

docker: Error response from daemon: failed to create shim task: OCI runtime create failed: runc create failed: unable to start container process

Asked 2 years, 3 months ago Modified 1 year, 11 months ago Viewed 265k times



45



I am trying to build this dockerfile and then run it but I'm getting this error `docker: Error response from daemon: failed to create shim task: OCI runtime create failed: runc create failed: unable to start container process: exec: "/deployment-service": permission denied: unknown.`

This is my docker file, I've created the volumes and networks

```
FROM golang:1.19.2-alpine as builder

RUN apk add bash

RUN apk add --no-cache openssh-client ansible git

RUN mkdir /workspace
WORKDIR /workspace

COPY go.mod ./
COPY go.sum ./

RUN go mod download

COPY . ./

RUN go build -o deployment-service cmd/deployment-service/main.go

FROM alpine:latest
RUN apk --no-cache add ca-certificates
WORKDIR /root/

COPY --from=builder /workspace .

ARG DEFAULT_PORT=8080
ENV PORT $DEFAULT_PORT

EXPOSE $PORT

CMD ["/deployment-service"]
```

this is my run command,

```
docker run --name=${CONTAINER_NAME} -d --rm -p ${PORT}:80 -e
DEPLOYMENT_SERVICE_DATABASE_CONNECTION_URI=mongodb://${MONGO_USERNAME}:${MONGO_PAS
-e DEPLOYMENT_SERVICE_SERVER_SECRET_KEY=${SECRET_KEY} -e ANSIBLE_CONFIG='./jam-
ansible/ansible.cfg' -e
```

```
DEPLOYMENT_SERVICE_ANSIBLE_SUBMISSION_ROOT=${DEPLOYMENT_ROOT} -v
${DEPLOYMENT_VOLUME}:${DEPLOYMENT_ROOT} --network=${NETWORK_NAME} server:latest
```

help to get my issue solved.

docker go

Share Improve this question Follow

asked Jan 11, 2023 at 5:40



Deeksha Sharma

619 ● 1 ● 4 ● 8

Does the same `go build ...; ./deployment-service` sequence work without Docker? Does specifying a package name and not just a single file name work better, `go build/cmd/deployment-service`? – [David Maze](#) Jan 11, 2023 at 12:07

yeah, it's working. I mistakenly deleted the the binary from the container once after that I started facing this issue. – [Deeksha Sharma](#) Jan 11, 2023 at 13:42 ✎

...in the `docker run` command, you have a `-v` option hiding a `$DEPLOYMENT_ROOT` directory in the container. If that's the image's `/root` directory, it will definitely cause this problem; delete that option to use the binary built into the image. (`docker build` ing an updated image shouldn't be much more expensive than `go build` ing a binary; you do not need a volume mount to inject code.) – [David Maze](#) Jan 11, 2023 at 14:01

No that's not the image's root directory, it's some data in the container that I want to keep. Also, I solved the error message when I changed this name of binary to something else. As I said earlier the issue started after I deleted the binary inside the container, but in my knowledge if a container is removed everything related to it is removed, so if I make a new container after removing the container(from which I deleted the binary) it should generate the binary again and should work fine, but it's not happening. – [Deeksha Sharma](#) Jan 11, 2023 at 14:29

7 A REALLY dumb way for this to happen is that you are specifying the docker run options in the wrong order. This will show the error: 'docker run <container-name> -p 8000:8000' ... but this will work 'docker run -p 8000:8000 <container-name>' I wasted 30 minutes on this stupidity. Docker 'run' really wants the image name as the last parameter, regardless of the convention for optional parameters (at least on Windows where I hit this issue and google lead me here!). – [Peter Walkley](#) Apr 28, 2023 at 13:43 ✎

5 Answers

Sorted by: Highest score (default) ▾



It seems lack of permission to execute **deployment-service**. You can add

14

```
RUN chmod +x deployment-service
```



before `CMD ["/deployment-service"]`



[Share](#) [Improve this answer](#) [Follow](#)

answered Feb 15, 2023 at 15:19

[pusponerow](#)

196 ● 4



Sometimes you may end up getting this error because you have not specified an entrypoint for your container. You can do that using `ENTRYPOINT` inside the `Dockerfile`

6

Below is a .NET example:



```
ENTRYPOINT ["dotnet", "application.dll"]
```

[Share](#) [Improve this answer](#) [Follow](#)

answered Feb 15, 2023 at 14:56

[GeralexGR](#)

3,602 ● 6 ● 26 ● 35

1 Just adding an entrypoint `"/bin/bash"` solved it for me. – [JoeFox](#) Nov 4, 2023 at 6:29



looks like you need to edit the permissions of the deployment-service file.

3

try running



```
chmod a+x ./deployment-service
```



in the terminal then trying again.

[Share](#) [Improve this answer](#) [Follow](#)

answered Jan 26, 2023 at 16:42

[Jacob Kearney](#)

431 ● 2 ● 12



In my case, I got the below error while trying to run the container using the **command**:

2

```
docker container run -p portnumber:portnumber amazoncorretto:17-alpine-jdk
```



ERROR:



```
docker: Error response from daemon: failed to create shim task: OCI runtime create failed: runc create failed: unable to start container process: exec: "bash": executable file not found in $PATH: unknown.
```

- I used `amazoncorretto:17-alpine-jdk` docker image and in `dockerfile` instead of `bash`, I changed it to `sh` and it worked as `bash` didn't work with `alpine`.

Command changed from:

```
CMD ["bash", "-c", "java -jar test.jar"]
```

to:

```
CMD ["sh", "-c", "java -jar test.jar"]
```

Share Improve this answer Follow

edited Feb 17, 2023 at 8:40

answered Feb 13, 2023 at 18:36



helvete

2,683 ● 15 ● 36 ● 41



veena dega

31 ● 4



-8



Run ***docker system prune -a*** clear all cache and stopped containers and then try starting the docker. It'll start.

Share Improve this answer Follow

answered May 23, 2023 at 3:12



Nikhil Vijayan

13 ● 1

8 This is excessive. You are removing all networks, volumes etc. – pierpy May 27, 2023 at 9:32

- 1 This might delete images, so **do not** run this command unless you don't mind your Docker images being wiped! While in some cases clearing the cache might solve some issues, `prune` with the `-a` option deletes unused images, so any Docker image that is not currently running in a container might get deleted. If you want to wipe stopped containers, unused networks, dangling images and unused build cache, but keep your Docker images, run `docker system prune` (without `-a`). – Adrian May 29, 2024 at 14:28

Start asking to get answers

Find the answer to your question by asking.

Ask question

Explore related questions

docker go

4/30/25, 8:16 PM

go - docker: Error response from daemon: failed to create shim task: OCI runtime create failed: runc create failed...

See similar questions with these tags.