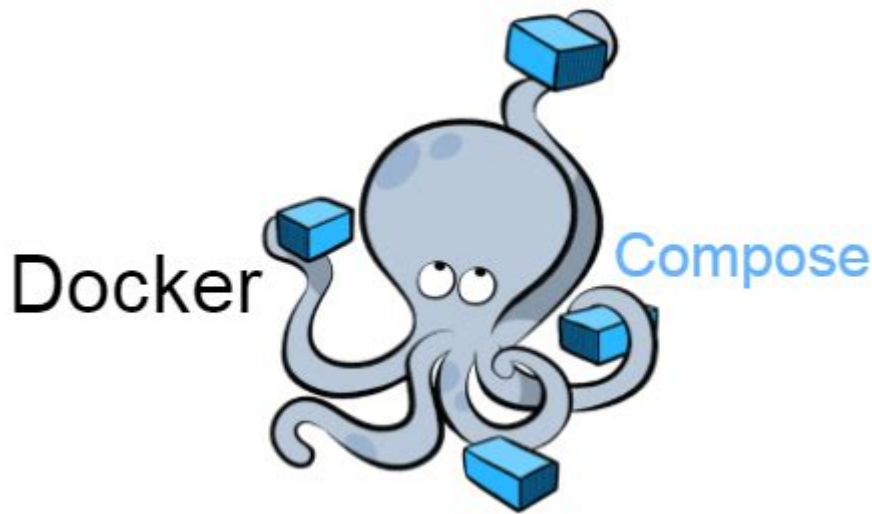


Docker Compose



Compose is a tool for defining and running multi-container Docker applications. With Compose, you use a YAML file to configure your application's services. Then, with a single command, you create and start all the services from your configuration.

I heavily utilize Docker Compose in my environment. I find it much easier to deploy Docker containers and, of greater importance, to document, update and make changes to my production containers and stacks.

I also use **Portainer** for some container management. However, I use it primarily for restarting or stopping containers and for development and testing purposes.

Installation

This guide assumes you are installing Docker Compose on Ubuntu 22.04 LTS.

To make sure you obtain the most updated stable version of Docker Compose, you'll download this software from its **official Github repository**.

You can confirm the latest version available in their **releases page**. The latest release at the time of this writing is version 2.16.0

Note: Starting with Docker Compose v2, Docker has migrated towards using the `compose` CLI plugin command, and away from the original `docker-compose`. The actual usage involves dropping the hyphen from `docker-compose` calls to become `docker compose`

1. Use the following command to download v2.16.0:

```
mkdir -p ~/.docker/cli-plugins/  
curl -SL https://github.com/docker/compose/releases/download/v2.16.0/docker-compose-linux-x86_64 -o  
~/.docker/cli-plugins/docker-compose
```

If the version has changed since this guide was written, simply replace "2.16.0" in the command above with the new version number.

2. Set the correct permissions so that the `docker compose` command is executable:

```
chmod +x ~/.docker/cli-plugins/docker-compose
```

3. Verify that the installation was successful by running the following:

```
docker compose version
```

The terminal should return the version of Docker Compose you selected.

Revision #20

Created 20 March 2023 23:21:08 by Tim

Updated 17 April 2023 16:08:19 by Tim