How can I run script automatically after Docker container startup

Asked 7 years, 2 months ago   Modified 1 year ago   Viewed 162k times

▲

32

▼

🔖

🕘

I'm using Search Guard plugin to secure an elasticsearch cluster composed of multiple nodes. Here is my Dockerfile:

```sh
#!/bin/sh
FROM docker.elastic.co/elasticsearch/elasticsearch:5.6.3

USER root

# Install search guard
RUN bin/elasticsearch-plugin install --batch com.floragunn:search-guard-
5:5.6.3-16 \
    && chmod +x \
        plugins/search-guard-5/tools/hash.sh \
        plugins/search-guard-5/tools/sgadmin.sh \
        bin/init_sg.sh \
    && chown -R elasticsearch:elasticsearch /usr/share/elasticsearch

USER elasticsearch
```

To initialize SearchGuard (create internal users and assign roles). I need to run the script `init_sg.sh` after the container startup. Here is the problem: Unless elasticsearch is running, the script will not initialize any security index.

```
                                                                  g/ -ts config/truststore.jks -
```

artup but since I'm running it on
ed automatically for some reason. In this
r the container startup!

eally appreciated.

ch-plugin

Share Improve this question Follow

PhiloJunkie
1,169 ● 4 ● 13 ● 27

---

1   You will only need to initialize SG once per cluster. Re-initialize SG on every restart of a pod will overwrite any custom roles/permissions not saved. – nafooesi Sep 11, 2019 at 22:51

@nafooesi Absolutely, you are right! The SG should be initialized only once per cluster. – PhiloJunkie Sep 15, 2019 at 18:17

This question is also addressed here – ManishM Nov 20, 2020 at 19:17

---

8 Answers

Sorted by: Highest score (default) ⇕

▲

30

▼

The image itself has an entrypoint `ENTRYPOINT ["/run/entrypoint.sh"]` specified in the Dockerfile. You can replace it by your own script. So for example create a new script, mount it and first call `/run/entrypoint.sh` and then wait for start of elasticsearch before running your `init_sg.sh`.

Share Improve this answer Follow

edited Sep 4, 2018 at 17:34   answered Dec 6, 2017 at 10:18

caub   **Jonathan Lechner**
2,769 ● 2 ● 30 ● 32   **581** ● 4 ● 9

---

3   I'm trying to do the exact same thing, but the `/run/entrypoint.sh` is not present in the container. I can't find where it sits. Any help? – RedGiant May 25, 2018 at 12:37 ✎

Found it at /entrypoint.sh for elasticsearch:1.5.2 – Marieke Jun 24, 2021 at 16:50

7, 2021 at 19:41

e container statically. If you change the script
an make a script which is called from
r 12, 2023 at 9:51

ck my repo's `Dockerfile`

age and in the `Dockerfile` I wrote `CMD`
in `run.sh` and write `CMD ["run.sh"]`.

```
FROM java:8

RUN apt-get update && apt-get install stress-ng -y
ADD target/restapp.jar /restapp.jar
COPY dockerrun.sh /usr/local/bin/dockerrun.sh
RUN chmod +x /usr/local/bin/dockerrun.sh
CMD ["dockerrun.sh"]
```

dockerrun.sh

```
#!/bin/sh
java -Dserver.port=8095 -jar /restapp.jar &
hostname="hostname: `hostname`"
nohup stress-ng --vm 4 &
while true; do
  sleep 1000
done
```

Share Improve this answer Follow   edited Dec 6, 2017 at 10:25   answered Dec 6, 2017 at 10:20

veerendra2
2,273 ● 7 ● 37 ● 65

2   thanks for the reply, the elasticsearch image has already an entrypoint.sh file, using CMD like this will
    override it and elasticsearch search won't start at all :/ – PhiloJunkie Dec 6, 2017 at 15:09

▲

7

This is addressed in the documentation here: https://docs.docker.com/config/containers/multi-service_container/

If one of your processes depends on the main process, then start your helper process FIRST
                                              SECOND and remove the fg %1 line.

...kground

...wait on the
...k and returns

...foreground

Share Improve this answer Follow

answered Nov 20, 2020 at 19:19

ManishM
137 ● 2 ● 3

duplicated here stackoverflow.com/questions/45371521/… – ManishM Nov 20, 2020 at 19:19

▲

5

▼

I was trying to solve the exact problem. Here's the approach that worked for me.

1. Create a separate shell script that checks for ES status, and only start initialization of SG when ES is ready:

Shell Script

```sh
#!/bin/sh

echo ">>>>  Right before SG initialization <<<<"
# use while loop to check if elasticsearch is running
while true
do
    netstat -uplnt | grep :9300 | grep LISTEN > /dev/null
    verifier=$?
    if [ 0 = $verifier ]
        then
            echo "Running search guard plugin initialization"
            /elasticsearch/plugins/search-guard-6/tools/sgadmin.sh -h 0.0.0.0 -
cd plugins/search-guard-6/sgconfig -icl -key config/client.key -cert
config/client.pem -cacert config/root-ca.pem -nhnv
            break
        else
            echo "ES is not running yet"
            sleep 5
    fi
```

ssible after it starts.

By clicking "Accept all cookies", you agree Stack Exchange can store cookies on your device and disclose information in accordance with our Cookie Policy.

your ES image. So that it starts the ss.

```
/sginit.sh &
```

This way the sginit.sh will start in the background, and will only initialize SG after ES is started.

The reason to have this sginit.sh script starts before ES in the background is so that it's not blocking ES from starting. The same logic applies if you put it after starting of ES, it will never run unless you put the starting of ES in the background.

Share Improve this answer Follow

edited Jun 20, 2020 at 9:12   answered Apr 17, 2019 at 2:50

Community Bot   nafooesi
1 ● 1   1,267 ● 13 ● 18

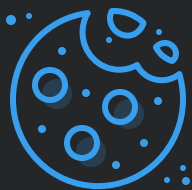How to wait for the ES get started, then initialize the SG? – Akira Sep 10, 2019 at 21:20

1   As commented below the original question, you do NOT want to initialize SG every single time. It will
    overwrite any modified config when a pod restarts. SG should only be initialized once per cluster at the
    beginning. If you need to run some other script after ES starts, my script can be used as an example to
    detect if ES is running in the background before kicking off the script. – nafooesi Dec 3, 2019 at 1:23

The general approach described here is very useful. I had a similar problem (not with ES), where I wanted
to run a script after the main process had a chance to start up. This pattern worked like a charm! – bischoje
Feb 25, 2020 at 20:07

▲

3

▼

You can also use wait-for-it script. It will wait on the availability of a host and TCP port. It is useful for synchronizing the spin-up of interdependent services and works like a charm with containers. It does not have any external dependencies so you can just run it as an RUN command without doing anything else.

A Dockerfile example based on this thread:

```
                              is not declared as a volume in

                              search /data && echo
                              && echo 'path.data: /data' >>


                              ait-for-
                              for-it.sh /utils/wait-for-


                              ipt


                              pid & /bin/bash /utils/wait-
                              ript.sh; kill $(cat
```

answered Oct 17, 2020 at 21:37

encodedwitch
31 ● 1

---

wait-for-it from master branch: raw.githubusercontent.com/vishnubob/wait-for-it/master/...
– Damien Golding Jul 25, 2023 at 11:03

▲

I would suggest to put the **CMD** in you docker file to execute the script when the container start

2

```
FROM debian
RUN apt-get update && apt-get install -y nano && apt-get clean
EXPOSE 8484
CMD ["/bin/bash", "/opt/your_app/init.sh"]
```

▼

There is other way , but before using this look at your requirement,

```
    ENTRYPOINT "put your code here" && /bin/bash
    #exemple ENTRYPOINT service nginx start && service ssh start &&/bin/bash
"use && to separate your code"
```

answered Dec 6, 2017 at 10:37

Sohan
6,809 ● 6 ● 40 ● 58

---

this solution overrides the entrypoint which is provided by Elasticsearch Image – PhiloJunkie Dec 6, 2017 at 15:06

I have no looked into elasticseacrh image installation as such, do you mean this does not work for you?
Sohan Dec 7, 2017 at 6:05

---

execute tasks like initialization (...)
operate in "The Docker Way"

w to install etc. which I won't repeat here.

example how to run a process + a script
heir docs.

Say we want to run `nginx` (or any process that runs until end of container lifetime) plus some shell script `myscript.sh`.

Local directory structure:

```
./Dockerfile
./myscript.sh
./s6-overlay/s6-rc.d/myapp/type
./s6-overlay/s6-rc.d/myapp/up
./s6-overlay/s6-rc.d/user/contents.d/myapp
```

`Dockerfile`:

```
FROM ubuntu
ARG S6_OVERLAY_VERSION=3.1.4.1

RUN apt-get update && apt-get install -y nginx xz-utils
RUN echo "daemon off;" >> /etc/nginx/nginx.conf

# Minimal set of dependecies required for s6
ADD https://github.com/just-containers/s6-
overlay/releases/download/v${S6_OVERLAY_VERSION}/s6-overlay-noarch.tar.xz /tmp
RUN tar -C / -Jxpf /tmp/s6-overlay-noarch.tar.xz
ADD https://github.com/just-containers/s6-
overlay/releases/download/v${S6_OVERLAY_VERSION}/s6-overlay-x86_64.tar.xz /tmp
RUN tar -C / -Jxpf /tmp/s6-overlay-x86_64.tar.xz

# Overhead of files to manage processes via s6
COPY s6-overlay /etc/s6-overlay
# Copy the script we intend to run
COPY myscript.sh /home

# CMD is the main process - nothing special here
CMD ["/usr/sbin/nginx"]
# ENTRYPOINT must be /init for s6 to work
```

our script has >1 line, we have to our `s6-overlay/s6-rc.d/myapp/up`:

```
/home/myscript.sh
```

`s6-overlay/s6-rc.d/myapp/contents.d/myapp` is an empty file.

Now we simply need to `docker build (...)` and `docker run -p 80:80 (...)`. If you have done everything correctly, you should see a log message `s6-rc: info: service myapp successfully started` at container startup.
You can then visit `localhost:80` and run `docker exec CONTAINER bash -c "cat /home/foo.txt"` to confirm it works as expected.

Note that utilizing `s6-rc.d` is the recommended way to do it. There's also a legacy way to accomplish this with less overhead by putting `myscript.sh` into folder `/etc/cont-init.d/`.

Share   Improve this answer   Follow   edited Apr 6, 2023 at 14:40   answered Apr 6, 2023 at 14:18

nichoio
7,667 ● 4 ● 31 ● 42

My success would be adding my script to CMD or ENTRYPOINT

0

Then add the bootom of the script i made run include the start application at the bottom.

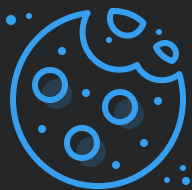apachectl -D FOREGROUND

Share   Improve this answer   Follow   answered Feb 15, 2024 at 19:31

Cyford Technologies LLC
31 ● 7