NURTURE
Education Solutions
TOMORROW'S HERE

Subject: Containerization using Dockers

Module Number: 5

Module Name: Docker Orchestration and Service Discovery

AIM:

The aim of this module is to learn container orchestration, kubernetes and service discovery.

Docker Orchestration and Service Discovery

Syllabus

- Docker Compose
- Consul
- Service Discovery and Docker
- Docker Swarm
- Container Orchestration
- Limitations to using Docker
- Orchestration Alternatives and Components- Fleet and etcd
- Kubernetes
- Docker vs Kubernetes
- Apache Mesos
- Helios
- Centurion

Docker Orchestration and Service Discovery

Objectives:

- List the benefits of containerization.
- Classify the Docker compose to create multi-container applications.
- Understand best practices of Dockerfiles and Image building.
- Explain how to Run Docker Commands on the command line.
- Develop and run Docker containers.

Course Outcomes:

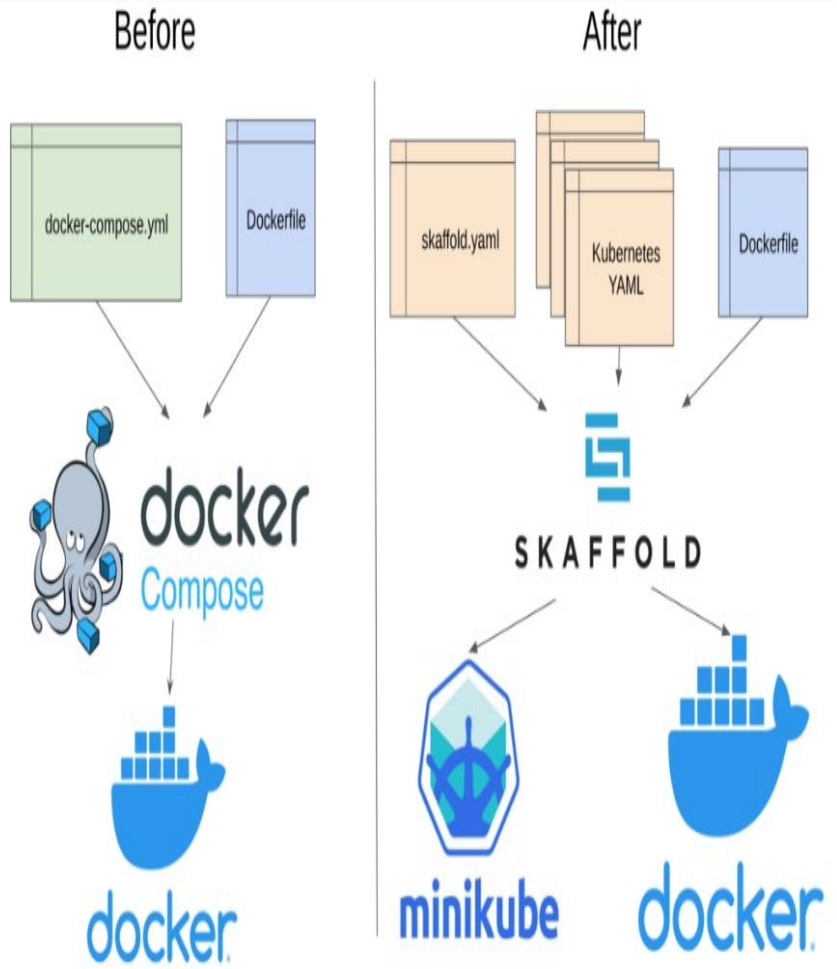
- Develop Containerized applications and implement continuous integration using Docker, explaining different types of cloud deployment and service models.
- Create own images and build the repository.
- Utilize Docker Orchestration and Service discovery features.
- Apply development tools, frameworks, platforms, libraries and packages to test hardware and software systems.
- Evaluate the fundamentals of solution architecture to provision cloud infrastructure.

Docker Compose

- Docker Compose is primarily a tool meant for defining and running multi-container Docker applications.

Using Compose is a three-step process:

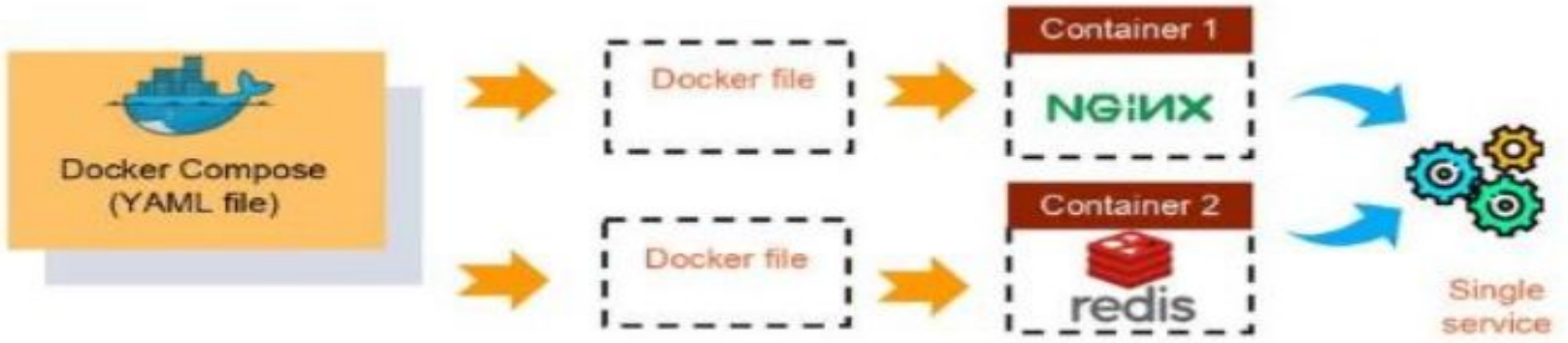
- First, define your app's environment with a *Dockerfile* so that it can be reproduced anywhere.
- Next, define and configure the services which form your app in *docker-compose.yml* so that they can be run together in an isolated environment.
- Run *docker-compose up* and then Compose starts and runs the entire application.



Features of Compose

The features of Compose that make it effective are:

- Multiple isolated environments on a single host
- Only recreate containers that have changed: Compose caches the configuration used to create a container. When you restart a service that has not changed, Compose re-uses the existing containers.



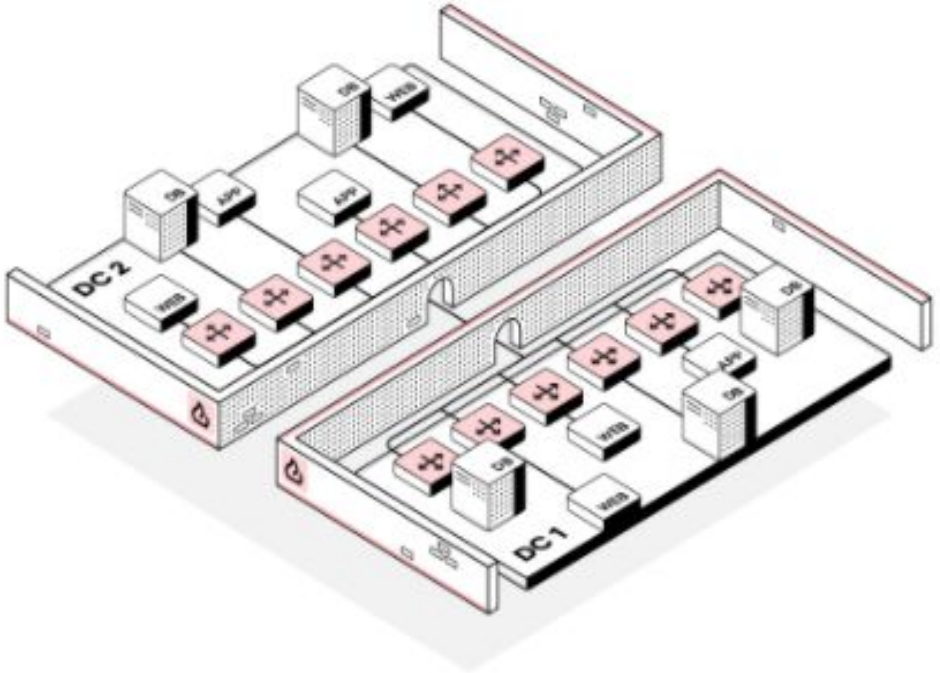
Consul

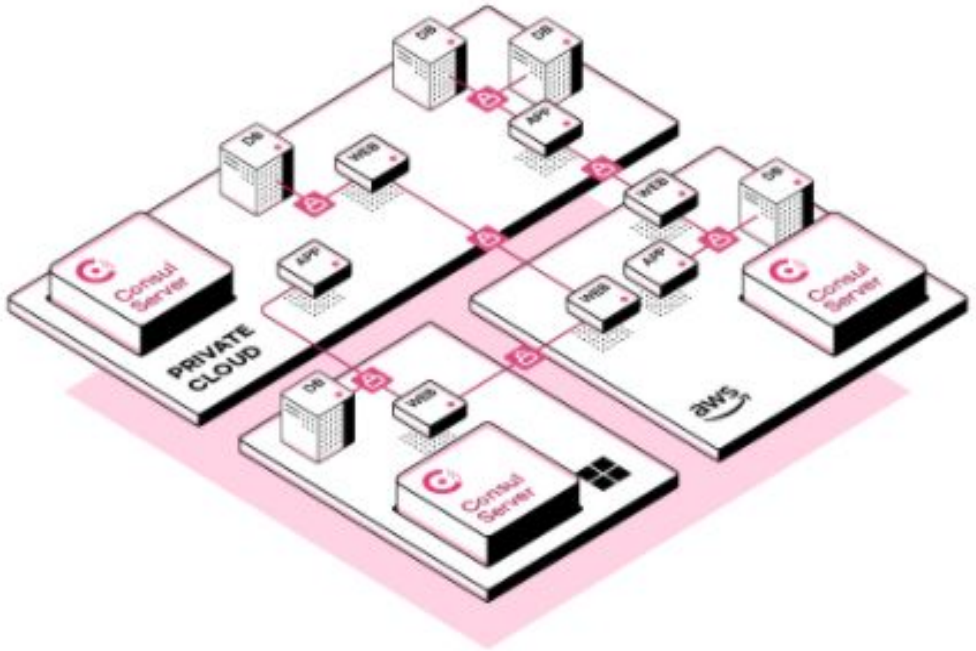
- Consul is a datacenter runtime and a distributed, highly-available, and multi-datacenter aware tool that provides service discovery, configuration, and orchestration capabilities.
- Consul is **a Hashicorp based tool for discovering and configuring a variety of different services in an infrastructure**. It is built on Golang.
- Consul enables rapid deployment, configuration and maintenance of service-oriented architectures at massive scale.

Why Consul ?

- Micro-service based Networking
- Secure Service-to-Service Access
- Automated Networking Tasks

Service-based networking for Dynamic Infrastructure





Static Infrastructure

Dynamic Infrastructure

Docker Orchestration and Service Discovery

Static Infrastructure

Private datacenters with static IPs, primarily north-south traffic, protected by perimeter security and coarse-grained network segments.

TRADITIONAL APPROACH

- ! Static connectivity between services
- ! A fleet of load balancers to route traffic
- ! Ticket driven processes to update network middleware
- ! Firewall rule sprawl to constrict access and insecure flat network zones

Dynamic Infrastructure

Multiple clouds and private datacenters with dynamic IPs, ephemeral containers, dominated by east-west traffic, no clear network perimeters.

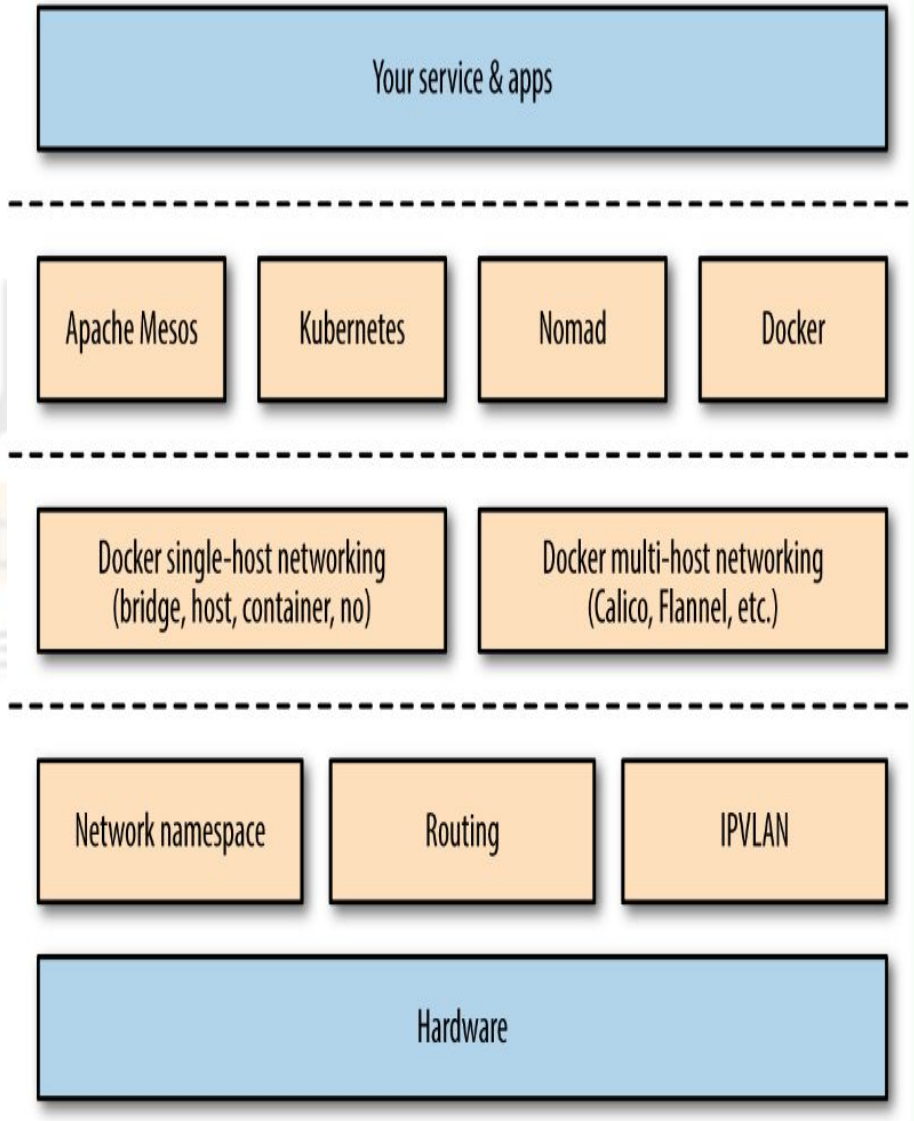
CONSUL APPROACH

- ✓ Centralized registry to locate any service
- ✓ Services discovered and connected with centralized policies
- ✓ Network automated in service of applications
- ✓ Zero trust network enforced by identity-based security policies

Docker Orchestration and Service Discovery

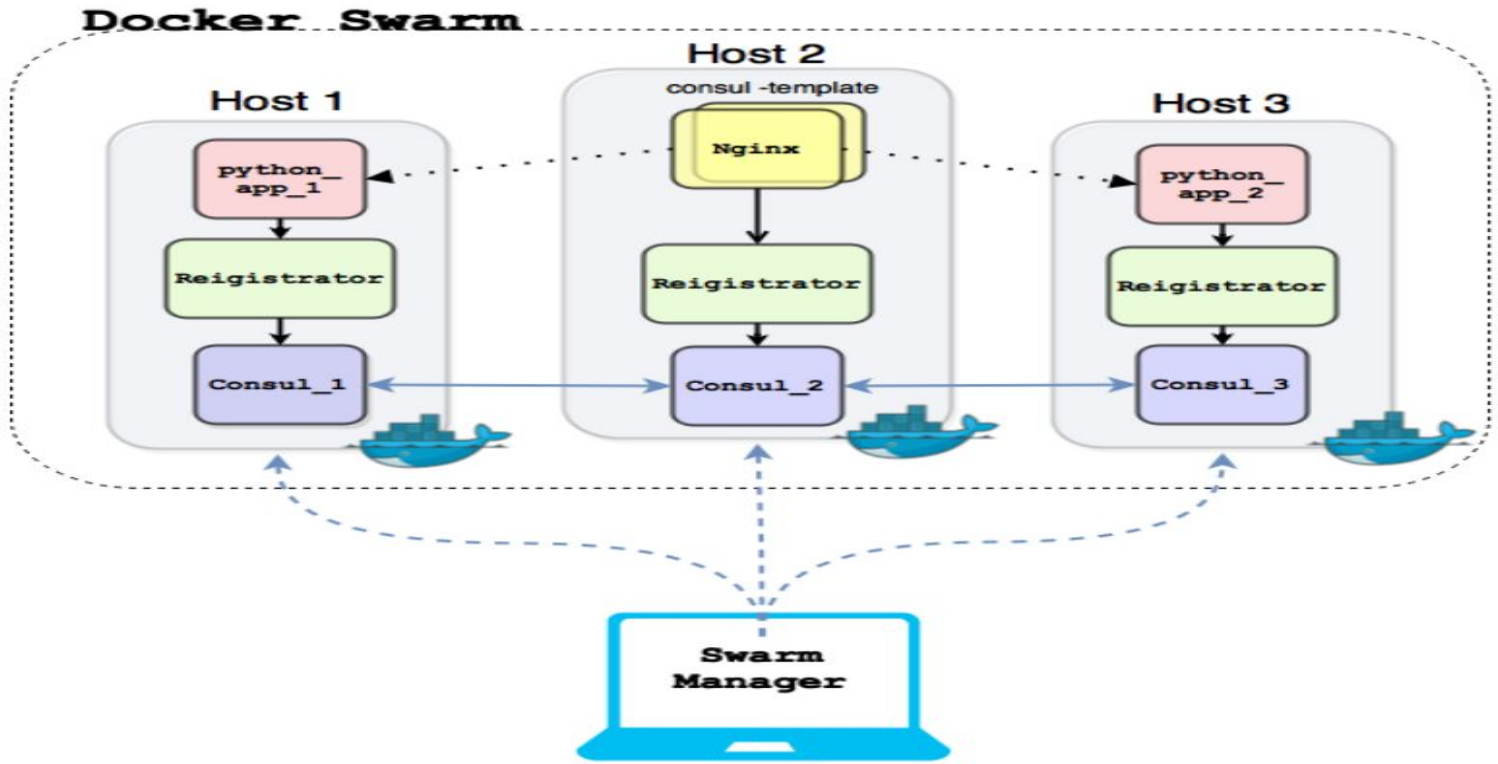
Docker Service Discovery

- Service discovery is **how applications and (micro)services locate each other on a network**.
- It is the **process of automatically detecting devices and services on a network**. Service discovery protocol (SDP) is a networking standard that carries out detection of networks by resource identification.
- The Docker Engine has an embedded DNS server inside it. Each **container forwards its queries to the Docker Engine**, which in turn checks if the container or service is on the same network as the container that sent the request. If it is on the same network, it searches the IP (or virtual IP) address that matches a container, a task's or a service's name in its internal key-value store and returns it to the container that sent the request.



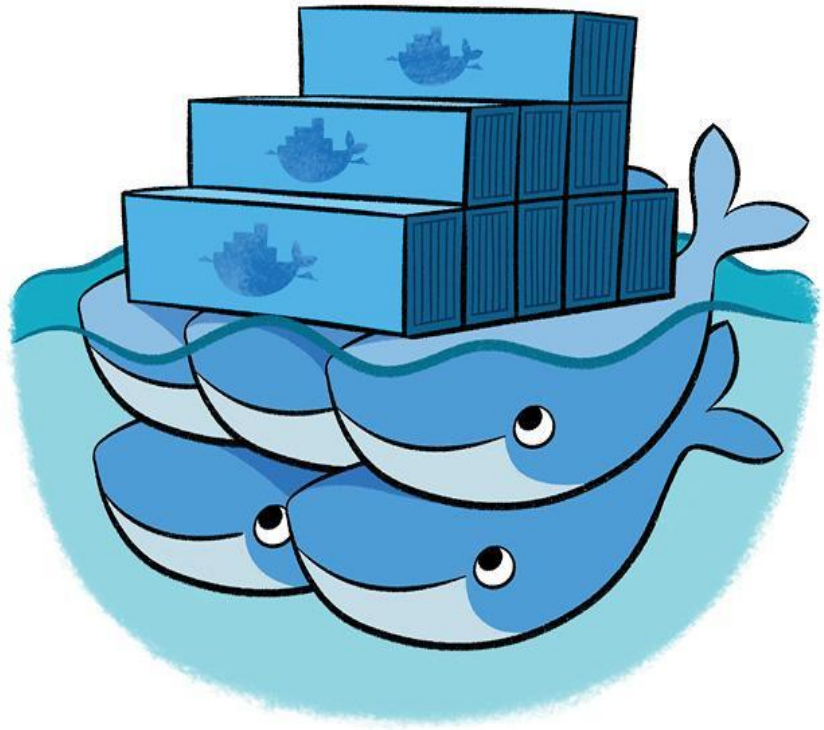
How does Consul Service Discovery work?

- Distributed applications can make use of **Consul to dynamically discover service endpoints**. Once a service is registered with Consul, it can be discovered using typical DNS or custom API.



Docker Swarm

- Docker swarm is a container orchestration tool, that allows the user to manage multiple containers deployed across multiple host machines .
- It provides native clustering functionality for Docker containers, which turns a group of Docker engines into a single virtual Docker engine.



Orchestration

- Orchestration describes automated arrangement, configuration and management of complex computer systems and services along with coordination among those automated activities.
- ***Cloud orchestration*** is the use of programming technology to manage the interconnections and interactions among workloads on public and private cloud infrastructure.

Cloud Orchestration vs Automation:

- Automation is a subset of orchestration, which means that orchestration provides coordination among and across many automated activities.
- Automation focuses on making one task repeatable rapidly with minimal operator intervention whereas orchestration does it on the whole.

Container Orchestration

- It is the automatic process of deploying, scaling, managing or scheduling the work of individual containers for applications within multiple clusters(group of machines).
- **Container orchestration tools** provide a framework for managing containers and are used for container lifecycle management. Some popular options are Kubernetes, Docker Swarm.

Limitations to using Docker

- Docker, by itself, can manage single containers but when we start using more and more containers and containerized apps, management and orchestration can get difficult.
- That's where **Kubernetes** comes into picture.
- Kubernetes is an open-source container-orchestration system for automated application deployment, scaling, and management of containerized applications.

Orchestration Alternatives- Etcd

- "Etcd" stands for "/etc distributed"; it is a highly reliable configuration mechanism that offers a uniform view across a cluster of machines. It provides the feature of "sequential consistency", which ensures that all changes are visible in the same order on all machines in the cluster — though not necessarily at the same time.
- Changes have sequence numbers attached to them; those numbers can be used to wait until a given change has been distributed throughout the cluster. Sequence numbers are unique and are not reused; getting a value at a specific sequence number will always return the same result, anywhere in the cluster.
- Etcd allows runtime reconfigurability.
- Clusters built with etcd are highly interconnected. One system is elected to be the "leader"; all others are "followers." At its core, etcd is a simple key/value data store. Along with the usual store, fetch, and delete operations, etcd provides an atomic compare-and-swap operation and a compare-and-delete operation as well.
- The etcdctrl command-line tool can be used to make changes to the etcd database or to look for changes made by others.
- Kubernetes is a cluster management system from Google that was built on top of etcd.

Orchestration Alternatives- Fleet

- Etcd is good for passing information to tasks on a clustered system, but for starting, stopping, and managing those tasks, fleet tool is used.
- Fleet is based on systemd. Systemd handles resource limits, control groups, and the secure computing (seccomp) subsystem; it can also take care of process monitoring and notification of dependent processes. It is an init system driven by an API that gives control over the system as a whole.
- Fleet is a **cluster scheduler**. Its job is to distribute tasks across the machines in a cluster. It has to respond to events such as when a machine goes down and reschedule tasks as required.
- Fleet can handle a number of requirements attached to the tasks it runs. For example, certain tasks need to be run together on the same machine; whereas others need to be run on a specific system within the cluster. Information about such requirements goes into the systemd unit files, using the special "X sections" that are ignored by systemd itself.

Kubernetes

- **Kubernetes** is an open-source container-orche system for automated application deployment, scal management of containerized applications.
- It was originally designed by Google and maintained by the Cloud Native Computing Founda

kubernetes

Kubernetes

- **Kubernetes is a cluster and container management tool.**
- Kubernetes is more extensive than Docker Swarm and is meant to efficiently coordinate and manage clusters of nodes.
- It deploys containers to clusters,i.e. a network of virtual machines. The basic idea is to further abstract machines, storage and networks away from their physical implementation.

kubernetes

Docker vs Kubernetes

Docker	Kubernetes
1. It is a platform and tool for building, distributing, and running Docker containers	1. It is a container orchestration system for Docker containers.
2. Runs on a single node	2. Runs across a cluster

Apache Mesos

- Apache Mesos is an open source cluster manager capable of handling workloads in a distributed environment with the help of dynamic resource sharing and isolation.
- Mesos is suitable for the **deployment and management of applications in large-scale clustered environments**.
- Apache Mesos abstracts CPU, memory, storage, and other computing resources from machines (physical or virtual), enabling fault-tolerant and elastic distributed systems to be built easily and run effectively.
- Mesos was quickly adopted by leading companies like Twitter, Apple(Siri), Uber, Netflix.

Benefits of Apache Mesos

- **Abstract data center resources** into a single pool to simplify resource allocation while providing a consistent application and operational experience across private or public clouds.
- **Colocate diverse workloads** on the same infrastructure such analytics, stateless microservices, distributed data services and traditional apps to improve utilization and reduce cost and footprint.
- **Automate day-two operations** for application-specific tasks such as deployment, self healing, scaling, and upgrades; providing a highly available fault tolerant infrastructure.
- **Provide evergreen extensibility** to run new application and technologies without modifying the cluster manager or any of the existing applications built on top of it.
- **Elastically scale** the application and the underlying infrastructure from a handful, to tens, to tens of thousands of nodes.

Helios

- *Helios is a Docker container orchestration platform, by Spotify*.
- It is also an open-source container tool like Kubernetes.
- It is a Docker orchestration platform for deploying and managing containers across an entire fleet of servers.
- Helios provides HTTP API as well as a command-line client to interact with servers running our containers.

Centurion

- Centurion is an open source deployment tool for Docker developed by New Relic software company. It is used internally to run their production infrastructure.
- Centurion is a mass deployment tool for Docker fleets. It takes containers from a Docker registry and runs them on a fleet of hosts with the correct environment variables, host volume mappings, and port mappings. It supports rolling deployments out of the box, and makes it easy to ship applications to Docker servers.
- Centurion works in a two phase deployment process where the build process ships a container to the registry and Centurion ships containers from the registry to the Docker fleet.
- If we want Centurion to set up the structure and create a sample config, we can simply run centurionize once the Ruby Gem is installed.

Tasks of Centurion:

Centurion performs a number of tasks with distributed containers:

- **Rolling deployment to a fleet of Docker servers-** ***rolling_deploy*****:** stops and starts each container one at a time to make sure that the application stays available from the viewpoint of the load balancer. As the deploy runs, a health check will hit each container to ensure that the application booted correctly.
- **Deploy to a fleet of Docker servers-** ***deploy*****:** hard stops, then starts containers on all the specified hosts. Not recommended for apps where one endpoint needs to be available at all times.
- **Deploy a bash console on a host-** ***deploy_console*****:** starts a command line shell with the existing environment passed to the container. The CMD from the Dockerfile will be replaced with */bin/bash*, using the first host from the host list.

Tasks of Centurion:

Centurion performs a number of tasks with distributed containers:

- **List all the tags running on servers for a particular project-** ***list:running_container_tags*****:** returns a list of all the current tags and which machines they are running on. Gives a unique list of tags across all hosts as well, useful for validating the state of the deployment in the case where something goes wrong mid-deploy.
- **List all the containers currently running for this project-** ***list:running_containers*****:** returns a list of all the containers for a project on each of the Docker servers from the config.
- **List registry images-** ***list***: returns a list of all the images in the registry for the project.

Summary

- Docker Compose is primarily a tool meant for defining and running multi-container Docker applications.
- Consul is a datacenter runtime , a distributed, highly-available, and multi-datacenter aware tool that provides rapid deployment, service discovery, configuration, and orchestration capabilities at massive scale. Service Discovery is the process of automatically detecting devices and services on a network. Once a service is registered with Consul, endpoints can be discovered using typical DNS or custom API.
- Docker swarm is a container orchestration tool, that allows the user to manage multiple containers deployed across multiple host machines . Automation is a subset of orchestration, which means that orchestration provides coordination among and across many automated activities.

Summary

- Docker, by itself, can manage single containers but when we start using more and more containers and containerized apps, management and orchestration can get difficult. That's where Kubernetes comes into picture.
- Kubernetes is an open-source container-orchestration system for automated application deployment, scaling, and management of containerized applications.
- Apache Mesos is an open source cluster manager capable of handling workloads in a distributed environment with the help of dynamic resource sharing and isolation.
- Helios is a open-source container orchestration platform by Spotify for deploying and managing containers across an entire fleet of servers.
- Centurion is a open-source mass deployment tool for Docker fleets. It takes containers from a Docker registry and runs them on a fleet of hosts with the correct environment variables, host volume mappings, and port mappings.

Self Assessment Questions

1. _____ is a tool for defining and running multi-container Docker application

a. Docker swarm

b. Docker compose

c. Docker hub

d. None

Answer: b

Self Assessment Questions

2. Consul is a data-center aware too that provides:

a. Rapid deployment

b. Service discovery

c. Orchestration capabilities

d. All of the above

Answer: d

Self Assessment Questions

3. On which language is Consul built ?

a. Python

b. C

c. Golang

d. Java

Answer: c

Self Assessment Questions

4. The process in which applications automatically detect and locate each other on a network:

a. Container Networking

b. Service Discovery

c. Container Orchestration

d. Automation

Answer: b

Self Assessment Questions

5. _____ is a container orchestration tool to manage multiple containers deployed across multiple host machines.

a. Docker swarm

b. Docker compose

c. Docker hub

d. None

Answer: a

Self Assessment Questions

6. _____ is the arrangement, configuration and management of systems and services along with coordination among those activities.

a. Automation

b. Orchestration

c. Networking

d. Service Discovery

Answer: b

Self Assessment Questions

7. State True or False: Automation is subset of Orchestration.

a. True

b. False

Answer: a

Self Assessment Questions

8._____ provides native clustering functionality for turning group of Docker engines into a single virtual Docker engine.

a. Docker swarm

b. Docker compose

c. Docker hub

d. None

Answer: a

Self Assessment Questions

9. _____ is a cluster scheduler whose job is to distribute tasks across the machines in a cluster and also reschedule tasks as per requirement.

a. Etcd

b. Fleet

c. Kubernetes

d. Helios

Answer: b

Self Assessment Questions

10. _____ is an open source container orchestration platform by Spotify for deploying and managing containers across an entire fleet of servers.

a. Kubernetes

b. Helios

c. Mesos

d. Centurion

Answer: b

Self Assessment Questions

11. _____ is an open source cluster manager capable of handling workloads in a distributed environment with the help of dynamic resource sharing and isolation.

a. Kubernetes

b. Helios

c. Apache Mesos

d. Centurion

Answer: c

Self Assessment Questions

11. _____ is an open-source mass deployment tool for Docker fleets.

a. Mesos

b. Helios

c. Kubernetes

d. Centurion

Answer: d

Self Assessment Questions

12. _____ is an open-source container-orchestration and cluster management tool designed by Google for automated application deployment, scaling, and management of containerized applications.

a. Mesos

b. Helios

c. Kubernetes

d. Centurion

Answer: c

Assignment

1. Write a note on Docker Service Discovery and its working.
2. Elaborate the usage of various Container Orchestration tools.
3. Deploy Docker Image across multiple host?
4. Managing availability and redundancy of containerized applications?
5. Auto scaling containers in response to changes in application load?
6. Ensuring containers run on a host with appropriate hardware resources, and moving them away from a host that does not provide the required resources?

Docker Orchestration and Service Discovery

Document Link

Topics	Video Link	Note
Docker Consul	https://www.consul.io/	Compose is a tool for defining and running multi-container Docker applications.
Docker Hub	https://hub.docker.com/_/swarm	Explain how to create Docker hub
Mesos Apache	http://mesos.apache.org/documentation/latest/	Discussed Fundamentals Build / Installation

Video Links

Topics	Video Link	Notes
Docker Compose	https://youtu.be/DX1T-PKHKhg	Docker Compose
Introduction to Consul	https://youtu.be/mxeMdl0KvBI	Introduction to Consul
Container Orchestration	https://youtu.be/kBF6Bvth0zw https://youtu.be/HDt_iN1hINA	Container Orchestration
Container Orchestration using Kubernetes, Kubernetes vs Docker Swarm	https://youtu.be/7CsgkXxnINE	Container Orchestration using Kubernetes, Kubernetes vs Docker Swarm
Docker Swarm and Service Discovery	https://youtu.be/uKu0Ca99XUQ	Docker Swarm and Service Discovery
Introduction to Kubrnetes-Orchestration Tools	https://youtu.be/WUU85wXv4mA https://youtu.be/fO3vUMc_WbI	Introduction to Kubrnetes-Orchestration Tools

E-Book Links

Topics	URL	Note
Docker Orchestration	https://itrevolution.com/wp-content/uploads/2022/06/DOHB2_Excerpt.pdf	Complete Text book Discussed What is Docker Orchestration
Service Discovery	http://lsi.vc.ehu.es/pablogn/docencia/manuales/The%20Docker%20Book.pdf	Complete Text book Discussed Service Discovery