SMW 01 Install or Run External Image Linux
Tod Casasent
2020-09-18-1200

Introduction

These instructions are aimed at people familiar with Linux administration and GitHub, Docker and Docker-Compose, with root access to their Linux environment. Installation is for external users. The username within the Docker image is docker_tcga.

Target Operating System and Installation

These instructions were tested on Linux. These instructions with appropriate modifications should work as a basis for installing on other distributions. Perform the installs in the order given in this document. Some steps are dependent on previous steps. All elements of this document expect a Linux host. It may be possible to convert these instructions to Windows or OSX, but no such ability is supported.

Operating System Prerequisites

Docker and Docker-Compose are required installs.

Docker versions used in development and testing were:

Version: 17.12.0-ce or 17.06.0-ce

API version: 1.35 or 1.30

Go version: go1.9.2 or go1.8.3

Docker-Compose versions used in development and testing were:

docker-compose version 1.16.1 or 1.14.0

docker-py version: 2.5.1 or 2.3.0

CPython version: 2.7.13

Clone the GitHub Repository

Clone the GitHub repository with a shallow clone, since you will not be checking anything back in. This clone call grabs the newest version from master.

git clone --depth 1 https://github.com/MD-Anderson-Bioinformatics/StandardizedData.git

5.

Preparation from Clone

1.

Required: sed Docker-ComposeTemplate

Whether you plan to compile or use the provided image from Docker Hub, the Docker-Compose template in docker-build/StdMW needs to have certain updates made.

You need to set the outside port for the application by replacing <OUTSIDE_PORT> docker-compose_template.yml. Below, we use 8080.

The external read-only path to the files and data providing in the repository at data/testing_static/MW_CACHE is provided by replacing <CONFIGPATH>. Below, we use the path /ext/SMW/cache. (This means we copies the contents of data/testing_static/MW_CACHE into /ext/SMW/cache.)

Similarly, the writable external path to Tomcat logs (external for debugging and monitoring) is provided by replacing <LOGPATH>. Below, we use the path /ext/SMW/logs. Note, that this path needs to be reachable by root when docker starts and writeable by the user in the Dockerfile (id 2002 by default).

The image name is provided by replacing <IMAGETXT> with the name desired. For the Docker Hub image, use mdabcb/smw_image, as shown below.

The image tag is provided by replacing <DESIREDTAG> with the name desired. For the Docker Hub image, we use SMW_2020-09-11-1000, as shown below.

sed to make usable docker-compose.yml file

sed -e "s|<OUTSIDE_PORT>|8080|g" \

-e "s|<CONFIGPATH>|/ ext/SMW/cache|g" \

-e "s|<LOGPATH>|/ ext/SMW/logs|g" \

-e "s|<IMAGETXT>|mdabcb/smw_image|g" \

-e "s|<DESIREDTAG>|SMW_2020-09-11-1000|g" ./docker-compose_template.yml > ./docker-compose.yml

Optional: Compile

If you want to use pre-built images from Docker Hub, you can skip compiling. Before compiling, you need to sed the Dockerfile in docker-build/StdMW.

The release version is a label on the image. We provide this by replacing <RELEASE_VERSION> with 2020-09-11-1000, as shown below.

The user id needs to provide write and read access to the paths given in the docker-compose.yml file, as described below. By default, we replace <USERID> with 2002.

log path writes, so it needs the local user id

sed -e "s|<RELEASE_VERSION>|2020-09-11-1000|g" \

-e "s|<USERID>|2002|g" ./Dockerfile_template > ./Dockerfile

You can build a fresh local image after using:

docker-compose -f docker-compose.yml build --force-rm --no-cache

Starting and Stopping Docker

You can start the container using:

docker-compose -p extsmw -f docker-compose.yml up --no-build -d

You can stop the container using:

docker-compose -p extsmw -f docker-compose. yml down