

```
[*] Building 343.7s (27/27) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 32B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/ubuntu:focal
=> [internal] load build context
=> => transferring context: 2.57kB
=> [ 1/22] FROM docker.io/library/ubuntu:focal@sha256:450e06588f42ebe1551f3b1a535034b0aa46cd936fe7f2c6b0d72997ec61dbd
=> CACHED [ 2/22] RUN apt-get -qq update --fix-missing && apt-get -qq --yes upgrade
=> CACHED [ 3/22] RUN DEBIAN_FRONTEND=noninteractive apt-get install -y software-properties-common g++ make cmake wget git bzip2 apache2 curl apache
=> CACHED [ 4/22] RUN apt-get install -y --fix-missing --no-install-recommends libxml2-dev libxslt1-dev libfribidi-dev libcairo2-dev
=> CACHED [ 5/22] RUN apt-get install -y libgdal-dev
=> CACHED [ 6/22] RUN apt-get install -y php7.4-fpm libapache2-mod-php7.4 php7.4-common php7.4-cli php7.4 php7.4 php7.4-opcache php7.4-gd php7.4-
=> CACHED [ 7/22] ADD resources /tmp/resources
=> [ 8/22] ADD setup.sh /setup.sh
=> [ 9/22] RUN chmod 0755 /setup.sh
=> [10/22] RUN /setup.sh
=> [11/22] RUN cp /tmp/resources/000-default.conf /etc/apache2/sites-available/
=> [12/22] RUN wget http://mirrors.kernel.org/ubuntu/pool/multiverse/liba/libapache2-mod-fastcgi/libapache2-mod-fastcgi_2.4.7-0910052141-1.2_amd64.deb
=> [13/22] RUN cp /tmp/resources/php7-fpm.conf /etc/apache2/conf-available/
=> [14/22] RUN a2enmod actions cgi alias proxy_fcgi fastcgi headers
=> [15/22] RUN a2enconf php7.4-fpm
=> [16/22] RUN chmod o+x /usr/local/bin/mapserv
=> [17/22] RUN ln -s /usr/local/bin/mapserv /usr/lib/cgi-bin/mapserv
=> [18/22] RUN chmod 755 /usr/lib/cgi-bin
=> [19/22] RUN wget https://github.com/jwilder/dockerize/releases/download/v0.6.1/dockerize-linux-amd64-v0.6.1.tar.gz && tar -C /usr/local/bin -x
=> [20/22] RUN apt-get install -y net-tools
=> [21/22] RUN mv /usr/local/lib/libcurl.so.4.4.0 /usr/local/lib/libcurl.so.4.4.0.backup
=> [22/22] RUN apt-get clean && rm -rf /var/lib/apt/lists/* /tmp/* /var/tmp/*
=> exporting to image
=> => exporting layers
=> => writing image sha256:c86ead2f7d5b675908dae4a23169fe95c17a100061d312c83aa13c17a0d20f55
=> => naming to docker.io/kartoza/mapserver_kartoza
```

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

```
D:\Studia\Semestr_7\Bazy_Danych_Przestrzennych\mapserver\docker-mapserver>docker run -d -p 8182:80 --name mapserver2 -v D:\Studia\Semestr_7\Bazy_Danych_Przestrzennych\mapserver\docker-mapserver\map:/map kartoza/mapserver_kartoza
3275e13ace9fc5dc026408a1093d67ea020c7f524a74c4e10cb7ba47ea0e0ce
D:\Studia\Semestr_7\Bazy_Danych_Przestrzennych\mapserver\docker-mapserver>docker exec -it mapserver2 /bin/bash
root@3275e13ace9f:/#
```

```
root@3275e13ace9f:/# psql postgresql://postgres:postgres@host.docker.internal/mapserver
psql (12.12 (Ubuntu 12.12-0ubuntu0.20.04.1), server 14.5)
WARNING: psql major version 12, server major version 14.
Some psql features might not work.
Type "help" for help.

mapserver=# \dt
          List of relations
 Schema |      Name      | Type | Owner
-----+-----+-----+-----
 public | airports       | table | postgres
 public | spatial_ref_sys | table | postgres
(2 rows)

mapserver=#
```

```
root@3275e13ace9f:/map# cat dem.map
MAP
  NAME 'map'
  SIZE 800 650
  STATUS ON
  EXTENT -58968 145487 30916 206234
  UNITS METERS

  WEB
    METADATA
      'wms_title' 'Terrian wms'
      'wms_srs' 'EPSG: 3763 EPSG: 4326 EPSG: 3857'
      'wms_enable_request' '*'
      'wms_onlineresource' 'http://54.37.13.53/mapservices/srtm'
    END
  END

  PROJECTION
    'init=epsg:3763'
  END

  LAYER
    NAME srtm
    TYPE raster
    STATUS OFF
    DATA "PG:host=host.docker.internal port=5432 dbname='cwiczenia_6' user='postgres' password='postgres' schema='rasters' table='dem' mode='2'"
    PROCESSING "SCALE=AUTO"
    PROCESSING "MODATA=-32767"
    OFFSITE 0 0
    METADATA
      'wms_title' 'srtm'
    END
  END
END
root@3275e13ace9f:/map#
```

