

**Lab Manual- Setup and Manage Docker**

**Compose for Multicontainer**

**Prepared for**:

**Date:** 18th Nov 2018

**Prepared by:** Bipin Sinhaa Document Name: Lab Manual **Document Number** DevOpsLab401

**Contributor:**

Shruti Sinhaa

**Table of Contents**

**1 OBJECTIVE ............................................................................................................3**

**2 PRE-REQUISISTE ..................................................................................................3**

**3 How Docker Compose Work .................................................................................3**

**4 Setup Up Docker Compose...................................................................................4**

4.1 **Manage Compose Conatiner** ............................................................................................... 6

**1 OBJECTIVE**

Docker composes

▪ Tool For Defining & Running Multi-Container Docker Applications

▪ Use Yaml Files To Configure Application Services (Docker-Compose.Yml)

▪ Can Start All Services With A Single Command : Docker Compose Up

▪ Can Stop All Services With A Single Command : Docker Compose Down

▪ Can Scale Up Selected Services When Required.

In This Lab will cover the basics of configuring docker compose with web and database services

**2 PRE-REQUISISTE**

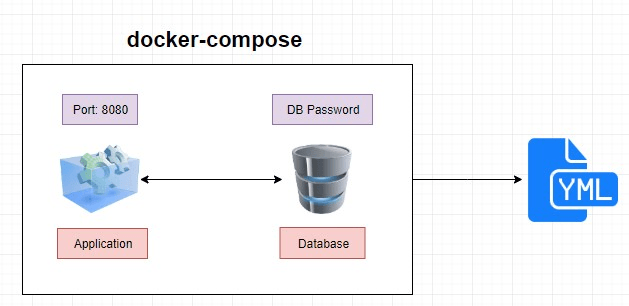
▪ Prior knowledge of Linux

▪ Accounts in Docker-Hub

▪ A local Computer with 4 CPU, 16 GB RAM, 200 GB disk space

**3 How Docker Compose Work**

▪ **Docker Compose** is a tool for defining and running multi-container docker applications. With Compose, we use a YAML file to configure our application’s services. And then we create and start all the services from the configuration with a single command. Here is a simple graphical illustration that shows how Docker compose works



**4 Setup Up Docker Compose**

***Steps 1:*** Check the docker compose version

$ docker-compose --version



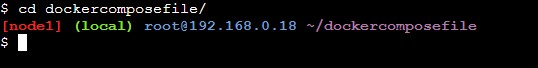
***Step 2:*** create a directory

$ mkdir dockercomposefile



***Step 3:*** go inside the directory

$ cd docekercomposefile/



***Steps 4:*** We are going to create simple docker compose file as an example. Here is the contents of my compose file.

vi docker-compose.yml



***Steps 5:*** Type below steps for docker compose

Version: ‘3’

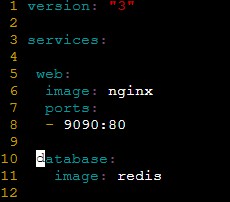
services:

web:

image: nginx

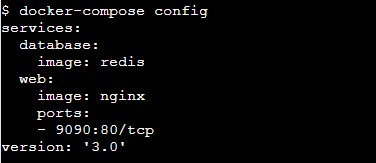
ports:

- 9090:80 database: image: redis



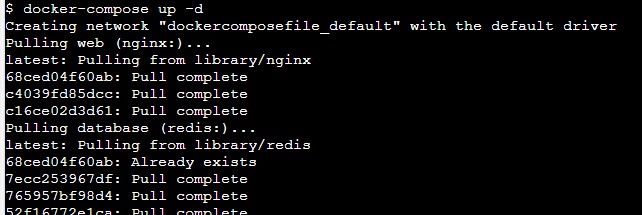
***Steps 6:*** Type below command to check everything in docker compose is correct

docker-compose config



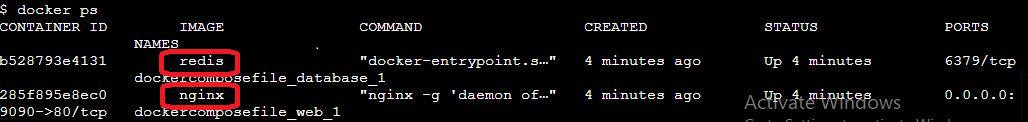
***Steps 7:*** Now let’s run the docker compose ( -d is for background)

docker-compose up -d



***Steps 8:*** Now let’s check the Docker ps

docker ps



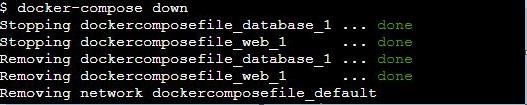
***Steps 9:*** Now open the browser and type the IP address with port 8080



4.1 **Manage Compose Conatiner**

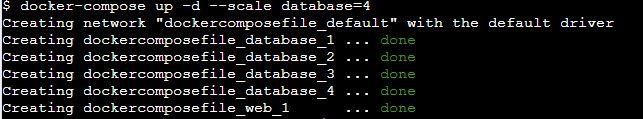
***Steps 1:*** Shut down the container

docker-compose down



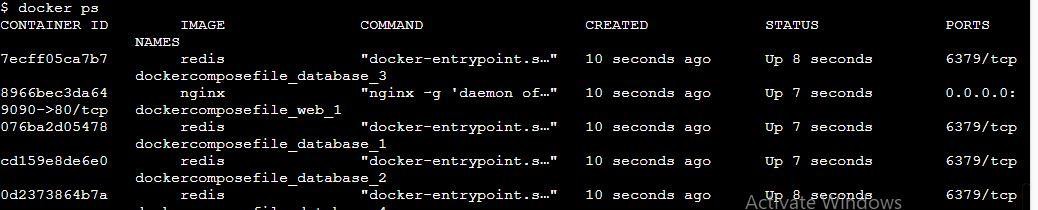
***Steps 2:*** scale the service like database

docker-compose up -d –scale database=4



***Steps 3:*** check the no of running container

docker ps



***Steps 4:*** down the container

Docker-compose down