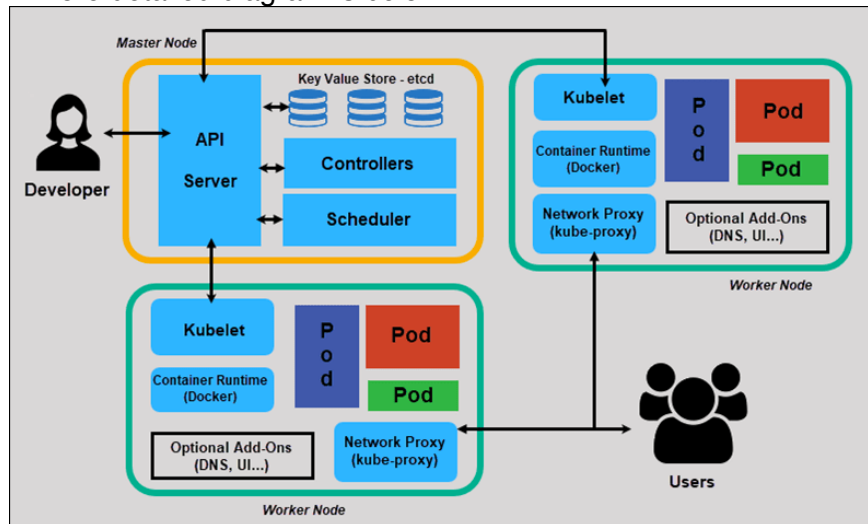
1.2. Kubernetes Architecture (Master, Worker)



The Kubernetes master is responsible for **maintaining the desired state for your cluster**. The master **can also be replicated for availability and redundancy**. When you interact with Kubernetes, eg. via the kubectl command-line interface, you're communicating with the master.

The worker nodes in a cluster are the machines (VMs, physical servers, etc) that run your applications and cloud workflows. The master controls each node; you'll rarely interact with nodes directly.

A more detailed diagram is below:



2. Amazon EKS

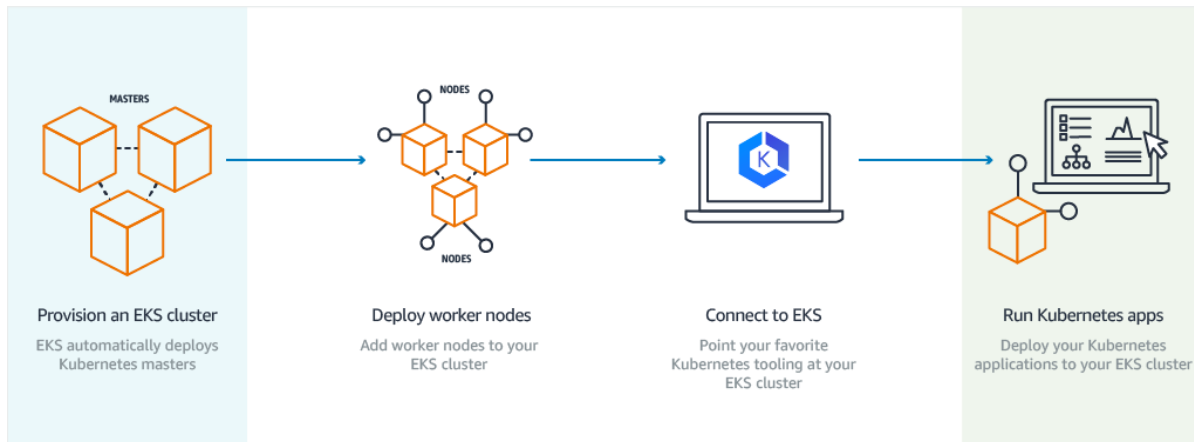
2.1. Overview

Amazon EKS is a **managed service** that helps make it very **easy to run Kubernetes on AWS**. The idea is that most applications will run on EKS with minimal mods, if any.

Through EKS, organizations can run Kubernetes without cumbersome steps, such as:

- Creating the Kubernetes master cluster
- Configuring service discovery, Kubernetes primitives
- Porting and Creating database instances
- Setting up load balancing (eg. with HA proxy)
- Security
- Networking
- Hosting Control Planes across different availability zones to prevent single point of failure (Highly Available)
- Managing Control Plane, so users do not need to worry about components like etcd, kube-controller-manager, kube-apiserver, cloud-controller-manager and kube-scheduler.

Basically **EKS = Kubernetes-as-a-service**

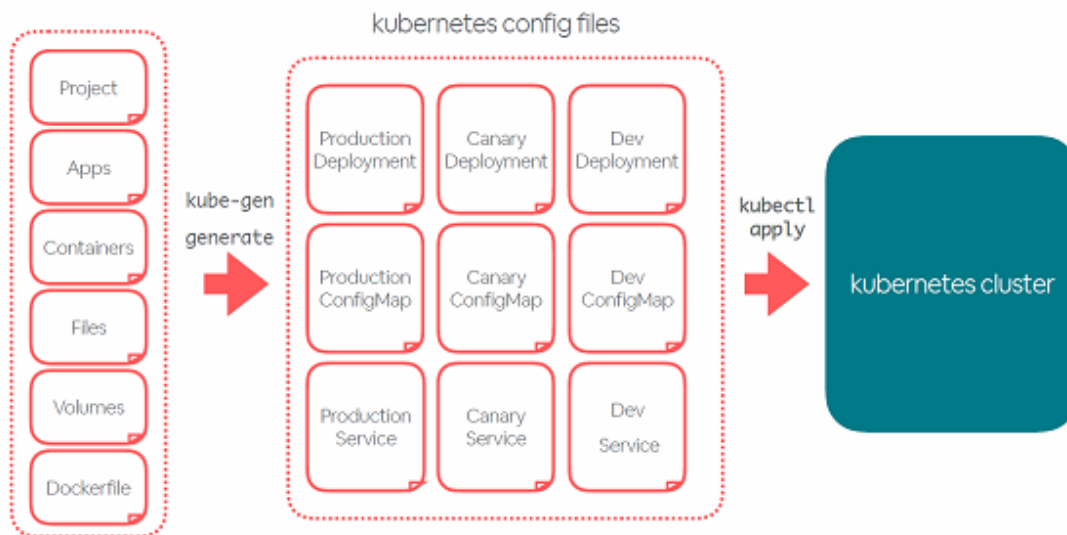


2.2. Running Kubernetes without EKS: Manual deployment on EC2

IT teams can run a self-hosted Kubernetes environment on an EC2 instance. Deploy with tools like kops, which help create and manage the Kubernetes cluster. In this case, the control plane will be visible and available to the users, who will be able to see components running on dedicated EC2 instances. Admins will also have to patch and maintain everything by hand, which can be inconvenient in many cases.

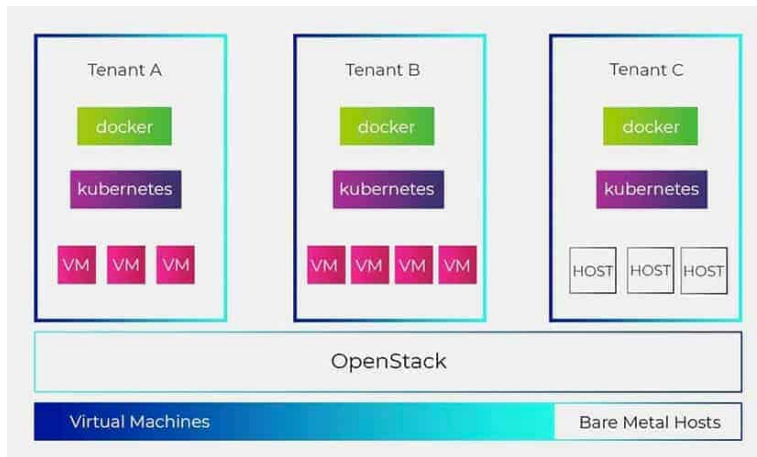
generating k8s configs

@MELANIECEBULA



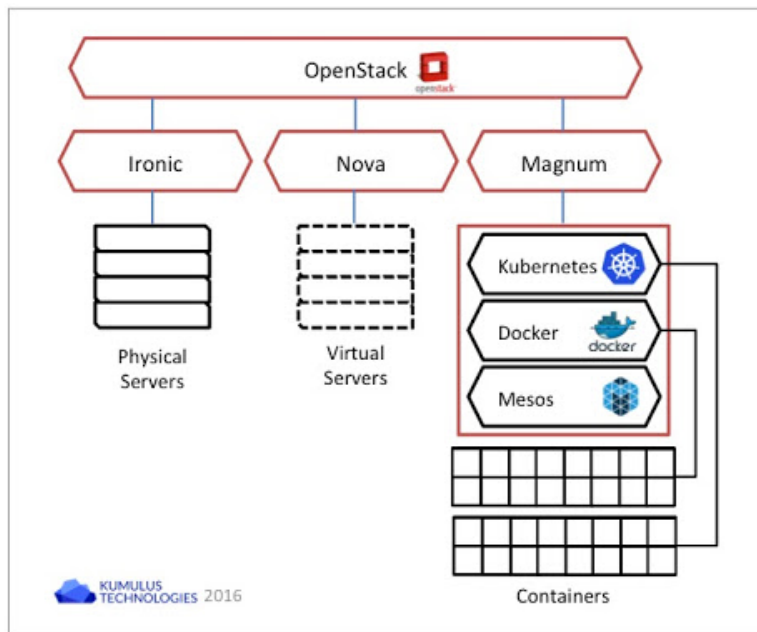
3. OpenStack

OpenStack an **open-source cloud operating system** that turns your server into cloud environments. In other words, it **provides an open alternative to the top cloud providers**. It is IaaS, and it can be used to automate resource allocation so customers can **provision virtual resources** like VPS, block storage, object storage among other things.



4. Magnum

Magnum is an **OpenStack API service that makes container orchestration engines**, such as Docker Swarm, Kubernetes, and Mesos available, **a first class resources in OpenStack**. Magnum uses **Heat** to orchestrate an OS image, which contains Docker and Kubernetes and runs that image in either virtual machines or bare metal in a cluster configuration.

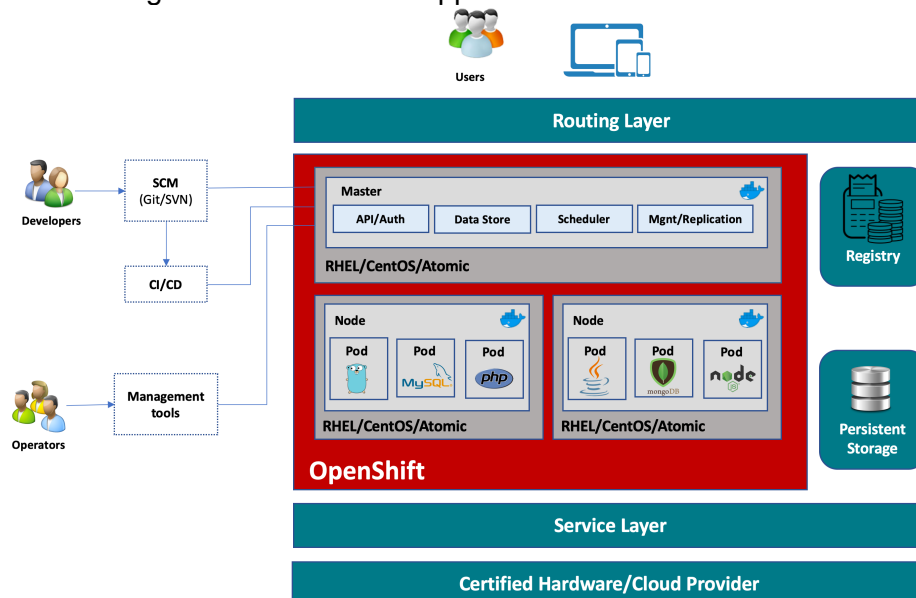


5. OpenShift

OpenShift is a **platform as a service (PaaS)** that leverages the core concepts of Kubernetes and packages them in a neat way for developers to deploy applications on the cloud. In short, it's a modded Kubernetes, and accepts `kubectl` commands.

OpenShift Container Platform adds the following features to Docker & Kubernetes:

- Source code management, builds, and deployments for developers
- Managing and promoting images at scale as they flow through your system
- Application management at scale
- Team and user tracking for organizing a large developer organization
- Networking infrastructure that supports the cluster



6. OpenShift vs OpenStack

OpenStack turns servers into cloud. It can be used to automate resource allocation so customers can **provision virtual resources**.

OpenShift is a container centric model that leverages core concepts of Kubernetes and packages them in a neat way for developers to deploy applications on the cloud.

6.1. Concerning Containers

OpenStack typically uses hypervisors like KVM, Xen or VMware to spin up virtual machines. On the other hand, **OpenShift** can run bare metal or it may run on Virtual Machines but it always uses containers on top of them. The containerization technology that they use is almost exclusively Docker.

(Note: OpenStack does offer containerization support as well, it is meant to be used more of less like VPS and is optional.)

6.2. Distributed System

OpenStack is **not exclusively a distributed system**. It can take control over an entire data center but that's nowhere as global as a Kubernetes cluster. You would need a lot of extra layers of software to turn several remote installations of OpenStack into a single distributed system. It may offer distributed storage technologies like Glusterfs and Ceph which you can use to build up your distributed storage stack, but it doesn't rely on them specifically.

OpenShift is **inherently a distributed system** (which makes sense since it incorporates Kubernetes into it) and since it is a distributed system it necessarily **has one or more master nodes controlling a few worker nodes**. Red Hat provides a decent layer for networking to facilitate communication between these nodes and there's good support for using distributed storage like Glusterfs and Ceph.

6.3. IaaS and PaaS

OpenStack is **Infrastructure as a Service (IaaS)**. It manages servers and offers services that you generally attribute to the cloud VPS providers like DigitalOcean, AWS, Microsoft Azure and Google compute engine. If you are planning to set up your own private data center (due to economics/security reasons), instead of developing your own **management utility to run these servers** you can deploy OpenStack on them and turn it into your organization's private **cloud**.

OpenShift offers **Platform as a Service**. If you are familiar with Google App Engine where developers are concerned with just the **development and testing of their app**. The entire **server configuration doesn't matter**. Google makes it possible because they containerize the apps running in the cloud.

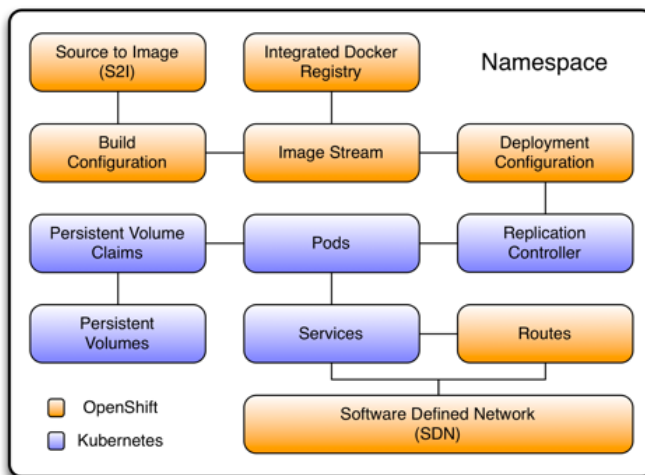
This is basically what OpenShift does. OpenShift can containerize the applications your organization is developing. Your DevOps team can set up an entire CD/CI pipeline, do fancy A/B testing, issue tracking and streamline the flow between various stages of development, testing and deployment using OpenShift.

6.4. OpenShift, OpenStack and Hybrid Cloud

It is not necessary to run OpenShift on top of OpenStack, but they can go hand in hand if you desire to. OpenStack does basically the same thing as your cloud service providers (AWS, Azure.etc). You can then install OpenShift on them.

7. OpenShift VS Kubernetes

Openshift is based on Kubernetes and docker. In other words, OpenShift is a modded version of Kubernetes. Below is an example of the namespace component of Kubernetes. As you can see, OpenShift replaces some of the original Kubernetes components with their own.



Below is a more extensive list of differences:


OPENSIFT

kubernetes

Point of Difference		
Product or Project	OpenShift is a product	Kubernetes is an open-source framework (project)
Flexibility	OpenShift has some limits in terms of flexibility	Offers better flexibility with operating systems
Security	Very strict; need certain level of permissions to maintain minimum security level	Easy to maintain security level
Deployment Approach	It has DeploymentConfig	It has deployment objects
Router vs Ingress	Router objects allows external access to the OpenShift	Ingress objects allows external access to the kubernetes services; mature than router
Management	ImageStreams make OpenShift better in terms of management	Management of container images in not so easy in kubernetes
User Experience	OpenShift provides better support to the users	Kubernetes requires additional tools for good user experience
Networking	It offers its native networking solutions to users	Some of the kubernetes components don't have networking solutions and offer third party plugins
Service Catalog	OpenShifts proves better in terms of service catalog	Kubernetes service catalog has less provision of services within clusters

8. Resources

Openstack:

- <https://phoenixnap.com/blog/kubernetes-vs-openstack>

Openshift

- <https://medium.com/levvel-consulting/the-differences-between-kubernetes-and-openshift-ae778059a90e>
- <https://docs.openshift.com/container-platform/3.5/architecture/index.html>
- <https://www.whizlabs.com/blog/openshift-vs-kubernetes/>
- <https://medium.com/levvel-consulting/the-differences-between-kubernetes-and-openshift-ae778059a90e>

Magnum

- <https://wiki.openstack.org/wiki/Magnum>
- <https://vexxhost.com/blog/openstack-magnum-and-why-it-matters-when-running-kubernetes/>
- <https://stackshare.io/stackups/kubernetes-vs-magnum-ci>