

How to extend nginx docker image without getting error systemctl: command not found?

Asked 5 years, 5 months ago Modified 5 years, 5 months ago Viewed 3k times



I want to build my own custom docker image from nginx image.

1

I override the ENTRYPOINT of nginx with my own ENTRYPOINT file.



Which bring me to ask two questions:

1. I think I lose some commands from nginx by doing so. am I right? (like expose the port..)
2. If I want to restart the nginx I run this commands: `nginx -t && systemctl reload nginx`. but the output is:

```
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

```
/entrypoint.sh: line 5: systemctl: command not found
```

How to fix that?

```
FROM nginx:latest

WORKDIR /

RUN echo "deb http://ftp.debian.org/debian stretch-backports main" >>
/etc/apt/sources.list

RUN apt-get -y update && \
    apt-get -y install apt-utils && \
    apt-get -y upgrade && \
    apt-get -y clean

# I ALSO WANT TO INSTALL CERBOT FOR LATER USE (in my entrypoint file)

RUN apt-get -y install python-certbot-nginx -t stretch-backports

# COPY ./something ./tothisimage
# COPY ./something ./tothisimage
# COPY ./something ./tothisimage
# COPY ./something ./tothisimage

COPY entrypoint.sh /entrypoint.sh

ENTRYPOINT ["bash", "/entrypoint.sh"]
```

entrypoint.sh

```
echo "in entrypoint"

# I want to run some commands here...
```

```
# After I want to run nginx normally....

nginx -t && systemctl reload nginx

echo "after reload"
```

`docker` `nginx`

Share Improve this question Follow

asked Oct 17, 2019 at 9:26



Jon Sud

11.7k ● 31 ● 103 ● 228

3 Answers

Sorted by: Highest score (default) ▾



Commands like `service` and `systemctl` mostly just don't work in Docker, and you should totally ignore them.

2



At the point where your entrypoint script is running, it is literally the only thing that is running. That means you don't need to restart nginx, because it hasn't started the first time yet. The standard pattern here is to use the entrypoint script to do some first-time setup; it will be passed the actual command to run as arguments, so you need to tell it to run them.



```
#!/bin/sh

echo "in entrypoint"
# ... do first-time setup ...

# ...then run the command, nginx or otherwise
exec "$@"
```

(Try running `docker run --rm -it myimage /bin/sh`. You will get an interactive shell in a new container, but *after* this first-time setup has happened.)

The one thing you do lose in your Dockerfile is the default `CMD` from the base image (setting an `ENTRYPOINT` resets that). You need to add back [that CMD](#):

```
COPY entrypoint.sh /entrypoint.sh
RUN chmod +x /entrypoint.sh
ENTRYPOINT ["/entrypoint.sh"]
CMD ["nginx", "-g", "daemon off;"]
```

You should keep the other settings from the base image, like `ENV` definitions and `EXPOSE` d ports.

Share Improve this answer Follow

answered Oct 17, 2019 at 10:47



David Maze

161k ● 46 ● 249 ● 289



this will work using `service` command:

1

```
echo "in entrypoint"

# I want to run some commands here...

# After I want to run nginx normally...

nginx -t && service nginx reload

echo "after reload"
```



output:

```
in entrypoint
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
Restarting nginx: nginx.
after reload
```

Share Improve this answer Follow

answered Oct 17, 2019 at 9:36



LinPy

18.6k ● 5 ● 46 ● 63



The "systemctl" command is specific to some SystemD based operating system. But you do not have such a SystemD daemon running on PID 1 - so even if you install those packages it wont work.

1



You can only check in the nginx.service file which command the "reload" would execute for real. Or have something like the [docker-systemctl-replacement](#) script do it for you.



Share Improve this answer Follow

answered Oct 17, 2019 at 18:24



Guido U. Draheim

3,296 ● 1 ● 22 ● 20

Start asking to get answers

Find the answer to your question by asking.

Ask question

Explore related questions

[docker](#)[nginx](#)

See similar questions with these tags.