Title: Running docker-compose inside a Google Cloud Engine

Post Body:

I'm trying to run a small docker-compose app inside a container-optimized Google Cloud Compute Engine node, but I'm getting stuck when it's trying to mount volumes during a `docker-compose up`:

```
Creating lightning_redis_1 ...  Creating lightning_db_1 ...  Creating lightning_redis_1 Creating lightning_db_1 ... done Creat
```

My docker-compose.yml file looks like this:

```
version: '3' services:   client:     build: ./client     volumes:       - ./client:/usr/src/app     ports:       - '4200:4200'
```

I don't want to have to change anything in the docker-compose.yml file - I'd prefer to be able to fix this issue by running commands inside the VM itself, or in how I set the VM up. Reason being is it's not my code and I can't change the docker-compose.yml file easily, and all I need to do is run it for a short period of time and execute a few docker-compose commands inside the VM.

Accepted Answer: None

Highest Rated Answer:

Container optimized OS usually mounts most of the paths as read-only. That is why you are getting the error

```
source path '/rootfs/home/jeremy/lightning': mkdir /rootfs: read-only file sys
```

So you have few options

Use named volumes in docker-compose

You will need to change your volumes like below

```
volumes:   - myappvol:/myapp
```

and define the top level volumes in compose

```
volumes:   myappvol: {}
```

As you said you don't want to modify the yaml then this may not work for you

Run docker-compose inside docker

Currently you run `docker-compose` on the main machine, instead you should use `docker-compose` inside another docker container which has the main root folder

```
docker run \     -v /var/run/docker.sock:/var/run/docker.sock \     -v '$PWD:/rootfs/$PWD' \     -w='/rootfs/$PWD' \     docke
```

This would work but the data would be persisted inside the docker container itself.

See below article for more details

https://cloud.google.com/community/tutorials/docker-compose-on-container-optimized-os