

docker-compose error "read-only file system"

Asked 6 years, 4 months ago Modified 1 year, 9 months ago Viewed 30k times



I designed a `docker-compose.yml` file that also supposed to work with individual volumes.

13

I created a raid-drive which is mounted as `/dataraid` to my system. I can read/write to the system, but when using it in my compose file, I get `read-only file system` error messages.



Adjusting the volumes to a other path like `/home/myname/test` the compose file works.



I have no idea what the `/dataraid` makes it "read-only".



What are the permissions settings a compose file needs?

error message:

```
ERROR: for db Cannot start service db: error while creating mount source path
'/dataraid/nextcloud/mariadb': mkdir /dataraid: read-only file system
```

compose:

```
version: '3'
services:
  db:
    image: mariadb
    command: --transaction-isolation=READ-COMMITTED --binlog-format=ROW
    restart: always
    volumes:
      - /dataraid/nextcloud/mariadb:/var/lib/mysql
    environment:
      - MYSQL_ROOT_PASSWORD=PASSWORD
    env_file:
      - db.env
  redis:
    image: redis
    restart: always
  app:
    image: nextcloud:fpm
    restart: always
    volumes:
      - /dataraid/nextcloud/html:/var/www/html
    environment:
      - MYSQL_HOST=db
    env_file:
      - db.env
    depends_on:
      - db
      - redis
  web:
    build: ./web
    restart: always
```

```
volumes:
  - /dataaid/nextcloud/html:/var/www/html:ro
environment:
  - VIRTUAL_HOST=name.de
  - LETSENCRYPT_HOST=name.de
  - LETSENCRYPT_EMAIL=x@y.de
depends_on:
  - app
ports:
  - 4080:80
networks:
  - proxy-tier
  - default
collabora:
  image: collabora/code
  expose:
    - 9980
  cap_add:
    - MKNOD
  environment:
    - domain=name.de
    - VIRTUAL_HOST=name.de
    - VIRTUAL_PORT=9980
    - VIRTUAL_PROTO=https
    - LETSENCRYPT_HOST=name.de
    - LETSENCRYPT_EMAIL=x@y.de
    - username=          #optional
    - password=         #optional
  networks:
    - proxy-tier
  restart: always
cron:
  build: ./app
  restart: always
  volumes:
    - /dataaid/nextcloud/html:/var/www/html
  entrypoint: /cron.sh
  depends_on:
    - db
    - redis
proxy:
  build: ./proxy
  restart: always
  ports:
    - 443:443
    - 80:80
  environment:
    - VIRTUAL_PROTO=https
    - VIRTUAL_PORT=443
  labels:
    com.github.jrcs.letsencrypt_nginx_proxy_companion.nginx_proxy: "true"
  volumes:
    - /dataaid/nextcloud/nginx-certs:/etc/nginx/certs:ro
    - /dataaid/nextcloud/nginx-vhost.d:/etc/nginx/vhost.d
    - /dataaid/nextcloud/nginx-html:/usr/share/nginx/html
    - /dataaid/nextcloud/nginx-conf.d:/etc/nginx/conf.d
    - /var/run/docker.sock:/tmp/docker.sock:ro
  networks:
    - proxy-tier
letsencrypt-companion:
  image: jrcs/letsencrypt-nginx-proxy-companion
  restart: always
```

```

volumes:
  - /dataraid/nextcloud/nginx-certs:/etc/nginx/certs
  - /dataraid/nextcloud/nginx-vhost.d:/etc/nginx/vhost.d
  - /dataraid/nextcloud/nginx-html:/usr/share/nginx/html
  - /var/run/docker.sock:/var/run/docker.sock:ro
networks:
  - proxy-tier
depends_on:
  - proxy
networks:
  proxy-tier:

```

see error messages:

```

bernd@sys-dock:/dataraid/Docker-Configs/nextcloud$ docker-compose up -d
Creating network "nextcloud_default" with the default driver
Creating network "nextcloud_proxy-tier" with the default driver
Creating nextcloud_db_1 ...
Creating nextcloud_proxy_1 ... error
Creating nextcloud_db_1 ... error
Creating nextcloud_collabora_1 ...
ERROR: for nextcloud_proxy_1 Cannot start service proxy: error while creating
mount source path '/dataraid/nextcloud/nginx-certs': mkdir /dataraid: read-only
file system
Creating nextcloud_redis_1 ... done
Creating nextcloud_collabora_1 ... done
ERROR: for proxy Cannot start service proxy: error while creating mount source
path '/dataraid/nextcloud/nginx-certs': mkdir /dataraid: read-only file system
ERROR: for db Cannot start service db: error while creating mount source path
'/dataraid/nextcloud/mariadb': mkdir /dataraid: read-only file system
ERROR: Encountered errors while bringing up the project.

```

docker

docker-compose

Share Follow

edited Jun 23, 2022 at 2:30



ggorlen

57.5k ● 8 ● 112 ● 155

asked Nov 4, 2018 at 18:08



speedAmaster

131 ● 1 ● 1 ● 4

Can you post your `docker-compose.yml` file? – searsaw Nov 4, 2018 at 18:33

Doesn't sound like anything to do with docker, more to do with how that drive is configured / mounted. Can you create regular files on the mount (without using Docker)? – johnharris85 Nov 4, 2018 at 18:35

see my docker-compose.yml – speedAmaster Nov 4, 2018 at 19:38 ✎

@johnharris85: that is the strange part. I can touch and modify files in that mounted drive...

– speedAmaster Nov 4, 2018 at 19:48

5 Did you mount the dataraid filesystem after docker started? If so, try restarting the docker daemon (`systemctl restart docker`) – BMitch Nov 5, 2018 at 23:39

2 Answers

Sorted by: Highest score (default)



22

If docker starts before the filesystem gets mounted, you could be seeing issues with the docker engine trying to write to the parent filesystem. You can restart the docker daemon to rule this out (`systemctl restart docker` in systemd base environments).



If restarting the daemon helps, then you can add a dependency between the docker engine and the external filesystem mounts. In systemd, that involves an `After=` clause in the unit file. E.g. you could create a `/etc/systemd/system/docker.service.d/override.conf` file containing:



```
[Unit]
After=nfs-client.target
```

(Note that I'm not sure that `nfs-client.target` is the correct unit file for your filesystem, you'll want to check where it gets mounted.)

Another issue I've seen people encounter recently is Snap based docker installs, which run docker inside of another container technology, which would prevent access to paths not explicitly configured in the Snap.

[Share](#) [Follow](#)

answered Mar 23, 2020 at 14:43

[BMitch](#)

266k ● 50 ● 542 ● 500

1 That was exactly my issue, had used snap. – [Jesper Kristiansen](#) Oct 18, 2022 at 20:32

The problem was resolved just after re-installing the docker. In my case, the shared storage had been mounted after docker installation. – [ArashMad](#) Mar 23, 2023 at 16:33



0

For me, changing `:ro` to `:rw` in my `docker-compose.yml` was the key to enabling write permissions.



In your case (for example--I'm not sure this is your issue):



```
web:
  build: ./web
  restart: always
  volumes:
    - /dataraid/nextcloud/html:/var/www/html:ro # change to :rw
```

With `:ro` on my case, I saw:

```
$ docker exec --privileged -it x_api bash
root@api# touch foo
```

```
touch: cannot touch 'foo': Read-only file system
```

But after using `:rw` and restarting the container, `touch` worked.

Share Follow

answered May 26, 2023 at 21:29



ggorlen

57.5k ● 8 ● 112 ● 155

Start asking to get answers

Find the answer to your question by asking.

[Ask question](#)

Explore related questions

`docker`

`docker-compose`

See similar questions with these tags.