docker / compose

privileged:true required to run app in container where systemd is pid 1 on Ubuntu 16.04 host #4633

New issue

Open aikchar opened this issue on Mar 17, 2017 · 3 comments

aikchar commented on Mar 17, 2017

I'm trying to run a container (systemd is pid 1) on host Ubuntu 16.04.

```
$ docker-compose --version
docker-compose version 1.11.2, build dfed245
$ docker --version
Docker version 17.03.0-ce, build 3a232c8
```

tl;dr: On Ubuntu 16.04 host `docker run` starts a container and my app in it with *privileged: false* but `docker-compose up` needs *privileged: true*. The same container runs on Fedora 25 host with *privileged: false*.

First reported in docker/docker#28614.

Docker Image

The Docker image is built with the following Dockerfile snippet that runs systemd as pid 1 in the container.

```
FROM centos/centos7

# Fix incompatibility between Docker and systemd
# copy/paste from https://forums.docker.com/t/systemctl-status-is-not-working-in-my-docker-container/9075/4
# additional steps from https://github.com/CentOS/sig-cloud-instance-images/issues/41
RUN (cd /lib/systemd/system/sysinit.target.wants/; for i in *; do [ $i == systemd-tmpfiles-setup.service ] || rm -f $i; done); \
    rm -f /lib/systemd/system/multi-user.target.wants/*; \
    rm -f /etc/systemd/system/*.wants/*; \
    rm -f /lib/systemd/system/local-fs.target.wants/*; \
    rm -f /lib/systemd/system/sockets.target.wants/*udev*; \
    rm -f /lib/systemd/system/sockets.target.wants/*initctl*; \
    rm -f /lib/systemd/system/basic.target.wants/*; \
    rm -f /lib/systemd/system/anaconda.target.wants/*; \
    mkdir -p /etc/selinux/targeted/contexts/ &&\
    echo '<busconfig><selinux></selinux></busconfig>' > /etc/selinux/targeted/contexts/dbus_contexts

VOLUME [ "/sys/fs/cgroup" ]

CMD ["/usr/sbin/init"]

ENV TERM=xterm

# Continue the rest here
```

docker run

When I use `docker run` to create container it starts and the app process starts in the container.

```
$ docker run --name=myapp -d --rm --privileged=false --cap-add=SYS_ADMIN --tmpfs /run --tmpfs /run/lock --tmpfs /tmp -p 2424:2424 -p 2480:2480 -v /sys/fs/cgroup:/sys/fs/cgroup:ro mycontainerimage

$ docker exec -it b2769703c135 /bin/bash

[root@b2769703c135 /]# ps -ef
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 00:30 ?        00:00:00 /usr/sbin/init
root        17     1  0 00:30 ?        00:00:00 /usr/lib/systemd/systemd-journald
root        19     1 37 00:30 ?        00:00:01 /bin/java -server -Xms512m -Xmx512m -Djna.nosys=true -XX:+HeapDumpOnOutOfMemoryError -Djava.awt.headless=true -Dfile.encoding=UTF8 -Drhino.opt.level=9 -Dprofi
root       117     0  1 00:30 ?        00:00:00 /bin/bash
root       135   117  0 00:30 ?        00:00:00 ps -ef
```

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Notifications

3 participants

```
[root@b2769703c135 /]# systemctl
  UNIT                           LOAD   ACTIVE SUB       DESCRIPTION
  -.mount                        loaded active mounted   /
  dev-mqueue.mount               loaded active mounted   POSIX Message Queue File System
<SNIP>
```

docker-compose up

When I use `docker-compose up` the container starts but the app process does not start in the container.

docker-compose.yml:

```
version: '2'
services:
myapp:
    cap_add:
    - SYS_ADMIN
    environment:
    - container=docker
    image: mycontainerimage
    ports:
    - "2424:2424"
    - "2480:2480"
    privileged: false
    tmpfs:
    - /run
    - /run/lock
    - /tmp
    volumes:
    - "/sys/fs/cgroup:/sys/fs/cgroup:ro"
```

Commands:

```
$ docker-compose up -d

$ docker exec -it 70c88ce6df22 /bin/bash

[root@70c88ce6df22 /]# ps -ef
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 00:18 ?        00:00:00 /usr/sbin/init
root        27     0  0 00:32 ?        00:00:00 /bin/bash
root        45    27  0 00:32 ?        00:00:00 ps -ef

[root@70c88ce6df22 /]# systemctl
Failed to get D-Bus connection: Operation not permitted
```

When I modify *docker-compose.yml* to change *privileged: false* to *privileged: true* the container starts and so does the app inside the container.

```
$ docker exec -it 9de86c00efd6 /bin/bash
[root@9de86c00efd6 /]# ps -ef
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 00:40 ?        00:00:00 /usr/sbin/init
root        17     1  0 00:40 ?        00:00:00 /usr/lib/systemd/systemd-journald
root        18     1  4 00:40 ?        00:00:01 /bin/java -server -Xms512m -Xmx512m -
Djna.nosys=true -XX:+HeapDumpOnOutOfMemoryError -Djava.awt.headless=true -Dfile.encoding=UTF8 -
Drhino.opt.level=9 -Dprofi
root       172     0  0 00:40 ?        00:00:00 /bin/bash
root       190   172  0 00:40 ?        00:00:00 ps -ef
```

Diffs between docker inspect

From container created by `docker run` .

```
"HostConfig": {
            "Binds": [
                "/sys/fs/cgroup:/sys/fs/cgroup:ro"
            ],
```

and

```
"Mounts": [
            {
                "Type": "bind",
```

```
                "Source": "/sys/fs/cgroup",
                "Destination": "/sys/fs/cgroup",
                "Mode": "ro",
                "RW": false,
                "Propagation": ""
            }
        ],
```

From container created by `docker-compose up` whether *privileged:false* or *privileged:true*.

```
  "HostConfig": {
            "Binds": [

  "321e05df2c34e16442f2834db896b42c8d0188d5c7a06977b246bbb32270cd5d:/sys/fs/cgroup:rw"
            ],
```

and

```
  "Mounts": [
            {
                "Type": "volume",
                "Name": "321e05df2c34e16442f2834db896b42c8d0188d5c7a06977b246bbb32270cd5d",
                "Source":
  "/var/lib/docker/volumes/321e05df2c34e16442f2834db896b42c8d0188d5c7a06977b246bbb32270cd5d/_data",
                "Destination": "/sys/fs/cgroup",
                "Driver": "local",
                "Mode": "rw",
                "RW": true,
                "Propagation": ""
            }
        ],
```

dnephin commented on Mar 20, 2017 Contributor

https://docs.docker.com/compose/overview/#preserve-volume-data-when-containers-are-created

What's your `docker-compose.yml` ?

You can remove the old container with `docker-compose rm`, and start it again. There should be a warning about the bind volume being masked.

aikchar commented on Mar 27, 2017

docker-compose.yml:

```
  version: '2'
  services:
      myapp:
          cap_add:
          - SYS_ADMIN
          environment:
          - container=docker
          image: mycontainerimage
          ports:
          - "2424:2424"
          - "2480:2480"
          privileged: false
          tmpfs:
          - /run
          - /run/lock
          - /tmp
          volumes:
          - "/sys/fs/cgroup:/sys/fs/cgroup:ro"
```

Which old container do I remove? I'm creating two containers from the same image; one with `docker up` and the other with `docker-compose up`. I'm sorry I didn't get what you meant.

guillaumeparis2000 commented on Oct 19, 2017

Hi **@aikchar** have you find a solution for that problem?
Thanks