**Docker Compose**

* Docker Compose is a “tool for defining and running your multi-container Docker applications”.
* Just need to download Docker Compose binary.
* Docker Compose is based on a docker-compose.yml file. This file defines all of the containers and settings you need to launch.
* Docker Compose supports all of the properties which can be defined using docker run.

Docker Compose to manage the orchestration of multi-container applications.

When working with multiple containers, it can be difficult to manage the starting along with the configuration of variables and links. To solve this problem, Docker has a tool called Docker Compose to manage the orchestration, of launching, of containers.

Docker Compose is based on a docker-compose.yml file. This file defines all of the containers and settings you need to launch

Docker Compose supports all of the properties which can be defined using docker run.

**Installation of Docker Compose**

$ sudo yum install -y docker

$ service docker start

$ sudo curl -L https://github.com/docker/compose/releases/download/1.24.1/docker-compose-`uname -s`-`uname -m` -o /usr/local/bin/docker-compose

$ sudo chmod +x /usr/local/bin/docker-compose

$ ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose

$ docker-compose –version

Simple three steps to use Docker Compose.

1. Define your app’s environment with a Dockerfile so it can be reproduced anywhere.
2. Define the services that make up your app in docker-compose.yml so they can be run together in an isolated environment.
3. Lastly, run docker-compose up and Compose will start and run your entire app.

**Lab Practice:**

Create Dockerfile and docker-compose.yaml

*vi Dockerfile*

FROM alpine

Maintaner Prakash

EXPOSE 4000

*vi docker-compose-yaml*

web:

 build: .

 links:

  - redis

 ports:

  - "4000"

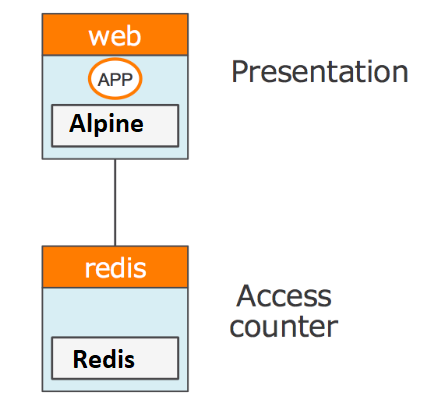
  - "8000"

redis:

 image: redis:alpine

 volumes:

  - /var/redis/data:/data



With the created docker-compose.yml file in place, you can launch all the applications with a single command of up. If you wanted to bring up a single container, then you can use up <name>.

*$ docker-compose up -d*

*$ docker-compose ps*

*$ docker-compose logs*

Docker Scale

*$ docker-compose scale web=2*

*$ docker-compose ps*

   Name          Command        State            Ports

---------------------------------------------------------------------------------------------------------

tutorial\_redis\_1  docker-entrypoint.sh redis ...  Up   6379/tcp

tutorial\_web\_1   npm start            Up   0.0.0.0:32700->4000/tcp, 0.0.0.0:32700->8000/tcp

tutorial\_web\_2   npm start            Up   0.0.0.0:32701->4000/tcp, 0.0.0.0:32701->8000/tcp

*$ docker-compose down*

to stop a set of containers you can use the command *docker-compose stop*

To remove all the containers use the command *docker-compose rm*