#!/bin/sh

sudo apt-get install nginx nginx-core #installing nginx

sudo apt-get install python3-certbot-nginx

sudo apt install certbot

# do the dhparam step and certbot

sudo certbot certonly --nginx -d "sregistry.idia.ac.za" --email "walter@idia.ac.za" --agree-tos --redirect

# copy certs to where the docker and minio expects them

# Docker

sudo cp /etc/letsencrypt/live/sregistry.idia.ac.za/fullchain.pem /etc/ssl/certs/chained.pem #mustn't be a dir

sudo cp /etc/letsencrypt/live/sregistry.idia.ac.za/privkey.pem /etc/ssl/private/domain.key

# Minio

sudo mkdir /etc/minio/

sudo mkdir /etc/minio/certs/

sudo cp /etc/letsencrypt/live/sregistry.idia.ac.za/fullchain.pem /etc/minio/certs/public.crt

sudo cp /etc/letsencrypt/live/sregistry.idia.ac.za/privkey.pem /etc/minio/certs/private.crt

# stop nginx on the host:

sudo systemctl stop nginx.service

# clone the sregistry repo master branch

git clone https://github.com/singularityhub/sregistry

# all these need to be run on sregistry dir

# configure settings/secrets

cd sregistry

# Add mozilla-django-oidc==3.0.0 in the sregistry/requirements (see oidc\_auth instructions for other configurations)

echo "mozilla-django-oidc==3.0.0">>requirements.txt

# build the docker containers inside the sregritry

docker build --network host -t ghcr.io/singularityhub/sregistry . # this builds the docker container from sregistry/Dockerfile

cd nginx # (sregistry/nginx)

docker build --network host -t ghcr.io/singularityhub/sregistry\_nginx . #this builds the docker container from sregistry/nginx/Dockerfile

cd ../ # back to sregistry dir

# copy out the https files

mkdir http

sudo mv nginx.conf http

sudo mv docker-compose.yml http

sudo cp https/docker-compose.yml .

# spin up the services: docker-compose up can be done after OIDC config

New\_secret: o#\_!p%qv^1dxkgcl1@c%g42s82cdd(3tl&+md08z\*tlhn26zu0

CA setup: https://arminreiter.com/2022/01/create-your-own-certificate-authority-ca-using-openssl/