

Docker

What is Docker?

- Virtualization software
- Makes developing and deploying applications much easier
- Packages application with all the necessary dependencies, configuration, system tools and runtime.
 - o Containers – A standardized unit, that has everything the application needs to run.
- Portable artifact easily shared and distributed.

Development process before Containers?

- Each developer needs to install and configure all services directly on their OS on their local machine.
- Installation process different for each OS.
- Many steps where something can go wrong.
- If your app uses 10 services, each developer needs to install these 10 services.

Development process with Containers?

- Do not install directly on local machine
- Own isolated environment
- Postgres packaged with all dependencies and configs in own environment
- Start service as Docker containers using a single Docker command.
- Docker command is same for all services.
- Standardizes the process of running any service on any local dev environment.
- Much faster and easier to setup using docker.
- Easy to run different versions of same app without any conflicts.

Deployment process before containers?

- Development team produces artifact – like a zip file for the application and installation instructions. Also, database setup and installation.
- Give the package to operations teams, and if any error occurs then they would go back to development team, and communication for setup increases.
- Miscommunications can occur between both the teams.

Deployment with containers?

- Docker artifact includes everything the app uses.
- Instead of textual everything is packed inside the docker artifact.
- No configurations needed on the server.
- Less room for errors.
- Ops team just run the docker command to fetch and run the docker artifact.
- Install docker runtime on the servers before running docker on server.

Docker vs Virtual Machines?

- OS has two layers, OS kernel and OS application layer. The kernel communicates with the hardware.
- The kernel is the core of every operating system – it interacts between hardware and software components.
- Docker virtualizes the applications layers
 - o Contains the OS application layer
 - o Services and apps installed on top that layer
 - o Uses kernel of the host.
- VM virtualizes the complete OS – Application and kernel layer both
 - o So, each app comes with its OS
- Size of docker is much smaller than VMs.
 - o Dockers in MB
 - o VMs in GB
- Docker container takes seconds to start and VM takes minutes to start
- VM is compatible with all OS and docker is not compatible with all OS.
- Most containers are Linux based. Docker was built for Linux originally.
- Docker Desktop uses a hypervisor layer with a lightweight Linux distro, so makes lightweight apps possible to run on Windows and Mac.

Docker Engine

- A server with a long running daemon process – dockerd
- Manages images and containers

Docker CLI – Client

- CLI to interact with the docker server.
- Execute Docker commands to start/stop/... containers.

Docker Images vs Docker Containers

- Docker allows to package app in an artifact. Can be easily shared and moved like a zip or tarfile.
- Artifact – Docker Image.
 - o An executable application artifact
 - o Includes app source code, but also complete environment configuration.
 - o Has Application, any services needed and the OS layer
 - o Add environment variables, create directories, files, etc.
 - o Immutable template that defines how a container will be realized
- Docker container
 - o Starts the application
 - o A running instance of an image
 - o That's when the container environment is created.
- You can run multiple containers from 1 image.
- Commands (can run on terminal too)

- Docker images – gives list of images current on system
 - Docker ps – gives list of running containers.
- Run containers from images
- How do we get these images?
 - Docker registries
 - A storage and distribution system for docker images
 - Official images available from applications like Redis, Mongo, Postgres, etc.
 - Official images are maintained by the software authors or in collaboration with the docker community.
 - Docker hosts one of the biggest docker registries called docker hub
 - Docker hub – find and share images.
- Image tags – tells the version of the image we are using.

Pull an image?

- Command to pull an nginx image with 1.23 version tag
 - Docker pull nginx:1.23
- Don't give an image tag then pulls the latest version.

Run an image:

- Command to run an image:
 - Docker run nginx:1.23
 - In order to not have logs block the terminal use the detach flag
 - Docker run -d nginx:1.23
 - Now you can directly pull and run using the docker run command even if you do not have the image stored locally.
 - Run different versions of same application with docker

Port Binding

- Application inside container runs in an isolated docker network
- We need to expose the container port to the host local network and bind the port.
- Bind the containers port to the hosts port to make the service available to the outside world.
- Command:
 - Docker run -d -p 9000:80 nginx:1.23
 - -p is used to publish/bind the host
 - 9000 – localhost and 80 is container port
- Standard to use the same port on your host as container uses.

Start and Stop Containers

- Docker run – creates a new container every time and doesn't reuse the previous ones.
- In order to see all the containers whether created use command
 - Docker ps -a

- -a means list all or all
- Docker stops actively running container.
- Can start the container using docker start.
- You can also name the container using --name tag and then the name
 - Docker run --name web-app -d -p 9000:80 nginx:1.23

Public and Private Docker Registries

- Docker hub is largest public registry

Private:

- You need to authenticate before accessing the registry
- All big cloud providers offer private registries: Amazon ECR, Google Container Registry, etc.
- Nexus
- Docker Hub also has a private section.

Registry vs Repository

- Docker Registry
 - A service providing storage
 - Can be hosted by a third party, like AWS or by yourself.
 - Collection of repositories
- Docker Repository
 - Collection of related images with same name but different versions.

Create Own Images

- Companies create custom images for their application
- Build -> Image -> run -> container
- Docker file is a text document that contains commands to assemble an image
- Docker can then build an image by reading those instructions.
- Node.js -> Dockerfile

FROM

- Docker files must begin with a FROM instruction
- Build this image from the specified image

RUN

- Will execute any command in a shell inside the container environment.

COPY

- Copies file or directories from src and adds them to the filesystem of the container at the path destination.
- While RUN is execute COPY is executed on the host.

WORKDIR

- Sets the working directory for all the following commands, just like `cd`

CMD

- This instruction that is to be executed when Docker container starts
- There can only be one CMD instruction in the Docker file.

Making an image of the app

- Linux operating system
- Node and npm installed
- Copy application files from host into the container
- Executing `npm install` to install dependencies.

Build:

- Build the image using the following command
 - o `Docker build -t node-app:1.0 .`

Write Docker file -> Build Docker Image -> Run as Docker container.